Networking Fundamentals

▼ Contents

▼ TCP/IP Model and OSI Model

What is a Networking Model?

A **network model** is like a blueprint that tells us how data should be sent from one device to another across networks (like the Internet). It breaks everything down into layers — each handling a specific task.

OSI Model

The **OSI (Open Systems Interconnection) model** has **7 layers** — more detailed and used for learning and designing protocols.(Theoretical, Conceptual Model)

Mnemonic: "**P**lease **D**o **N**ot **T**hrow **S**ausage **P**izza **A**way"

Layer No.	Name	Function
7	**Application**	Interfaces with the user (HTTP, FTP)
6	**Presentation**	Data translation, encryption, compression (e.g., JPEG, SSL)
5	**Session**	Maintains sessions between apps (e.g., logging in)
4	**Transport**	Reliable delivery (TCP/UDP), segmentation
3	**Network**	Routing, IP addressing

2	**Data Link**	MAC addressing, framing, error detection
1	**Physical**	Transmission of raw bits (cables, signals, voltages)

Real-world Explanation

- **Application** – You write the message.
- **Presentation** – You write it in a language your friend understands.
- **Session** – You keep the conversation alive with multiple letters.
- **Transport** – You cut the letter into pages and number them.
- **Network** – You put an address on the envelope.
- **Data Link** – The post office adds their stamp and tracking info.
- **Physical** – The truck or plane physically delivers the letter.

TCP/IP Model

The **Transmission Control Protocol / Internet Protocol model** is the model the Internet actually uses. It has **4 layers**:

1. Application Layer

- What it does: Provides services to the user.
- Examples: HTTP (web browsing), FTP (file transfer), SMTP (email), DNS (domain lookup).
- Think of this like the "top-level" where the user interacts (e.g., typing a URL in a browser).

2. Transport Layer

- What it does: Provides reliable or unreliable delivery.
- Protocols: **TCP** (reliable, like a phone call), **UDP** (unreliable, like a text message).
- It breaks data into segments and makes sure they reach properly.

3. Internet Layer

- What it does: Handles addressing and routing.

- Protocols: **IP**, **ICMP**, **ARP**.
- Decides **which path** data should take to reach the destination.

4. Network Access Layer (or Link Layer)

- What it does: Deals with physical transmission of data.
- Includes: Ethernet, Wi-Fi, etc.
- Involves MAC addressing, switches, etc.

TCP/IP vs OSI

Feature	TCP/IP Model	OSI Model
Layers	4	7
Developed By	US DoD	ISO
Usage	Real-world Internet	Theoretical/educational
Application Layer	Combines OSI's App + Presentation + Session	3 separate layers
Transport Layer	TCP, UDP	TCP, UDP
Network Layer	IP	IP
Link + Physical	Combined into 1	Separated
Protocols Defined	Yes	No (it's a reference model)
Flexibility	More practical	More structured and strict

▼ IPv4 and IPv6 Addressing

What is an IP Address?

An **IP address (Internet Protocol address)** is a **unique numerical identifier** assigned to each device connected to a computer network that uses the Internet Protocol for communication.

It serves **two main purposes**:

1. **Identification** – to uniquely identify the device on the network
2. **Location addressing** – to specify where the device is located in the network so data can be routed correctly

IPv4 (Internet Protocol version 4)

IPv4 is the **fourth version of the Internet Protocol** that uses a **32-bit address** format to identify devices on a network, allowing around **4.3 billion unique addresses** (e.g., `192.168.1.1`).

It has two main sections:

- **Network Portion**: Identifies the network the device belongs to.
- **Host Portion**: Uniquely identifies a device within the network.

IPv4 addresses are divided into classes based on the length of the network and host portions:

- **Class A**: 8-bit network ID, 24-bit host ID.
- **Class B**: 16-bit network ID, 16-bit host ID.
- **Class C**: 24-bit network ID, 8-bit host ID.

IPv6 (Internet Protocol version 6)

IPv6 is the **newer version of the Internet Protocol** designed to replace IPv4. It uses a **128-bit address** format, allowing a **massive number of unique addresses**, and includes **better security, speed, and efficiency** (e.g., `2001:0db8:85a3::8a2e:0370:7334`).

Strategies For Switching From IPV4 to IPV6

- **Dual Stacking** : Devices can use both IPv4 and IPv6 at the same time. This way, they can talk to networks and devices using either version.
- **Tunneling** : This method allows IPv6 users to send data through an IPv4 network to reach other IPv6 users. Think of it as creating a "tunnel" for IPv6 traffic through the older IPv4 system.
- **Network Address Translation (NAT)** : NAT helps devices using different versions of IP addresses (IPv4 and IPv6) to communicate with each other by translating the addresses so they understand each other

IPv4 VS IPv6

Feature	IPv4	IPv6
Address Length	32-bit address	128-bit address
Address Format	Decimal format (e.g., 192.168.0.1)	Hexadecimal format (e.g., 2001:0db8::1)
Configuration	Manual and DHCP configuration	Auto-configuration and renumbering supported
Connection Integrity	End-to-end integrity is unachievable	End-to-end integrity is achievable
Security	No built-in security; external tools like IPSec needed	IPSec is built-in for encryption and authentication
Fragmentation	Performed by sender and routers	Performed only by the sender
Flow Identification	Not available	Uses Flow Label field in header for packet flow identification
Checksum Field	Present	Not present
Transmission Scheme	Supports broadcast	Uses multicast and anycast; no broadcast
Header Size	Variable: 20–60 bytes	Fixed: 40 bytes
Conversion	Can be converted to IPv6	Not all IPv6 addresses can be converted to IPv4
Field Structure	4 fields separated by dots (.)	8 fields separated by colons (:)

Feature	IPv4	IPv6
Address Classes	Has address classes (A, B, C, D, E)	No concept of address classes
VLSM Support	Supports Variable Length Subnet Mask (VLSM)	Does not support VLSM
Example	66.94.29.13	2001:0000:3238:DFE1:0063:0000:0000:FEFB

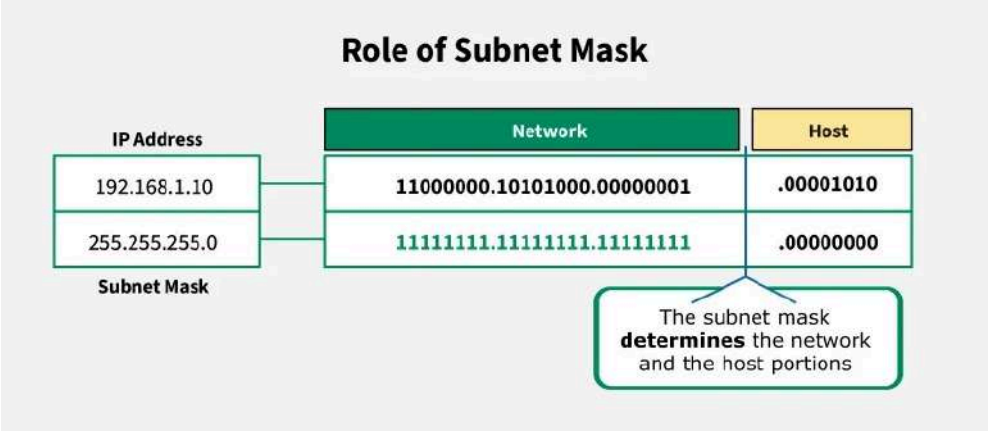
Subnet

A subnet is like a smaller group within a large network. It is a way to split a large network into smaller networks so that devices present in one network can transmit data more easily. For example, in a company, different departments can each have their own subnet, keeping their data traffic separate from others.

By subnetting, we:

- **Save IP addresses (Efficiency)**
- **Keep networks faster (Better Performance)**
- **Protect sensitive data (Improved Security)**

Subnet Mask



Supernet

Supernetting is the process of combining multiple smaller networks (subnets) into a larger network (supernet).Supernetting is mainly used in Route Summarization,

where routes to multiple networks with similar network prefixes are combined into a single routing entry, with the routing entry pointing to a Super network, encompassing all the networks.

By Supernetting, we:

- Control and reduce network traffic
- Helpful to solve the problem of lacking IP addresses
- Minimizes the routing table i.e, it cannot cover a different area of the network when combined and all the networks should be in the same class and all IP should be contiguous

CIDR

CIDR stands for Classless Inter-Domain Routing. It is a method for **allocating IP addresses and routing IP packets** more efficiently than the old **classful** system (Class A, B, C...).

It allows variable-length subnet masks.
CIDR is a flexible way to divide IP addresses using
prefix lengths like `/24`, `/25`, `/20`, etc., rather than being stuck with class A, B, C rules.

Notation:

192.168.1.0/24
The /24 means the first 24 bits are the network portion

The remaining 8 bits (32 - 24) are for hosts

NAT

It is a Private **IP address** or local address that is translated into the public IP address. NAT is used to slow down the rate of decrease of the available IP address by translating the local IP or Private IP address into a global or public IP address. NAT can be a one-to-one relation or many-to-one relation.

PAT

In Port Address Translation (PAT), Private IP address are translated into the public IP address through port numbers. PAT also uses IPv4 address but with port number. It

have two types:

1. Static
2. Overloaded PAT

NAT VS PAT

Network Address Translation (NAT)	Port Address Translation (PAT)
NAT stands for Network Address Translation.	**PAT** stands for Port Address Translation.
In NAT, Private IP addresses are translated into the public IP address.	In PAT, **Private IP addresses** are translated into the public IP address via Port numbers.
NAT can be considered PAT's superset.	PAT is a dynamic NAT.
NAT uses **IPv4** address.	PAT also uses IPv4 address but with port number.
It have 3 types: Static, Dynamic NAT and PAT/ NAT Overloading/IP masquerading.	It also have two types: Static and Overloaded PAT.

Public VS Private IP Address

The **Private IP Address** of a system is the IP address that is used to communicate within the same network. The router basically assigns these types of addresses to the device. Unique private IP Addresses are provided to each and every device that is present on the network. These things make Private IP Addresses more secure than Public IP Addresses.

The Public IP Address of a system is the IP address that is used to communicate outside the network. A public IP address is basically assigned by the **ISP (Internet Service Provider)**.

Public IP Address is basically of two types:

- **Dynamic IP Address:** **Dynamic IP Address** are addresses that change over time. After establishing a connection of a smartphone or computer with the Internet, ISP provides an IP Address to the device, these random addresses are called Dynamic IP Address.
- **Static IP Address:** **Static IP Address** are those addresses that do not change with time. These are stated as permanent internet addresses. Mostly these are

used by the **DNS (Domain Name System)** Servers.

Private IP Address	Public IP Address
The scope of Private IP is local.	The scope of Public IP is global.
It is used to communicate within the network.	It is used to communicate outside the network.
Private IP addresses of the systems connected in a network differ in a uniform manner.	Public IP may differ in a uniform or non-uniform manner.
It works only on LAN.	It is used to get internet service.
It is used to load the network operating system.	It is controlled by ISP.
It is available free of cost.	It is not free of cost.
Private IP can be known by entering “ipconfig” on the command prompt.	Public IP can be known by searching “what is my ip” on Google.
Range:*10.0.0.0 – 10.255.255.255,* ***172.16.0.0 – 172.31.255.255,*** ***192.168.0.0 – 192.168.255.255***	**Range:** Besides private IP addresses, the rest are public.
Example: 192.168.1.10	Example: 17.5.7.8
Private IP uses numeric code that is not unique and can be used again	Public IP uses a numeric code that is unique and cannot be used by other
Private IP addresses are secure	The public IP address has no security and is subjected to attack
Private IP addresses require NAT to communicate with devices	Public IP does not require a network translation

▼ Network Devices and Protocols

Router

Definition: A router is a network device that connects multiple networks together and forwards data packets between them based on their IP addresses.

Functions:

- Operates at Layer 3 (Network Layer) of the OSI model
- Routes data between different networks using IP addresses

- Creates boundaries between networks and provides security
- Enables internet connectivity for all connected devices

Examples: Home Wi-Fi routers, enterprise routers (Cisco, Juniper)

Switch

Definition: A switch is a network device that connects multiple devices on the same network and forwards data packets to specific destinations based on MAC addresses.

Functions:

- Operates at Layer 2 (Data Link Layer) of the OSI model
- Creates direct connections between devices using MAC addresses
- Maintains a MAC address table to efficiently forward traffic
- Provides full bandwidth to each connected device

Examples: Office network switches, managed switches, unmanaged switches

Hub

Definition: A hub is a simple network device that connects multiple devices in a network, but forwards incoming data packets to all connected devices regardless of the intended recipient.

Functions:

- Operates at Layer 1 (Physical Layer) of the OSI model
- Broadcasts all data to every connected device (no filtering)
- Creates a shared collision domain, reducing network efficiency
- Offers no traffic management or security features

Examples: Largely obsolete and replaced by switches in modern networks

ARP (Address Resolution Protocol)

- Maps IP addresses to MAC addresses on a local network
- Essential for data link layer communication
- Used before devices can communicate on a LAN

DHCP (Dynamic Host Configuration Protocol)

- Automatically assigns IP addresses to devices on a network
- Eliminates manual IP configuration
- Manages IP address leases and renewals
- UDP port 67 (server) and 68 (client)

DNS (Domain Name System)

- Translates domain names (like www.instagram.com) to IP addresses
- Hierarchical naming system for computers and services
- Enables users to use memorable names instead of numeric IP addresses
- UDP/TCP port 53

ICMP (Internet Control Message Protocol)

- Used for error reporting and network diagnostics
- Operates at the Internet layer of TCP/IP
- Used by tools like ping and traceroute

SNMP (Simple Network Management Protocol)

- Used for monitoring and managing network devices
- Collects information from network devices
- Enables administrators to monitor network performance
- UDP port 162 (traps)
- UDP port 161 (queries)

FTP (File Transfer Protocol)

- Used for transferring files between computers on a network
- Works at the Application layer of TCP/IP
- Uses separate connections for commands and data transfer
- TCP port 20 (data transfer)
- TCP port 21 (command/control)

HTTP (Hypertext Transfer Protocol)

- Foundation of data communication on the World Wide Web
- Works at the Application layer
- Uses port 80
- Not secure; data transmitted in plain text

HTTPS (Hypertext Transfer Protocol Secure)

- Secure version of HTTP using encryption
- Uses SSL/TLS for secure communications
- Uses port 443
- Provides authentication, privacy, and integrity

Network Discovery Protocols

Network discovery protocols are specialized protocols that help devices on a network automatically find and learn about each other.

1. ARP (Address Resolution Protocol)

Maps IP addresses to MAC addresses on a local network. While primarily for address resolution, it helps in discovering devices at the data link layer.

2. DHCP (Dynamic Host Configuration Protocol)

Helps devices discover network configuration parameters including IP addresses. It's essential for network discovery as new devices join a network.

3. CDP (Cisco Discovery Protocol)

CDP (Cisco Discovery Protocol) is a proprietary network protocol developed by Cisco Systems that enables network administrators to gather information about directly connected Cisco equipment.

Operating at the data link layer of the OSI model, CDP automatically broadcasts information about device capabilities, hardware specifications, and network settings.

This protocol runs by default on most Cisco devices, sending periodic multicasts containing device information which is then stored in a local CDP table.

While CDP provides valuable insights for network mapping and troubleshooting, it can pose security risks if left enabled in production environments facing untrusted networks, as it may reveal sensitive information about your network infrastructure.

4. LLDP (Link Layer Discovery Protocol)

LLDP (Link Layer Discovery Protocol) serves as an industry-standard alternative to CDP, defined by the IEEE 802.1AB standard.

Unlike Cisco's proprietary solution, LLDP is vendor-neutral, allowing network devices from different manufacturers to advertise their identity, capabilities, and neighbors across multi-vendor environments.

Similar to CDP, LLDP operates at the data link layer but differs in that it's typically disabled by default on most devices.

The protocol stores discovered information in the LLDP Management Information Base (MIB) and uses a Type-Length-Value (TLV) format for information exchange.

LLDP proves particularly valuable in heterogeneous networks with equipment from multiple vendors, where CDP would provide only partial discovery capabilities.

5. mDNS (Multicast DNS)

Used for zero-configuration networking, allowing devices to discover services on a local network without traditional DNS servers.

▼ Network Scanning

Nmap

Nmap (“Network Mapper”) is a free and open source utility for network exploration and security auditing. It uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.

```
pentester@TryHackMe$ sudo nmap -sS 10.10.48.181

Starting Nmap 7.60 ( https://nmap.org ) at 2021-08-30 09:53 BST
Nmap scan report for 10.10.48.181
Host is up (0.0073s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
111/tcp open  rpcbind
143/tcp open  imap
MAC Address: 02:45:BF:8A:2D:6B (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.60 seconds
```

Port Scanning Using Nmap

Scanning network ports to discover open, closed, or filtered ports on a device.

Common Nmap commands:

- **Basic scan**: `nmap <target>` – scans default 1,000 ports.
- **All ports**: `nmap -p- <target>` – scans all 65,535 TCP ports.

Scan types:

- **TCP SYN ("-sS")**: Sends SYN, waits for SYN/ACK → faster, stealthy. Closed if RST returned; filtered if no response.
- **TCP Connect ("-sT")**: Completes full handshake. Easier to detect.
- **UDP ("-sU")**: Sends UDP packets; open|filtered if no reply, closed if ICMP port unreachable.

Service Enumeration & Version Detection

Once open ports are found, identify *what service* (e.g., HTTP, SSH) and *which version* is running.

Use `sV` flag: `sudo nmap -p- -sV -v <target>`

- `p-`: scan all ports
- `sV`: version detection
- `v`: verbose, shows progress in real-time

What it finds:

- Service protocol (e.g., Apache HTTP, OpenSSH)
- Software version (e.g., Apache httpd 2.4.29)
- Many other details (hostname, CPE, OS hints)

Banner Grabbing: Captures welcome messages like `Server: Apache/2.0.46` to identify apps/versions.

Identifying Open Ports & Vulnerabilities

Port states:

- **Open**: listening (SYN/ACK response)
- **Closed**: reachable but no service (RST)
- **Filtered**: no response (firewall)

  Additional states: unfiltered, open|filtered, closed|filtered

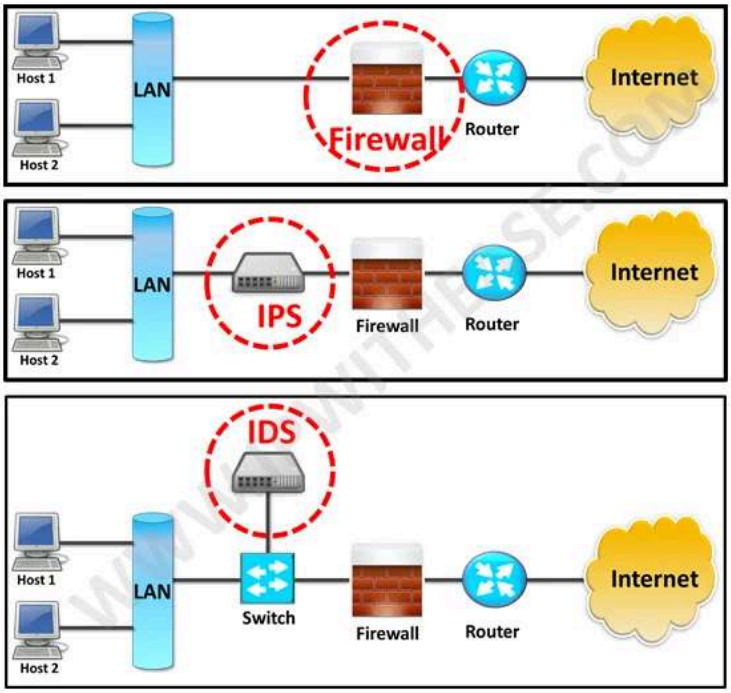
Vulnerability scanning:

- Use **Nmap Scripting Engine (NSE)** – e.g., `-script vuln` or `sV --script=vulners` to identify CVEs.
- Outputs known vulnerabilities with links to databases.

Example command: `nmap -p- -sV --script=vulners -oA scan_results <target>`

Task	Command
Default port scan	`nmap <target>`
Scan all TCP ports	`nmap -p- <target>`
SYN scan	`nmap -sS <target>`
UDP scan	`nmap -sU <target>`
Version detection	`nmap -sV <target>`
Full scan + OS detection	`nmap -A -T4 <target>`
Vulnerability check via NSE	`nmap -sV --script=vulners <target>`

▼ Firewalls, VPNs, IDS/IPS



Firewall

A network-based or host-based security system that **monitors and controls traffic** based on configurable rules.

Key Features:

- Operates at **Layer 3/4**, often up to **Layer 7** in Next-Gen Firewalls.
- Uses **stateful inspection** to track session states (e.g., ESTABLISHED, RELATED).
- Filters based on **IP addresses, ports, protocols**.

Use Cases:

Perimeter defense, DMZ protection, internal segmentation.

VPN (Virtual Private Network)

A secure and **encrypted tunnel** over untrusted networks, allowing remote users to access private networks safely.

Key Features:

- **Authentication & Encryption** (e.g., IPsec, SSL/TLS)
- **IP masking** and secure access over public Wi-Fi.

- Commonly used for **remote work** and **geo-restricted access**.

IDS (Intrusion Detection System)

A system that **monitors traffic or host activity for malicious behavior**, generating alerts without blocking traffic.

Types:

- **NIDS** – Network-based (traffic monitoring)
- **HIDS** – Host-based (file and process monitoring)

Detection Methods:

- **Signature-based** (known attack patterns)
- **Anomaly-based** (deviations from baseline behavior)

Workflow: Passive → Logs/Alerts → Manual response.

IPS (Intrusion Prevention System)

An **inline system** that builds on IDS capabilities to **detect and actively block** malicious traffic.

Key Features:

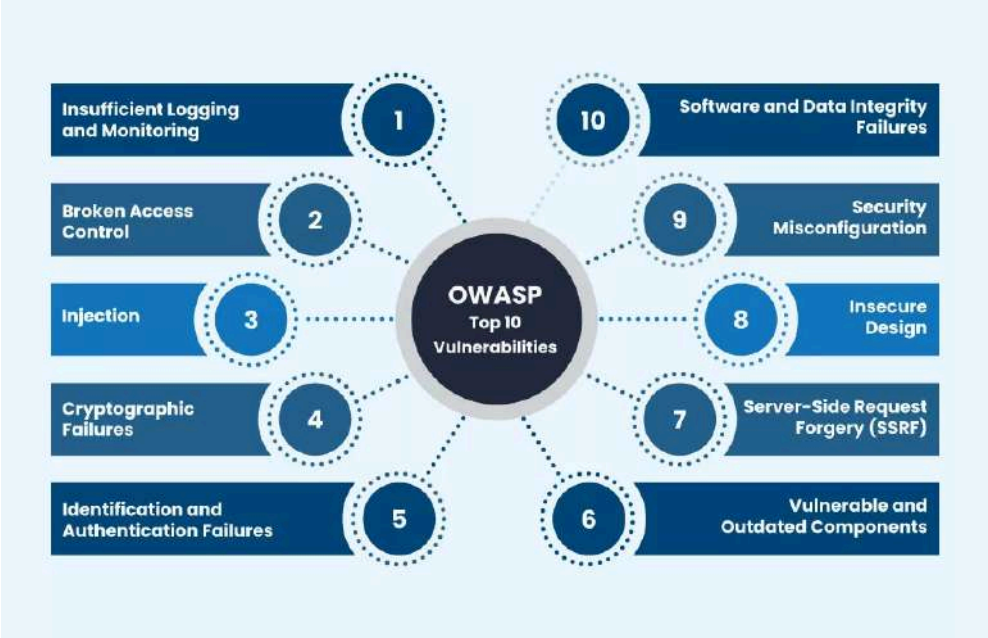
- Positioned **after firewall** on traffic path.
- Can **drop packets, reset connections, block IPs** on detection.
- Often integrated into **UTM/NGFW** platforms.

Component	Functions	Traffic Path	Action	Placement
Firewall	Filter/allow traffic by rules	Inline	Allow/Block	At perimeter
VPN	Encrypt/authenticate traffic	Tunnel	Secure access	Endpoint or gateway
IDS	Monitor & alert on threats	Tap/SPAN (copy)	Alert/log	Inside or edge
IPS	Detect & **prevent** intrusions	Inline	Drop/block traffic	Inline, after firewall

How They Work Together

1. **Firewall** filters basic, rule-based traffic. → **Prevent**
2. **VPN** secures remote connections behind the firewall. → **Secure access**
3. **IDS** monitors allowed traffic; alerts admins on suspicious events. → **Detect & alert**
4. **IPS** actively blocks detected threats in real time. → **Prevent & respond**

▼ OWASP Top 10 Vulnerabilities



A community-backed list of the most critical web application security risks, updated every few years. It's used as a global standard for secure development and compliance.

OWASP Top 10 is **industry-standard**, improves developer security awareness, helps meet compliance, and guides penetration testing and defenses.

#	Vulnerability	Description	Impact	Mitigation
A01	Broken Access Control	Users access data/functionality beyond authorization (Wikipedia, LinkedIn)	Data breaches, admin-level misuse	Enforce RBAC/MAC, verify authorizations server-side, deny-by-default policy

#	Vulnerability	Description	Impact	Mitigation
A02	Cryptographic Failures	Sensitive data poorly encrypted or using weak algorithms	Data theft, compliance violations (e.g. GDPR, PCI DSS)	Use TLS 1.2+, strong ciphers, manage keys securely, encrypt in transit and at rest
A03	Injection (incl. XSS)	Attackers inject malicious payloads (SQL, NoSQL, XSS)	Data loss, code execution, stolen cookies	Use parameterized queries, filter/validate input, escape contexts, content-security-policy
A04	Insecure Design	Architecture-level flaws due to missing controls	Entire system compromised	Threat modeling, secure design patterns, design reviews & reference architectures
A05	Security Misconfiguration	Misconfigured servers, frameworks, permissions	Information leaks, unauthorized access	Harden OS/services, disable unused features, use automated config scanning like CIS Bench
A06	Vulnerable & Outdated Components	Use of libraries/frameworks with known vulnerabilities	Exploits via known CVEs	Maintain SBOM, update dependencies, scan for vulnerabilities
A07	Identification & Authentication Failures	Weak auth allowing bypass, brute-force, session hijack	Account takeover, privilege escalation	MFA, secure password storage (bcrypt), session timeout,

#	Vulnerability	Description	Impact	Mitigation
				monitor auth logs
A08	Software & Data Integrity Failures	Unsigned updates, insecure CI/CD pipelines	Malware injection, compromised apps	Enforce code signing, validate integrity, verify software supply chain
A09	Security Logging & Monitoring Failures	Poor detection & alerting of breaches	Long dwell times, delayed response	Log all security events, centralize logs, set alerts, conduct SIEM/UEBA monitoring
A10	Server-Side Request Forgery (SSRF)	Server fetches attacker-supplied URLs without validation	Internal resource access, port scanning, SSRF chaining	Validate/allowlist URLs, block local IPs, use network-level isolation