NextGen Security Careers: Pathway for Fresh Graduates to Become Security Engineers in Industry

หลักสูตร

วิศวกรความมั่นคงปลอดภัย Security Engineer

ภายใต้โครงการเส้นทางสู่วิศวกรความมั่นคงปลอดภัย สำหรับนักศึกษาจบใหม่สู่การทำงานในภาคอุตสาหกรรม

Speaker

- นาย เจษฎา ทองก้านเหลือง (ต้อม)
- กรรมการผู้จัดการ (Managing Director)
- บริษัท ที-เน็ต ไอที โซลูชัน จำกัด
- Certificate: CCNA, CEH, CHFI ,ECSA, CISO
- CompTIA Security+, CompTIA Pentest+, CompTIA Project+, CompTIA Network+, CompTIA CySA+, CompTIA Cloud+
- Peplink Certified Engineer (PCE), Peplink Sales Specialist (PSS)
- Fortinet NSE1, Fortinet NSE2
- IT Specialist Certification (ITS): Cyber Security, Network Security
- Certificate Name: Certificate of Competence in Zero Trust, Certificate of Cloud Security Knowledge
- E-Mail : Jedsada@tnetitsolution.co.th
- Facebook: ผู้ดูแลกลุ่มสอนแฮกแบบหมูๆ
- Website : www.tnetitsolution.co.th

Agenda

- ❑ Introduction to Network Security
- ❑ Network Architecture & Threat Surface
- ❑ Firewalls, IDS/IPS, and Access Controls
- ❑ Network Protocols & Encryption
- ❑ Monitoring, Logging, and Incident Response
- ❑ Common Attacks & Defense Mechanisms
- ❑ Best Practices

Introduction to Network Security

Why Security ?

- เพราะกลัว Threat (ภัยคุกคาม) ทำให้เกิดความสุญเสีย (lossed) ทางด้าน
  - กายภาพ (Life/Physical)
  - ทรัพย์สิน (Money/Asset/Infomation)
  - ความเป็นส่วนตัว (Privacy)
  - สิทธิ (Right)
  - เสรีภาพ (Freedom)
  - อื่นๆ

What is Threat of IT Security Risk ?

- ❑ Hacker/Attacker
- ❑ Malware (e.g. Worm, Virus, Trojan, Adware, and etc)
- ❑ Disaster (e.g. Earthquake, Flood, Fire, Electricity loss, and etc)
- ❑ Crisis (e.g. Protest, Pestilence, War, Immigration, and etc)

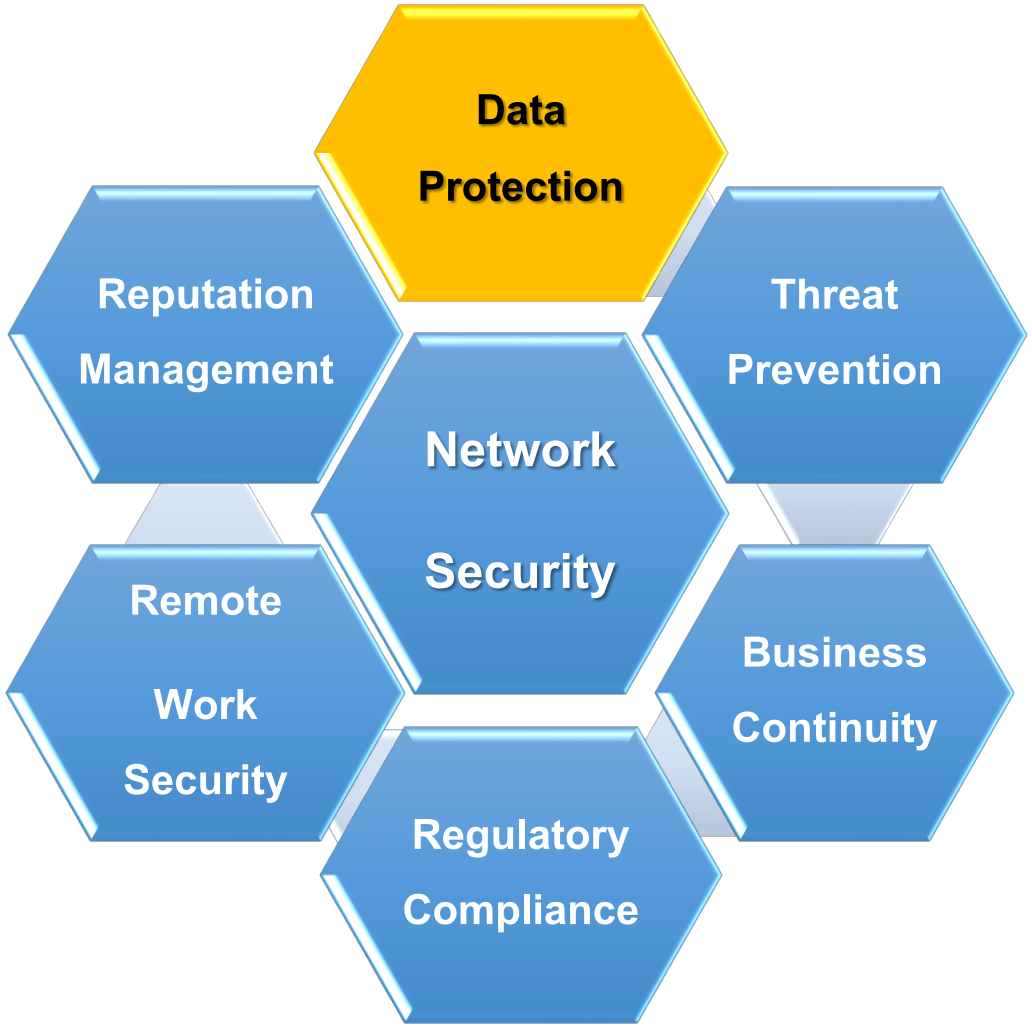
What is Network Security?

- **Network security** is the practice of protecting computer networks and the data that travels through them from unauthorized access, misuse, modification, or destruction.
- It involves a combination of **hardware**, **software**, **policies**, and **procedures** designed to defend the **integrity**, **confidentiality**, and **availability** of data and systems connected through a network

Why Network Security is Important

- ❑ Protects sensitive business and personal data
- ❑ Prevents cyberattacks such as malware, phishing, DDoS
- ❑ Ensures business continuity
- ❑ Helps meet legal and regulatory compliance (e.g., GDPR, HIPAA)
- ❑ Builds customer trust and protects brand reputation

Importance of network security in modern environments



❑Protecting Sensitive Data

❑Modern organizations handle vast amounts of confidential data — such as personal information, financial records, and intellectual property. Network security ensures that this data is:

- **Protected from unauthorized access**
- **Encrypted during transmission**
- **Monitored for suspicious activity**

❑Without proper safeguards, a single breach can lead to identity theft, data leaks, or reputational damage. Reputation Management

❑Preventing Cyberattacks

❑Cyber threats like ransomware, phishing, Distributed Denial of Service (DDoS) attacks, and malware are constantly evolving.

A robust network security system:

- Detects and blocks malicious traffic
- Prevents attackers from exploiting vulnerabilities
- Reduces downtime and financial losses caused by cyber incidents

❑In modern environments, proactive prevention is far more cost-effective than reacting after an attack occurs.

❑Ensuring Business Continuity

❑Network availability and reliability are vital for daily operations. Security tools such as **firewalls**, **intrusion prevention systems**, and **redundant network design** help maintain uptime by:

- Detecting and mitigating threats before they disrupt services
- Protecting mission-critical systems
- Supporting disaster recovery and incident response efforts

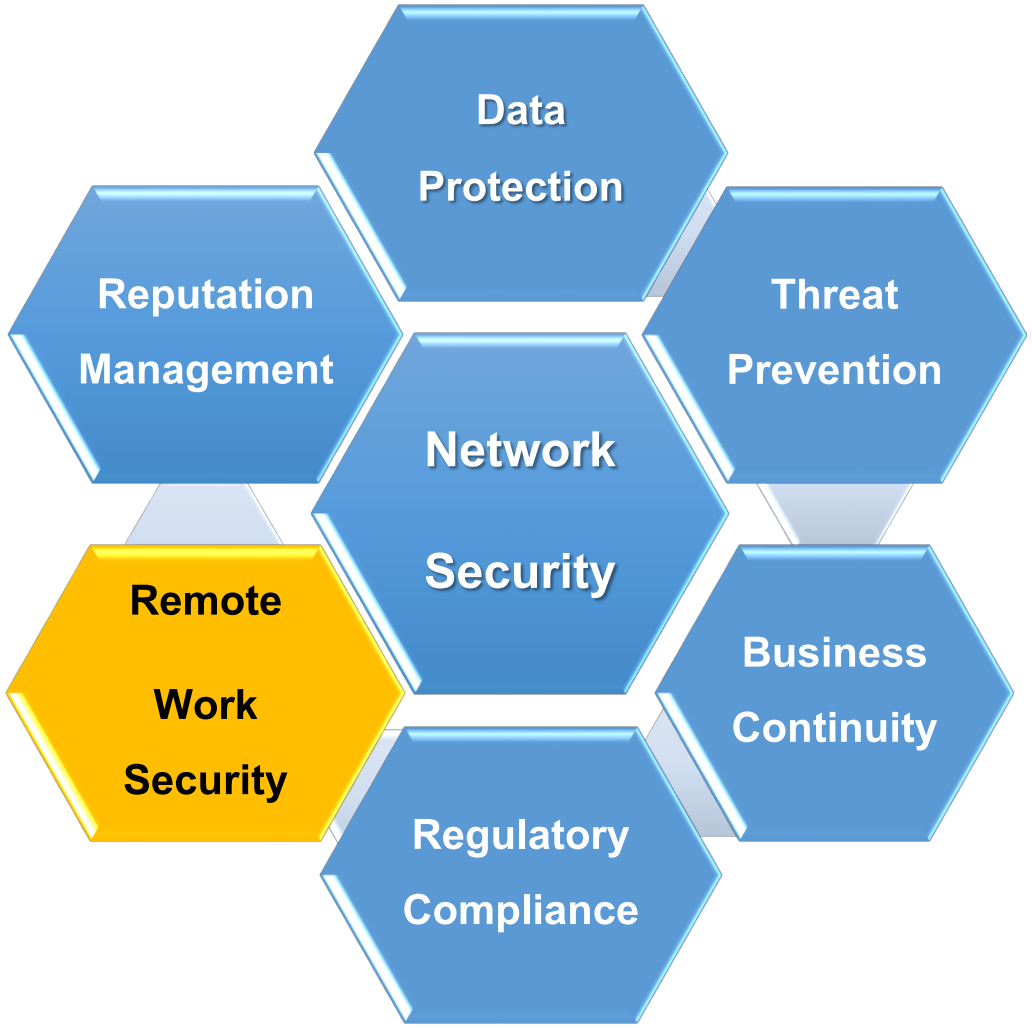
❑Downtime due to a network attack can cause severe productivity and financial losses.

❑Maintaining Customer Trust and Compliance

❑Clients, partners, and regulators expect organizations to safeguard their data. Implementing network security helps meet legal and regulatory requirements such as:

- **GDPR** (General Data Protection Regulation)
- **ISO/IEC 27001** (Information Security Management Standard)
- **NIST Cybersecurity Framework**

❑A strong security posture demonstrates professionalism and builds customer confidence



❑Supporting Digital Transformation and Remote Work

❑With cloud computing, remote access, and IoT integration, networks are expanding beyond traditional office boundaries.

Network security:

- Enables **secure remote access** via VPNs
- Protects **cloud-based assets** and SaaS applications
- Manages security for **hybrid and distributed environments**

❑Without adequate protection, digital transformation initiatives can become new attack vectors

❑Protecting Reputation and Reducing Financial Losses

❑Cyber incidents not only disrupt operations but can also cause:

- Regulatory fines
- Legal liabilities
- Loss of customer trust
- Damage to brand image

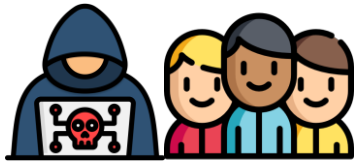
❑Investing in network security minimizes these risks and demonstrates a commitment to responsible data stewardship.

Common threats and attack vectors

Types of Network Attacks

Malware

DDoS Attacks



Man-in-the-Middle

Ransomware

IP Spoofing

Configuration issues

Packet sniffer

Phishing

DNS Spoofing

Types of Network Attacks

Malware

Definition: Software intentionally designed to cause damage, steal data, or gain unauthorized access.

Types: Viruses, Trojans, Worms, Spyware, Adware, etc..

Example: A Trojan disguised as a PDF viewer that installs a backdoor.

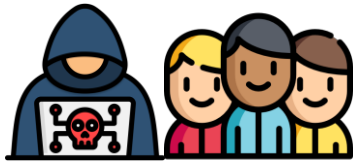
DDoS Attacks

Definition: Attackers use a botnet to flood a server or network with massive traffic to overwhelm it.

Goal: Make services unavailable to legitimate users.

Example: A retail site going offline due to a sudden flood of requests during a sale.

Types of Network Attacks



Man-in-the-Middle

Definition: Attacker intercepts communications between two parties without their knowledge.

Goal: Eavesdrop, steal data (like credentials), or inject malicious content.

Common Techniques: Session hijacking, HTTPS spoofing, SSL stripping

Example: A rogue Wi-Fi hotspot at a coffee shop intercepting login details.

Ransomware

Definition: Malware that encrypts data and demands a ransom to unlock it.

Effect: Business operations can halt; loss of access to critical data.

Example: WannaCry (2017) encrypted data across healthcare networks worldwide.

Types of Network Attacks

IP Spoofing

Definition: Attacker sends IP packets with a forged source IP address to impersonate another device.

Goal: Evade firewalls, perform MitM, or launch DoS/DDoS.

Example: Bypassing access control that trusts a specific IP.

Configuration issues

Definition: Insecure or incorrect system settings that expose the network to attack.

Common Issues: Open ports, Default credentials, Exposed admin interfaces, Poor firewall rules

Example: Exposing a cloud storage bucket with sensitive data to the public.

Types of Network Attacks

Packet sniffer

Definition: Capturing network traffic to analyze data in transit.

Goal: Steal credentials, session tokens, or other sensitive data.

Example: An attacker on public Wi-Fi captures login credentials from users accessing websites over HTTP (unencrypted).

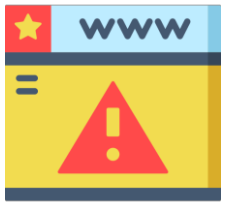
Phishing

Definition: Social engineering attack where attackers trick users into revealing sensitive information.

Types: Email phishing, Spear phishing, Whaling, Smishing, etc..

Example: Fake password reset email from "Microsoft" asking for login details.

Types of Network Attacks



DNS Spoofing

Definition: Attacker corrupts a DNS resolver's cache, redirecting users to malicious sites.

Goal: Divert traffic, steal data, or serve malware.

Example: Typing "yourbank.com" goes to a malicious IP that looks like the real site.

CIA Triad

CIA Triad

การรักษาความลับของสารสนเทศ

- Encryption
- Access Controls

ความถูกต้องของข้อมูลสารสนเทศ

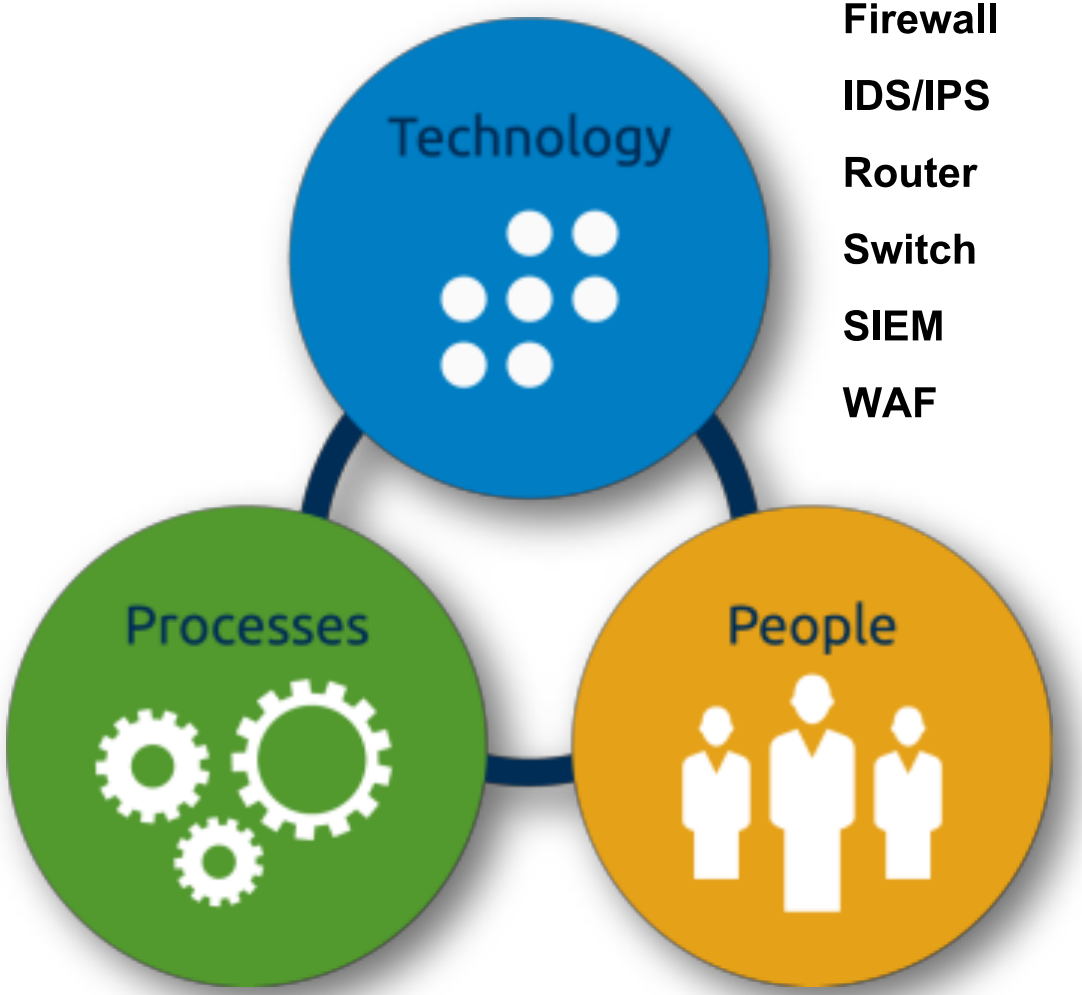
- Hash function
- Digital Signatures

Information Security

ความพร้อมใช้งานของเทคโนโลยีสารสนเทศ

- Redundancy
- Load Balancing

Information Security Management



Firewall

IDS/IPS

Router

Switch

SIEM

WAF

Risk Assessment

Incident Response Plan

Information Security Policy

Continuous Improvement

Staff Training & Security Awareness

Professional Skills and Certifications

Cyber Hygiene

Network Architecture & Threat Surface

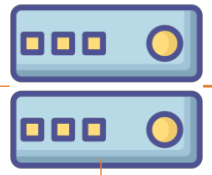
Network Architecture

ISP

Router

External Firewall

IPS



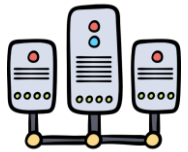
Core Switch

IDS

DMZ Zone

Web Server

Mail Server



Proxy Server

VPN Gateway

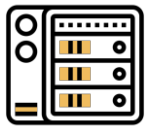
Reverse proxy server

Internal Firewall

Intranal Zone

AD Server+DNS+DHCP

Database Server

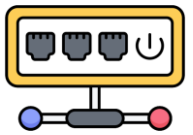


NAS Server

Log Server

NTP Server + SNMP Server + IDS/IPS Server + SIEM

Endpoints Zone

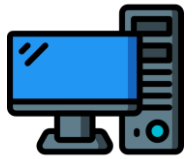


Switch

Mobile

Notebook

Access point

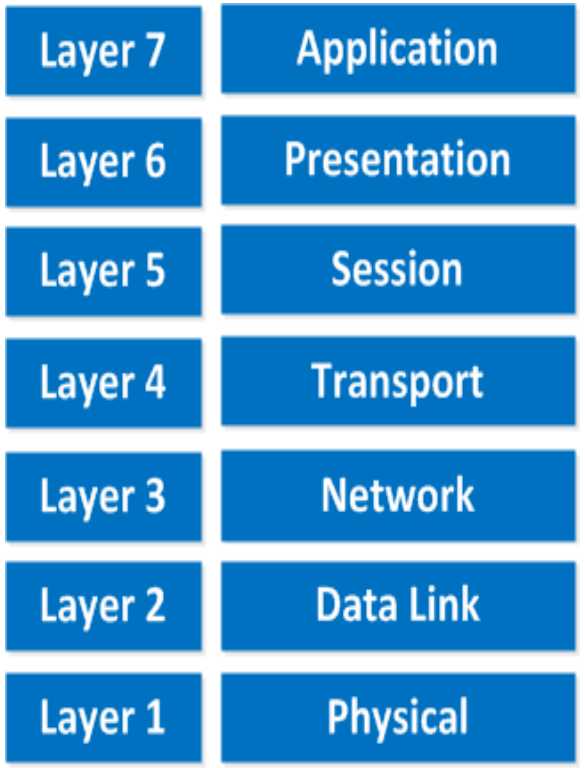


Computer PC

Network layers (OSI model)

OSI Model [1]

- The OSI Model (Open Systems Interconnection Model) is a conceptual framework used to understand and standardize how different networking protocols interact in a telecommunications or computing system. It divides the process of communication in a network into 7 layers, each with specific functions.



OSI Model [2]

- **Physical Layer**
- **Purpose:** Physical transmission of raw bits over a medium.
- **Examples:** Cables, fiber optics, Wi-Fi, Hubs.
- **Functions:** Bit transmission, voltage levels, pin layouts.

DSL Cable

LAN Cable

Layer 7	Application
Layer 6	Presentation
Layer 5	Session
Layer 4	Transport
Layer 3	Network
Layer 2	Data Link
Layer 1	Physical

OSI Model [3]

- **Data Link Layer**
- **Purpose**: Node-to-node data transfer and error detection.
- **Examples**: Ethernet, PPP, MAC, ARP, VLAN.
- **Functions**: Framing, MAC addressing, error detection.

Layer 7	Application
Layer 6	Presentation
Layer 5	Session
Layer 4	Transport
Layer 3	Network
Layer 2	Data Link
Layer 1	Physical

OSI Model [4]

- **Network Layer**
- **Purpose**: Routing, addressing, and delivering packets.
- **Examples**: IP (IPv4/IPv6), ICMP, OSPF, BGP.
- **Functions**: Logical addressing, routing.

Layer 7	Application
Layer 6	Presentation
Layer 5	Session
Layer 4	Transport
Layer 3	Network
Layer 2	Data Link
Layer 1	Physical

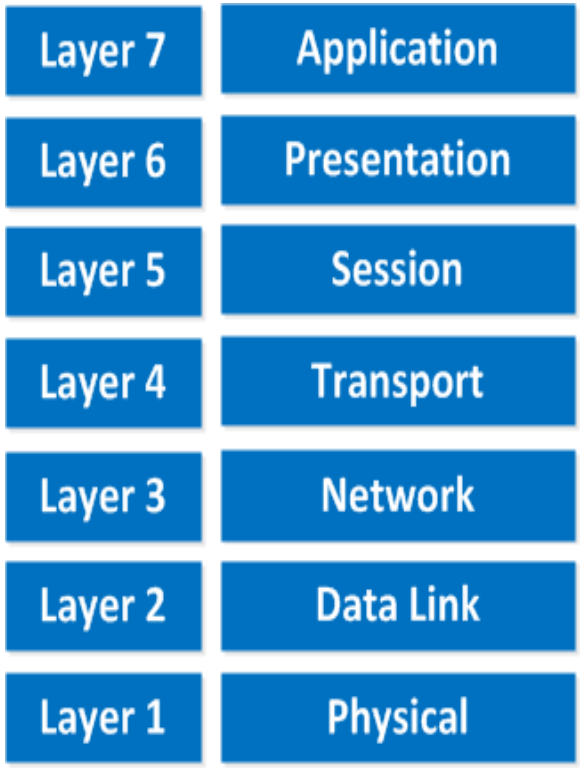
OSI Model [5]

- **Transport Layer**
- **Purpose**: Reliable data transfer between end systems.
- **Examples**: TCP, UDP.
- **Functions**: Flow control, error correction, segmentation/reassembly.

Layer 7	Application
Layer 6	Presentation
Layer 5	Session
Layer 4	Transport
Layer 3	Network
Layer 2	Data Link
Layer 1	Physical

OSI Model [6]

- **Session Layer**
- **Purpose**: Establishes, manages, and terminates sessions between applications.
- **Examples**: NetBIOS, RPC.
- **Functions**: Controls the dialogues (connections) between computers.

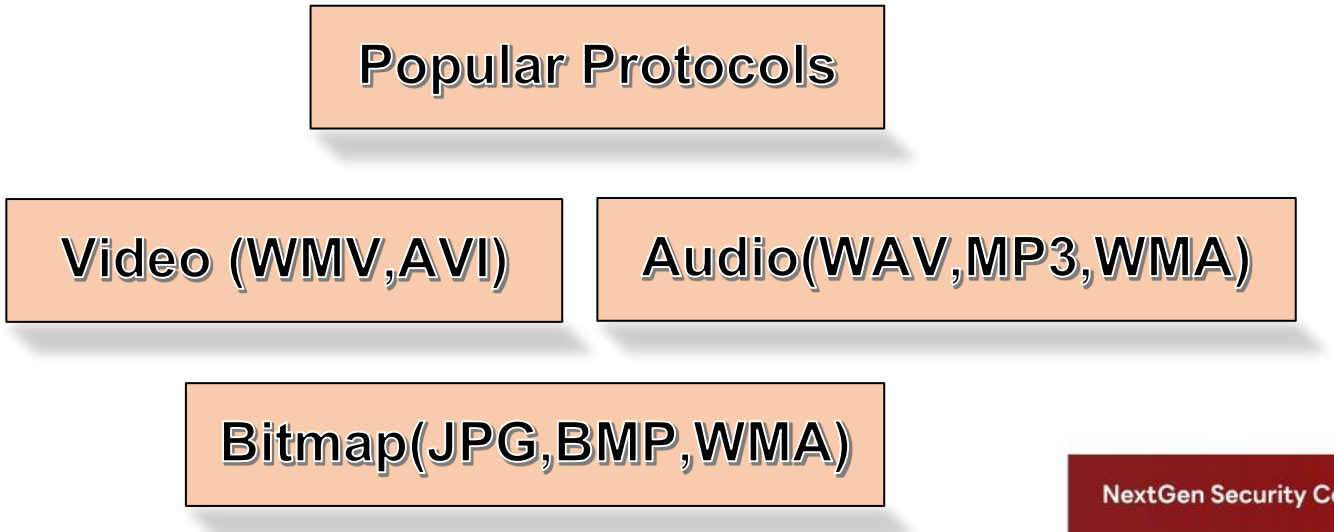


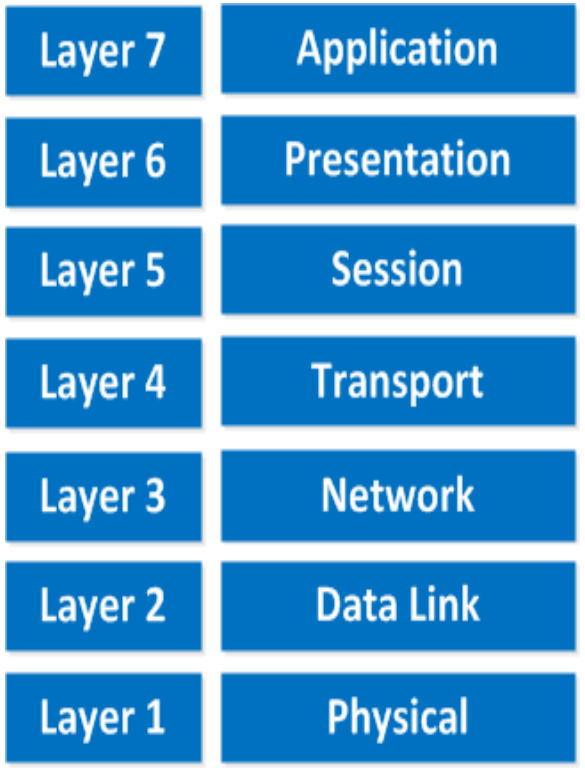
Popular Protocols

NetBIOS | RPC | PPTP | SMPP | SOCKS

OSI Model [7]

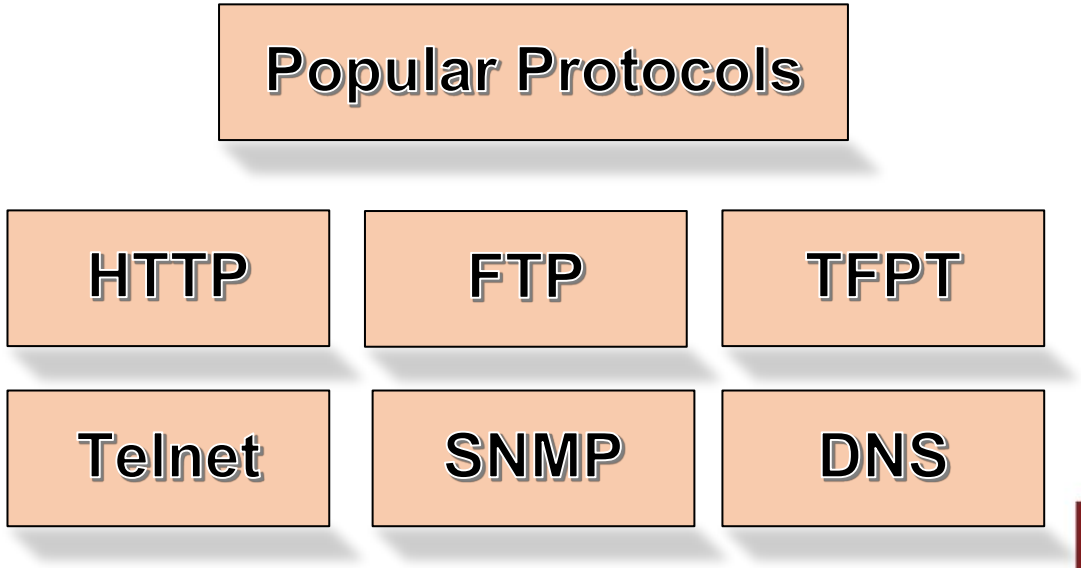
- **Presentation Layer**
- **Purpose**: Data translation, encryption, and compression.
- **Examples**: SSL/TLS, JPEG, MPEG.
- **Functions**: Ensures data is in a usable format and handles encryption/decryption.

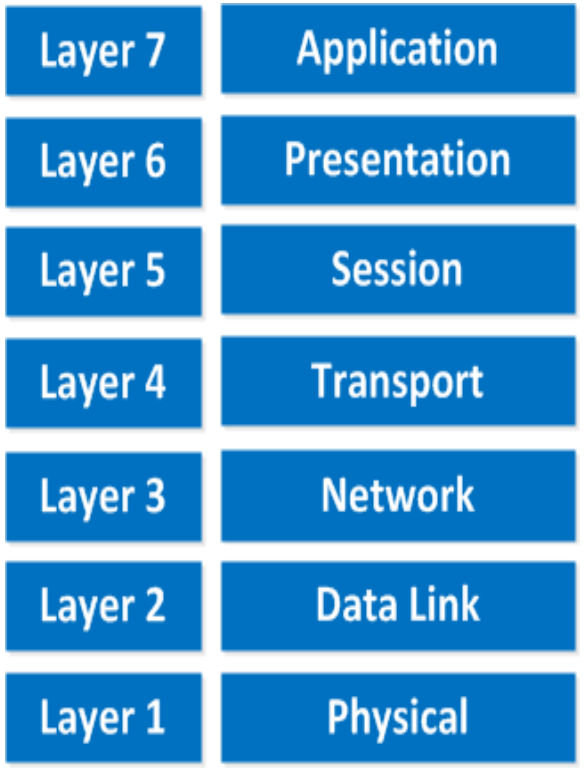




OSI Model [8]

- **Application Layer**
- **Purpose**: Interface between the user and the network.
- **Examples**: HTTP, FTP, SMTP, DNS, Telnet.
- **Functions**: Provides network services to end-user applications.



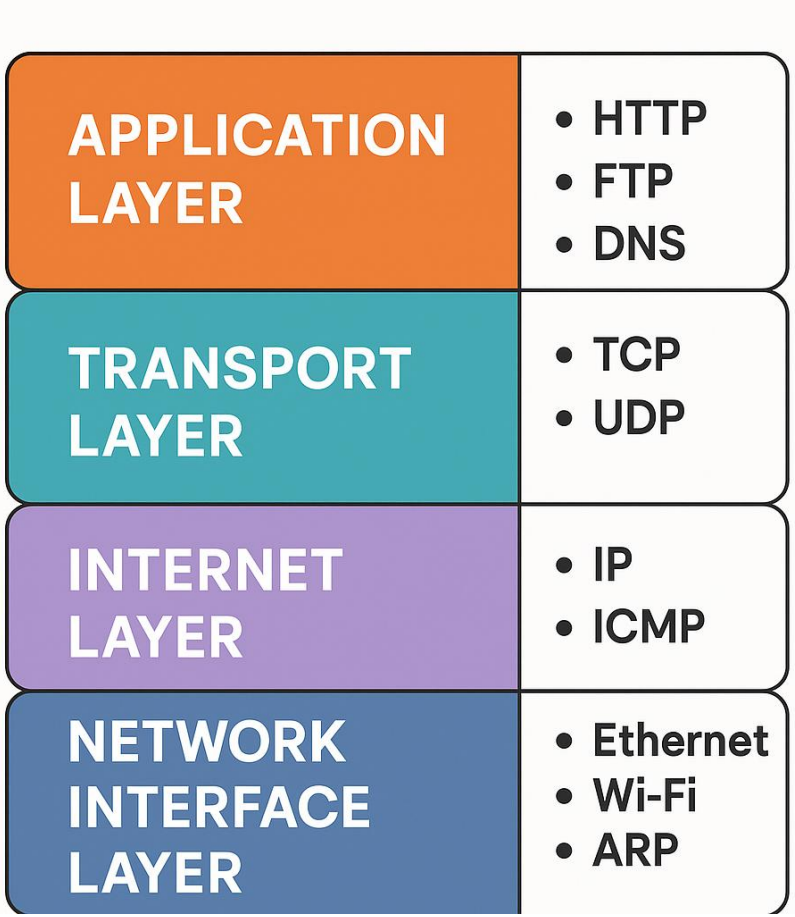


TCP/IP Networking Model

TCP/IP Model

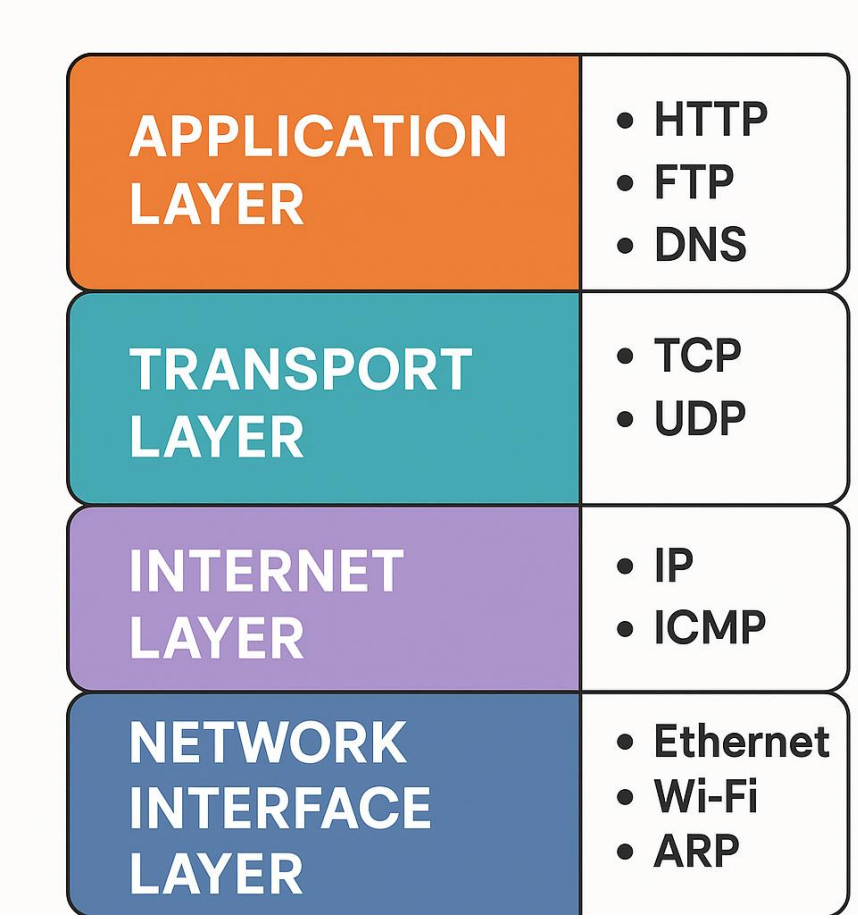
- The **TCP/IP Model** (Transmission Control Protocol/Internet Protocol Model) is a set of communication protocols used for interconnecting network devices on the internet. It defines how data is packaged, transmitted, routed, and received across network boundaries.

- **TCP/IP 4-Layer Model:**
  - Application
  - Transport
  - Internet
  - Network Access



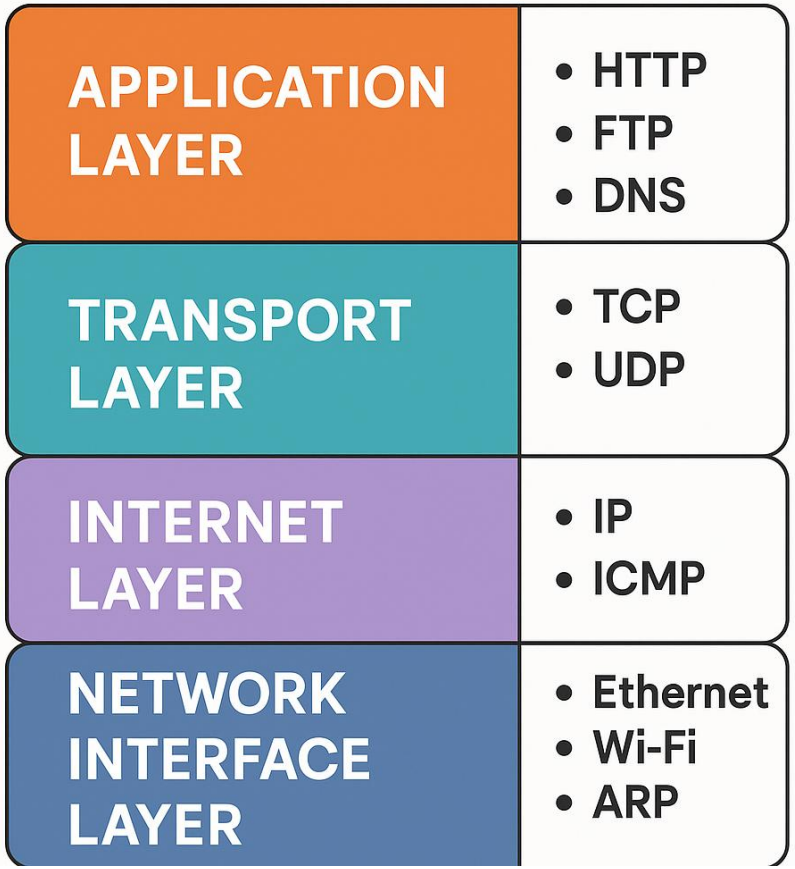
Application Layer

- **Application Layer**
- **Protocols**: HTTP, HTTPS, FTP, SMTP, DNS, SNMP, DHCP, Telnet, SSH
- **Functions**:
  - User interface for communication
  - Provides services like web browsing, email, file transfers
- **Note**: This merges OSI's Application, Presentation, and Session layers.



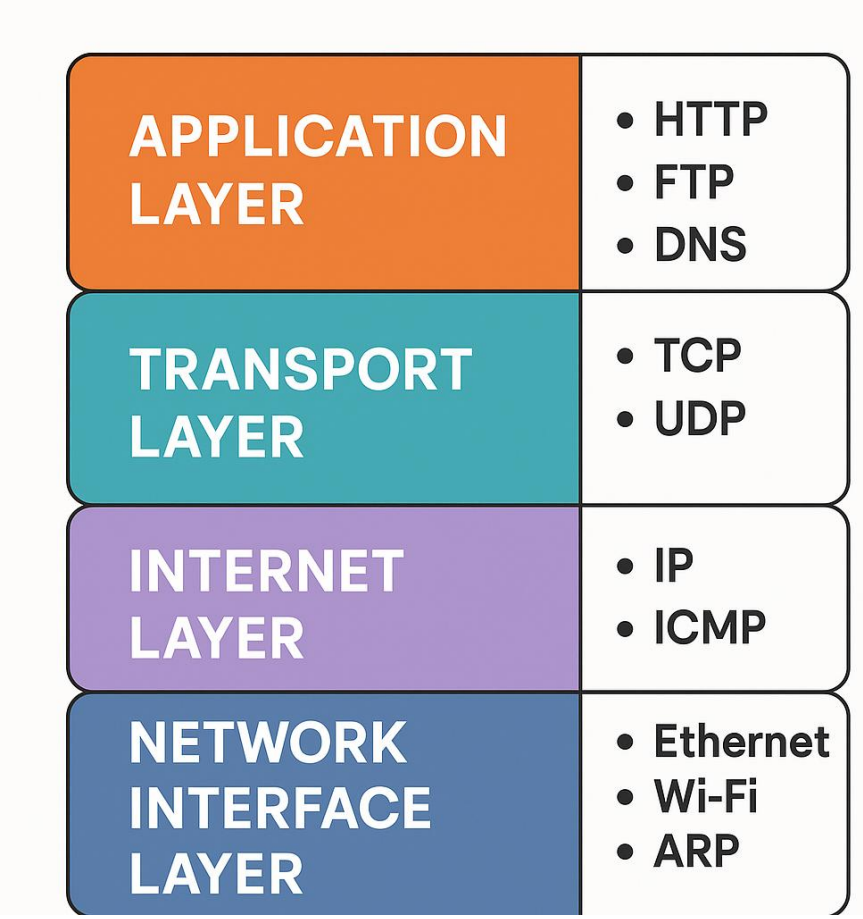
Transport Layer

- ❑ **Transport Layer**
- ❑ **Protocols**: TCP, UDP
- ❑ **Functions**:
  - Reliable communication (TCP)
  - Connectionless communication (UDP)
  - Segmentation, reassembly, and flow control
- **Note**: This merges OSI's Application, Transport layer



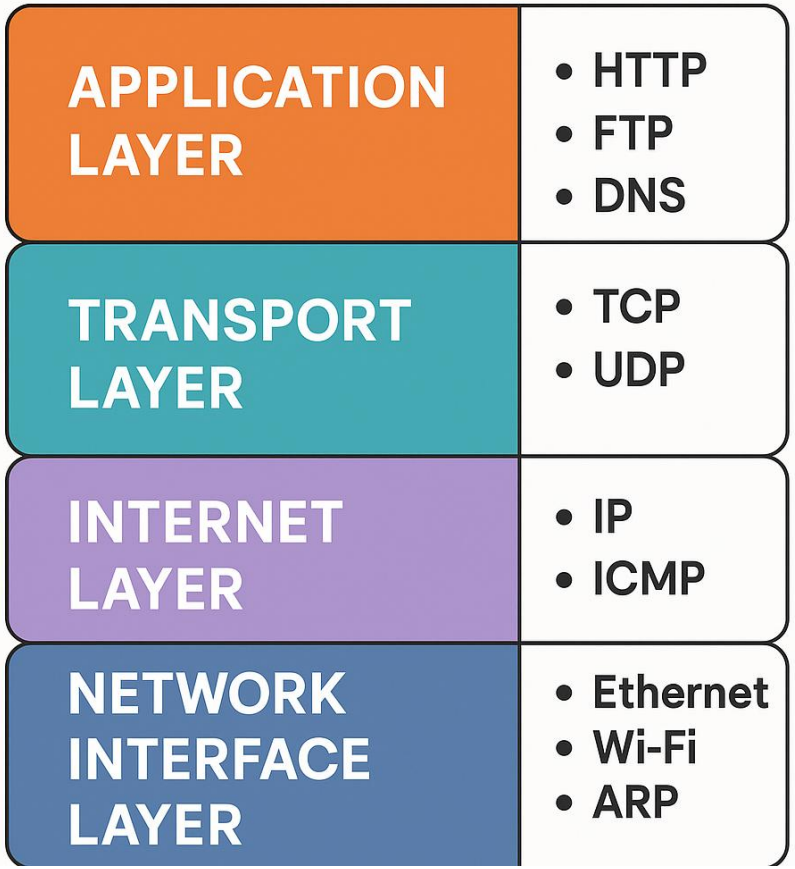
Internet Layer

- **Internet Layer**
- **Protocols**: IP (IPv4/IPv6), ICMP, IGMP, ARP, RARP
- **Functions**:
  - Logical addressing (IP addresses)
  - Packet routing
  - Error reporting and diagnostics (ICMP)
- **Note**: This merges OSI's Application, Network layer

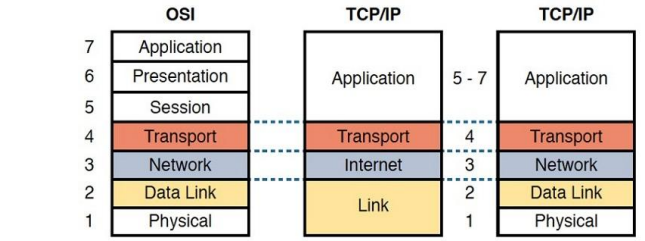


Network Access Layer

- **Network Access Layer**
- **Protocols**: Ethernet, Wi-Fi (802.11), Frame Relay, ATM, ARP
- **Functions**:
  - Data framing
  - MAC addressing
  - Transmission of raw bits over the physical medium (cables, radio)
  - Defines how data is physically sent over the network
- **Note**: This merges OSI's Application, Data Link and Physical layer



Network layers (OSI model recap)

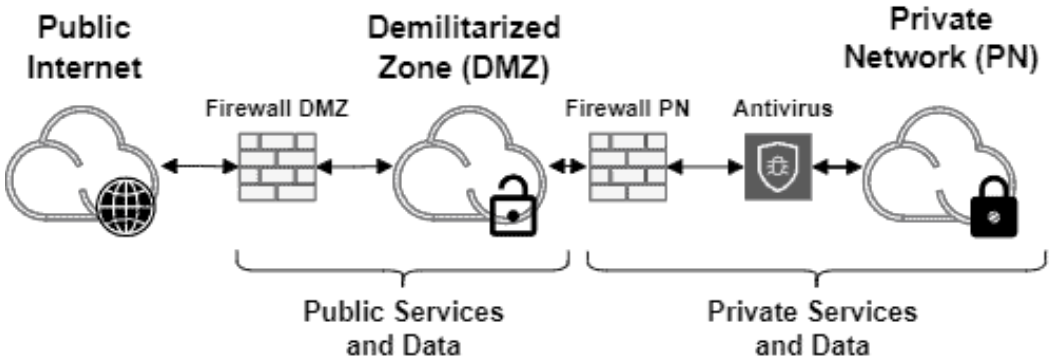


TCP/IP Model (4 Layers)	OSI Model (7 Layers)	Description / Function
Application Layer	**Application, Presentation, Session**	Provides user interaction, data formatting, and session management (e.g., HTTP, FTP, DNS, SMTP).
Transport Layer	**Transport**	Provides end-to-end communication, error handling, and reliability (e.g., TCP, UDP).
Internet Layer	**Network**	Handles logical addressing and routing (e.g., IP, ICMP).
Network Interface Layer	**Data Link, Physical**	Deals with physical transmission of data and hardware addressing (e.g., Ethernet, Wi-Fi).

Secure Network Design

DMZ (Demilitarized Zone) [1]

- A **DMZ** is a **buffer zone** between the public internet and your internal/private network, designed to host **public-facing services** (e.g., web servers, mail servers, DNS) in a controlled environment.
- **Typical Structure:**
  - **Firewall 1**: Between internet and DMZ
  - **Firewall 2**: Between DMZ and internal network
  - **Servers in DMZ**: Web, proxy, SMTP, VPN concentrators



DMZ (Demilitarized Zone) [2]

❑ Benefits:

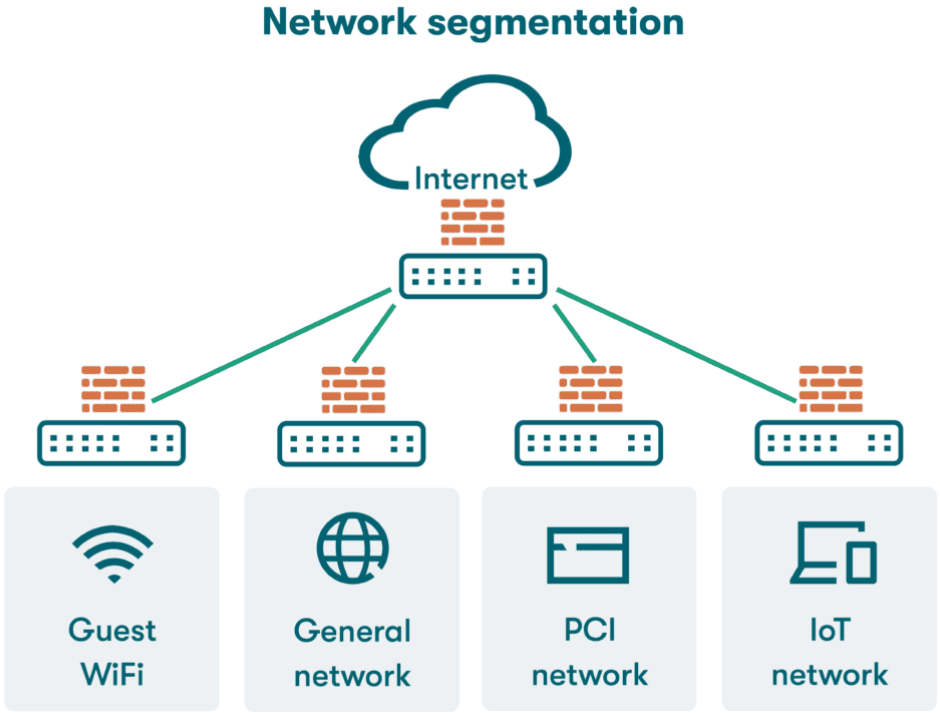
- Limits exposure of internal network
- Controls traffic between external users and internal resources
- Reduces risk if a public server is compromised

❑ Best Practices

- Place only externally accessible services in the DMZ.
- Use separate firewalls (or zones in a next-gen firewall) to control traffic between Internet ↔ DMZ ↔ LAN.
- Apply strict access control and logging.

Network Segmentation [1]

- **Network segmentation** is the practice of dividing a network into **smaller, isolated segments or zones**, each with its own security policies and access controls.
- **Methods:**
  - Physical segmentation (separate switches/cables)
  - Logical segmentation (VLANs, subnets)
  - Segmentation by role (e.g., HR, Finance, Guest, Production)



Network Segmentation [2]

❑ Benefits:

- Limits lateral movement in case of a breach
- Contains malware outbreaks (e.g., ransomware)
- Enforces **least privilege** and access control
- Improves performance through reduced broadcast traffic

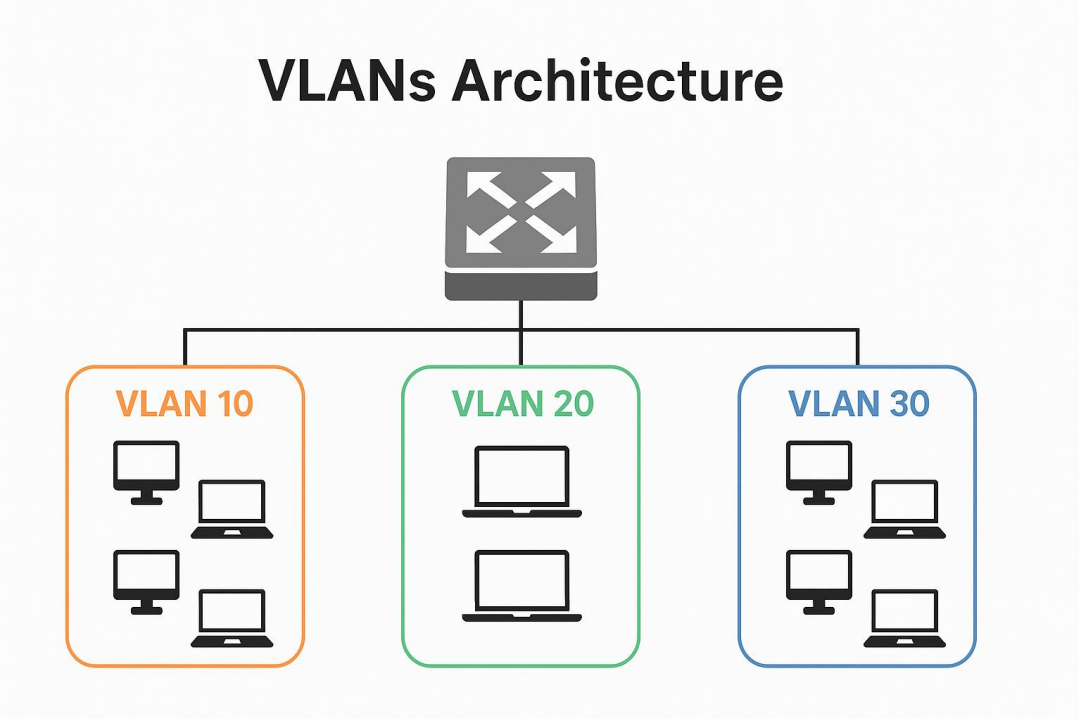
❑ Best Practices:

- Use **firewalls or ACLs** between segments
- Apply **zero trust** principles (don’t assume internal trust)
- **Separate user and server networks**, and guest from corporate
- Monitor and log **cross-segment traffic**

VLANs (Virtual Local Area Networks) [1]

- A **VLAN** allows you to logically separate devices on the same physical switch or network infrastructure into **isolated broadcast domains**, even if they share the same hardware.

- **Uses:**
  - Separate departments (e.g., HR VLAN, IT VLAN)
  - Isolate VoIP traffic
  - Create **guest networks** isolated from internal systems



VLANs (Virtual Local Area Networks) [2]

❑ Benefits:

- Limits broadcast domains
- Enhances segmentation and policy enforcement
- Reduces risk of ARP spoofing and Layer 2 attacks
- Easier policy management and traffic prioritization (QoS)

❑ Best Practices:

- **Use separate VLANs** for users, servers, management, and VoIP
- Implement **router-on-a-stick** or Layer 3 switch for inter-VLAN routing
- Use **802.1Q trunking** securely and disable unused ports
- Combine VLANs with **ACLs and firewalls** for strong access control

VLANs (Virtual Local Area Networks) [3]

VLAN ID	Department	Subnet	Access
10	Finance	192.168.10.0/24	Internal servers only
20	HR	192.168.20.0/24	Limited internal resources
30	Guest Wi-Fi	192.168.30.0/24	Internet only
40	DMZ Servers	10.0.0.0/24	Externally accessible services

Threat Surface Mapped to OSI Model

Threat Surface Mapped to OSI Model

OSI Layer	Function	Key Threats	Mitigation Techniques
7 – Application	Interface for user apps	- Web app vulnerabilities (XSS, SQLi, CSRF)- API abuse- Malware- Phishing- Credential stuffing	- Web Application Firewall (WAF)- Input validation & sanitization- Secure coding practices- API authentication/throttling- MFA, user training
6 – Presentation	Data format & encryption	- Weak encryption- Improper SSL/TLS implementation- Data leakage in transit	- Enforce TLS 1.3- Strong encryption (AES-256)- Disable deprecated protocols (SSL, TLS 1.0/1.1)- Certificate pinning
5 – Session	Session management	- Session hijacking- Man-in-the-middle (MitM) attacks- Cookie theft	- Use secure, encrypted sessions (TLS)- HttpOnly and Secure cookie flags- Session timeouts- Token rotation (e.g., OAuth2)

Threat Surface Mapped to OSI Model

OSI Layer	Function	Key Threats	Mitigation Techniques
4 – Transport	End-to-end communication	- TCP/UDP flooding (DDoS)- Port scanning- SYN/ACK attacks	- Rate limiting & DDoS protection (e.g., Cloudflare, AWS Shield)- Stateful firewalls- TCP hardening
3 – Network	Routing and addressing	- IP spoofing- Routing attacks (BGP hijack)- DoS/DDoS at network level	- IPsec for secure tunnels- ACLs (Access Control Lists)- Anti-spoofing rules- Use of secure routing protocols
2 – Data Link	MAC & switching	- MAC flooding- ARP spoofing- VLAN hopping	- Port security on switches- Dynamic ARP inspection- VLAN segmentation- Private VLANs
1 – Physical	Hardware transmission	- Physical tampering- Cable tapping- Device theft- EMI (Electromagnetic Interference)	- Physical access controls (locked racks, surveillance)- Port blocking- Tamper-evident seals- Air-gapped critical systems

Firewalls, IDS/IPS, and Access Controls

Agenda

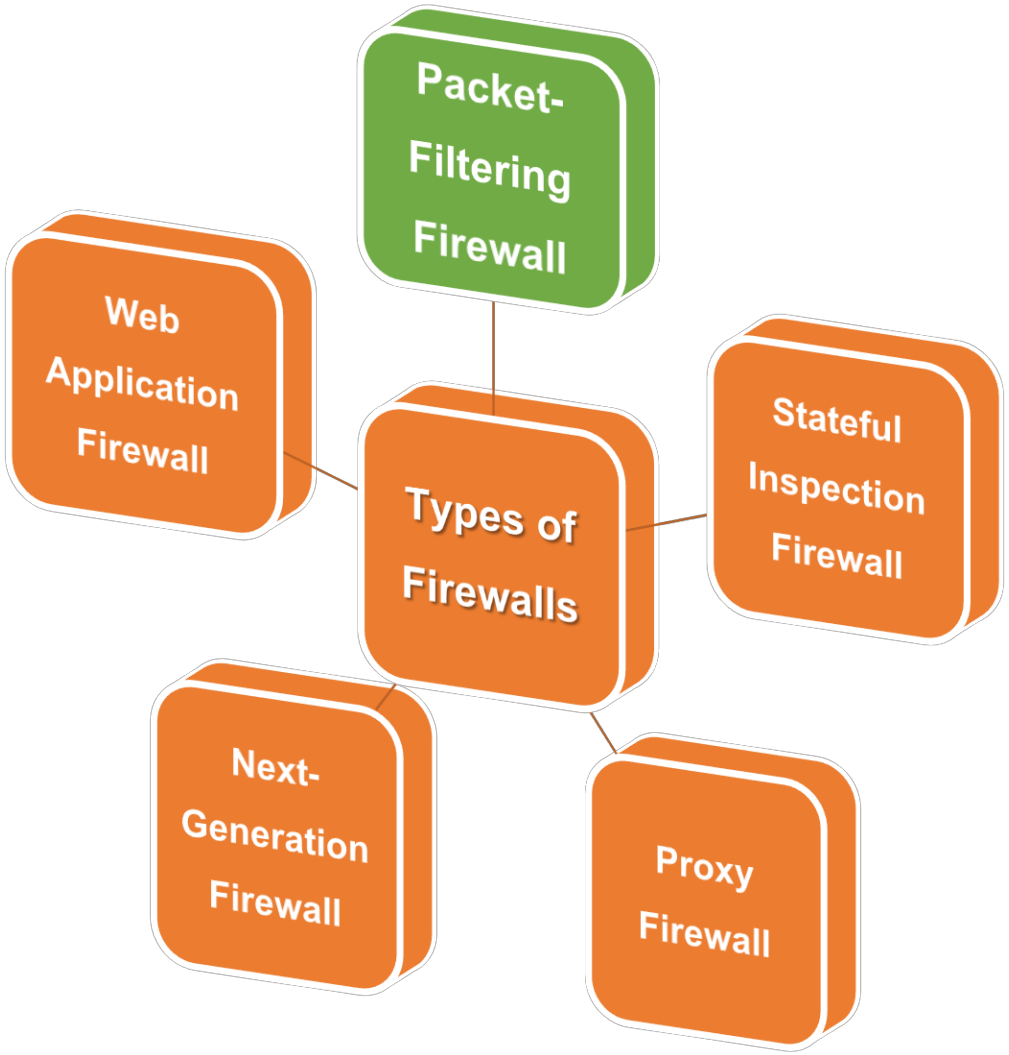
- Firewalls, IDS/IPS, and Access Controls
  - Types of firewalls (packet filtering, stateful, proxy, NGFW)
  - Intrusion Detection vs. Intrusion Prevention Systems
  - Role-based access control (RBAC), ACLs, Zero Trust principles

Firewalls

Firewalls [1]

- A **firewall** is a network security device (hardware, software, or cloud-based) that monitors and controls incoming and outgoing network traffic based on predetermined security rules.
- **Purpose of a Firewall:**
  - Prevent unauthorized access to or from a private network.
  - Enforce security policies by controlling traffic flows.
  - Inspect data packets and decide whether to allow or block them based on rules.
  - Protect internal network resources from external threats and attacks.

Firewalls [2]



❑Packet-Filtering Firewall (Stateless)

❑This is the most basic type of firewall that inspects **individual packets** in isolation based on:

- Source IP address
- Destination IP address
- Port numbers
- Protocol type

❑Operation:

- No awareness of connection state
- Simple allow/deny based on rules (ACLs)

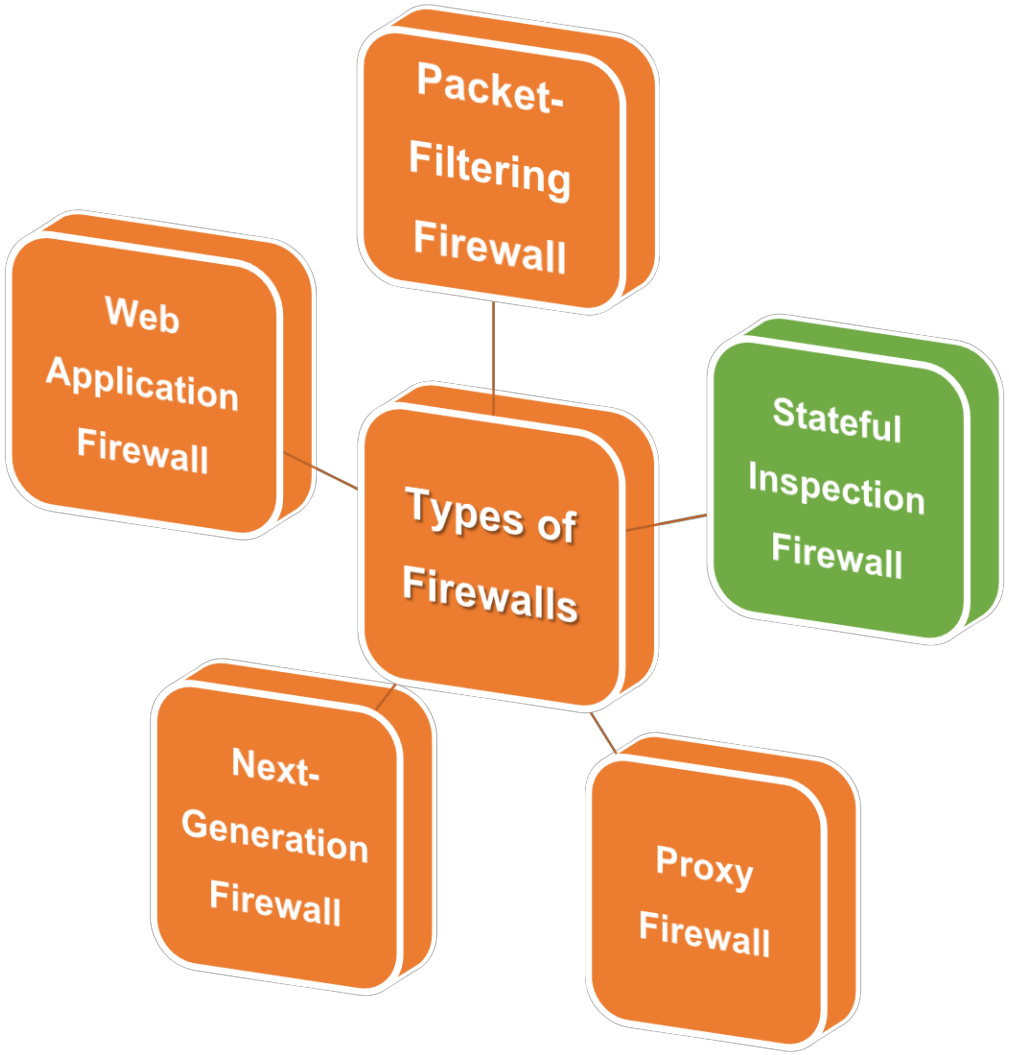
❑Pros:

- Fast and lightweight
- Easy to configure

❑Cons:

- No context of ongoing connections
- Vulnerable to spoofing and state-based attacks

Firewalls [3]



❑Stateful Inspection Firewall

❑Also known as a **dynamic packet-filtering firewall**, it tracks the **state of active connections** and makes decisions based on the **state of the traffic** (e.g., TCP handshakes).

❑Operation:

- Maintains a state table for established sessions
- Blocks unsolicited packets not matching known sessions

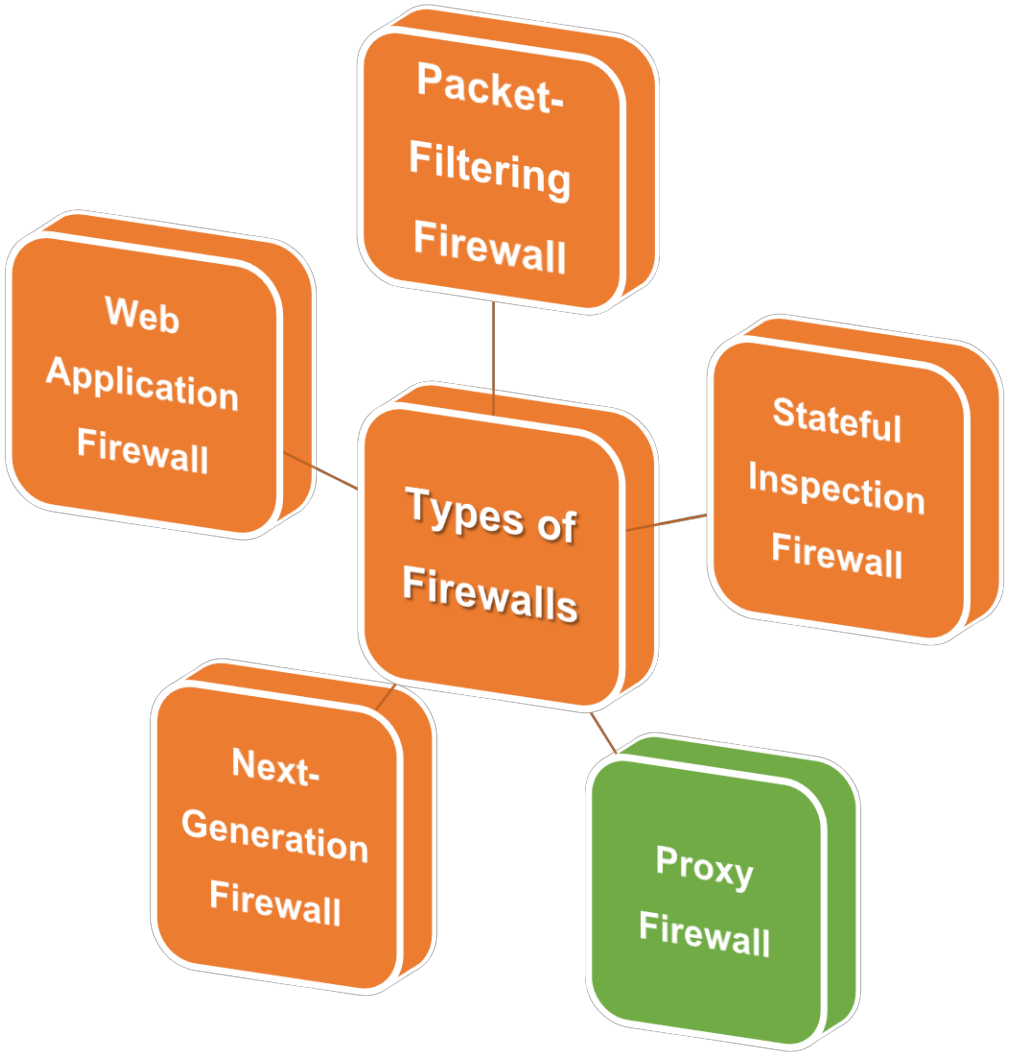
❑Pros:

- More secure than stateless
- Recognizes connection initiation, established sessions

❑Cons:

- Slightly more resource-intensive
- May be bypassed by encrypted malicious traffic

Firewalls [4]



❑Proxy Firewall (Application-Level Gateway)

❑Acts as an **intermediary** between end users and the services they access. Instead of forwarding packets directly, it **analyzes entire application-layer payloads**.

❑Operation:

- Accepts client request
- Forwards to destination on client's behalf
- Inspects traffic at Layer 7

❑Pros:

- Deep inspection at application level
- Hides internal network structure

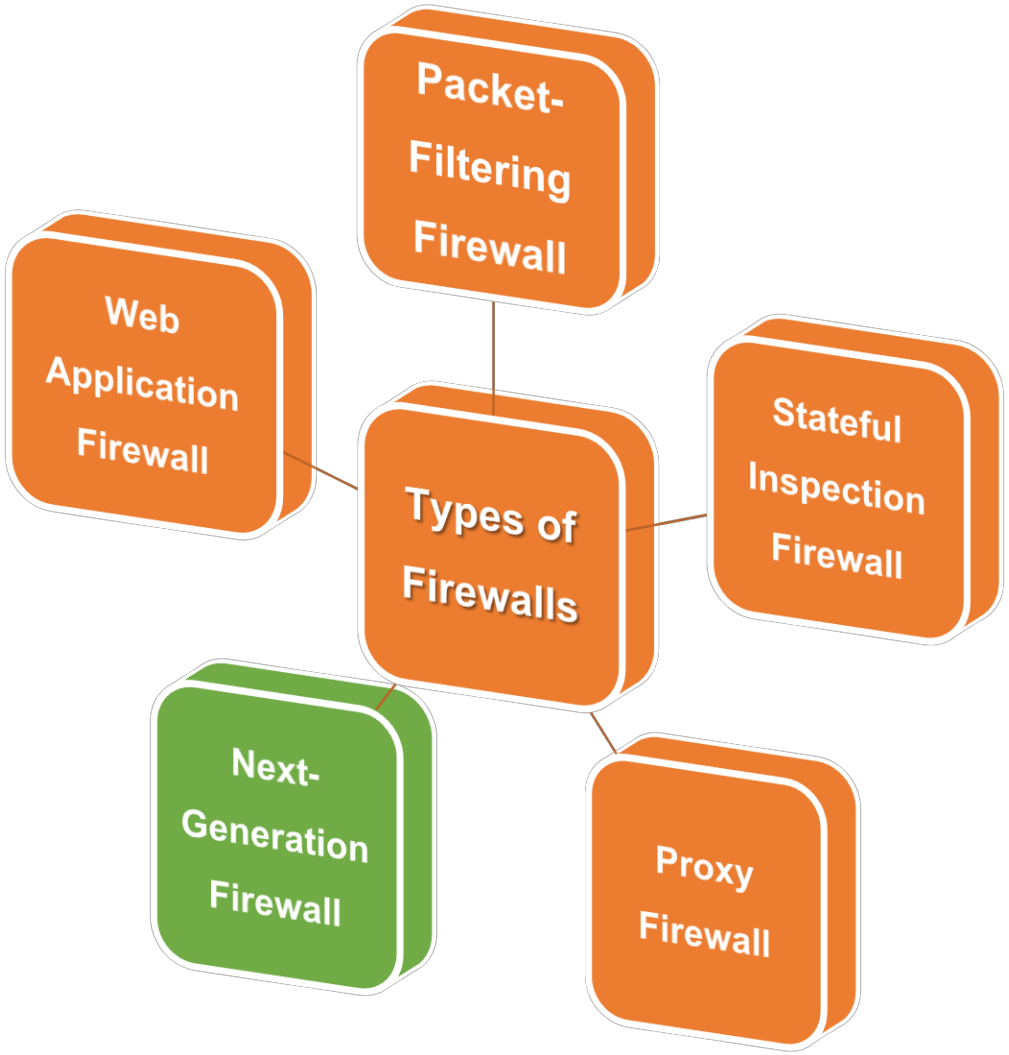
❑Cons:

- Higher latency
- Needs specific proxy for each protocol (HTTP, FTP, etc.)

❑Use Case:

- Secure web gateways
- Isolating users from the internet

Firewalls [5]



❑Next-Generation Firewall (NGFW)

❑A **comprehensive firewall** that combines **traditional firewall features** with advanced security features like:

- Application awareness
- Integrated Intrusion Prevention System (IPS)
- Deep packet inspection
- User identity integration (Active Directory, SSO)
- SSL/TLS inspection

❑Pros:

- Context-aware filtering
- Blocks modern threats (malware, botnets)
- Application-level visibility

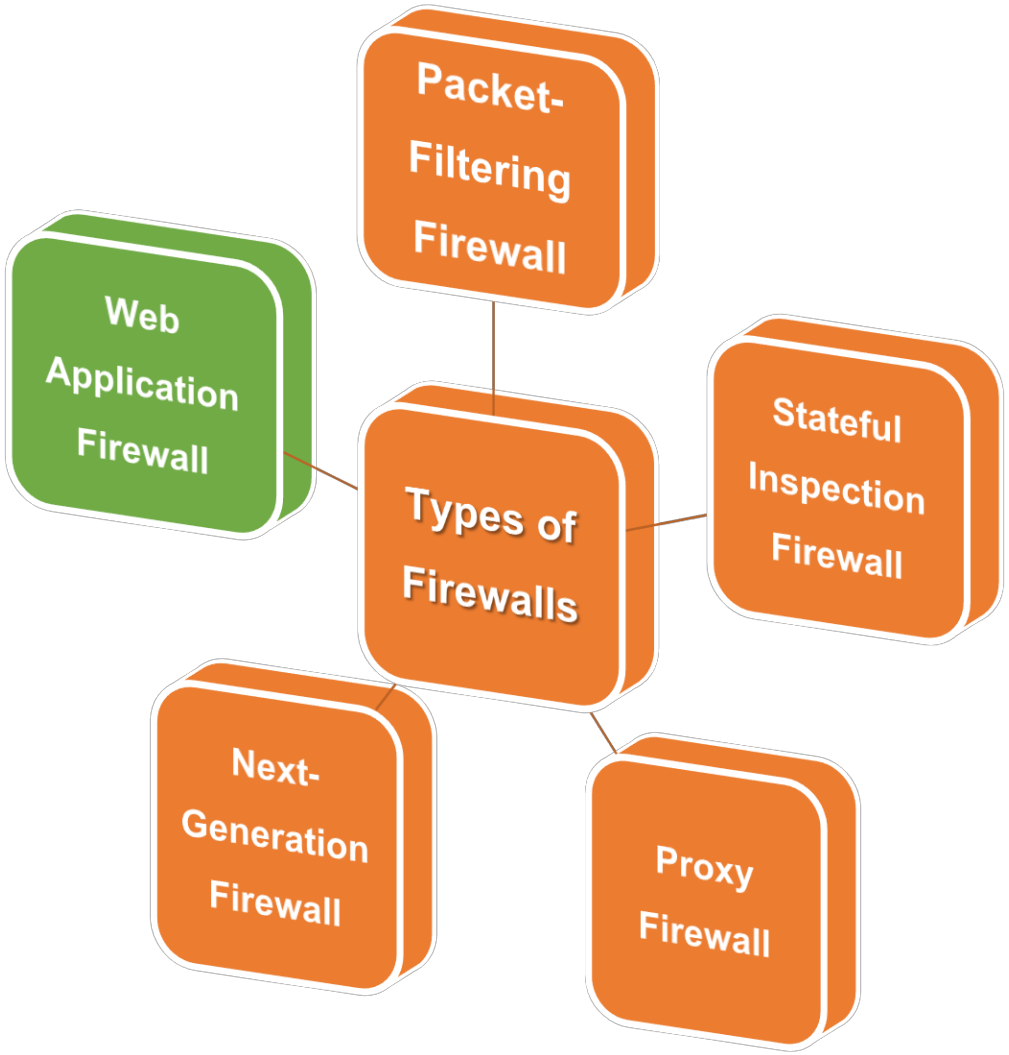
❑Cons:

- Higher cost
- May require frequent updates and tuning

❑Examples:

- Palo Alto Networks NGFW
- Cisco Firepower
- FortiGate

Firewalls [6]



❑Web Application Firewall (WAF)

❑A **specialized firewall** that **protects web applications** by monitoring and filtering HTTP/HTTPS traffic between a web client and a web server.

❑Operation:

- Filters based on app-layer attacks (e.g., SQL injection, XSS)
- Often deployed inline in front of web servers

❑Pros:

- Protects against OWASP Top 10 threats
- Can enforce input validation, rate limiting

❑Cons:

- False positives if misconfigured
- Doesn’t inspect non-web traffic

❑Examples:

- AWS WAF
- Cloudflare WAF
- F5 BIG-IP ASM

Summary

Firewall Type	Layer	Stateful?	Application-Aware	Use Case
Packet-Filtering	L3/L4	❌	❌	Basic perimeter filtering
Stateful Inspection	L3/L4	✅	❌	Enterprise network edge
Proxy Firewall	L7	✅	✅	App-specific filtering
Next-Gen Firewall (NGFW)	L3–L7	✅	✅	Comprehensive threat prevention
Web App Firewall (WAF)	L7	✅	✅ (HTTP/HTTPS only)	Web app protection

IDS/IPS

IDS (Intrusion Detection Systems)

- An **IDS** is a **passive monitoring system** that **detects** unauthorized, suspicious, or malicious activity within a network or host and **generates alerts**. It does **not block or prevent** the detected activity.
- **Purpose:**
  - Monitor network or host traffic for **known attack patterns or anomalies**
  - **Alert** administrators or log incidents for further investigation
  - Used for **forensic analysis**, **compliance**, and **security visibility**

IPS (Intrusion Prevention Systems)

- An **IPS** is an **active, inline security device** that not only detects threats but also **blocks, drops, or prevents** malicious activity in real-time.
- **Purpose:**
  - **Prevent** attacks before they impact systems
  - Block known threats, unusual patterns, and policy violations
  - Often integrated with **firewalls** for layered security

IDS vs. IPS

Feature	IDS	IPS
Action	Detects and alerts	Detects and blocks
Deployment	Out-of-band (SPAN port, TAP)	Inline (in path of traffic)
Traffic Latency	No impact	May introduce slight delay
Risk of False Positives	Low risk (no blocking)	Higher risk (may block legitimate traffic)
Use Case	Visibility, threat hunting, compliance	Real-time protection, threat mitigation

Types of IDS (Intrusion Detection Systems)

Type	Description	Use Case
NIDS (Network-based IDS)	Monitors **network traffic** at strategic points (e.g., between VLANs, DMZ, or perimeter). Detects attacks like port scanning, DDoS, malware traffic.	Detect threats across the entire network.
HIDS (Host-based IDS)	Installed directly on an **individual host/server**. Monitors system logs, file integrity, registry, and local processes.	Detect insider threats, rootkits, or unauthorized file changes.
WIDS (Wireless IDS)	Monitors **wireless traffic**. Detects rogue access points, unauthorized devices, and Wi-Fi attacks like deauthentication or spoofing.	Secure Wi-Fi networks (e.g., in hospitals, enterprises, campuses).
Hybrid IDS	Combines NIDS and HIDS for broader detection and correlation across endpoints and networks.	Unified visibility and more accurate alerts.

Based on Detection Technique

Type	Description	Example Detection
Signature-Based IDS	Compares traffic patterns to a database of known **attack signatures**.	Detects known malware, exploits, command injection.
Anomaly-Based IDS	Builds a baseline of **normal behavior**, then alerts on any **deviation**.	Detects zero-day attacks, insider threats, traffic spikes.
Heuristic-Based IDS	Uses rules, algorithms, and behavioral models to identify suspicious patterns, often using AI or Machine Learning.	Detects sophisticated or evolving threats.
Policy-Based IDS	Triggers alerts when **security policies** (e.g., usage rules, file access restrictions) are violated.	Unauthorized login attempts or protocol usage.

Summary

IDS Type	Layer	Detection Method	Pros	Cons
NIDS	Network	Signature/Anomaly	Broad visibility	Encrypted traffic blind spots
HIDS	Host	File/Log/Behavior	Deep host insight	Limited to single device
WIDS	Wireless	Signature/Anomaly	Wi-Fi attack detection	Physical placement required
Signature-Based	Any	Known attack matching	Low false positives	Can’t detect unknown threats
Anomaly-Based	Any	Behavior deviation	Detects zero-day	High false positives if not tuned
Hybrid IDS	Network + Host	Mixed	High accuracy	Complex to manage and correlate

Types of IPS (Intrusion Prevention Systems)

Type	Description	Use Case
Network-based IPS (NIPS)	Monitors and prevents malicious traffic **on the network**, usually placed inline at network chokepoints (e.g., between LAN and WAN).	Perimeter defense, data center protection
Wireless IPS (WIPS)	Monitors wireless networks for rogue access points, unauthorized devices, or wireless-specific attacks (e.g., deauthentication, fake APs).	Securing Wi-Fi environments (e.g., campuses, hospitals)
Host-based IPS (HIPS)	Installed on **individual devices** (servers, workstations) to monitor and block suspicious system calls, file modifications, or local network activity.	Server protection, endpoint hardening
Network Behavior Analysis (NBA) IPS	Uses flow data and traffic patterns to detect anomalies (e.g., DDoS, port scans, policy violations).	Detecting zero-day threats and traffic anomalies

Based on Detection Technique

Type	Description	Example
Signature-Based IPS	Detects known threats using a database of predefined patterns (signatures).	Detecting specific malware or known exploits (e.g., MS17-010)
Anomaly-Based IPS	Detects deviations from established baselines (e.g., unusual traffic volume, protocol behavior).	Detecting zero-day attacks or insider threats
Policy-Based IPS	Uses predefined rules and security policies to determine if traffic is allowed.	Blocking applications or protocols not permitted (e.g., peer-to-peer, FTP)
Hybrid IPS	Combines **signature**, **anomaly**, and **policy-based** detection for better accuracy and coverage.	Most modern commercial IPS systems (e.g., Cisco Firepower, Palo Alto NGFW)

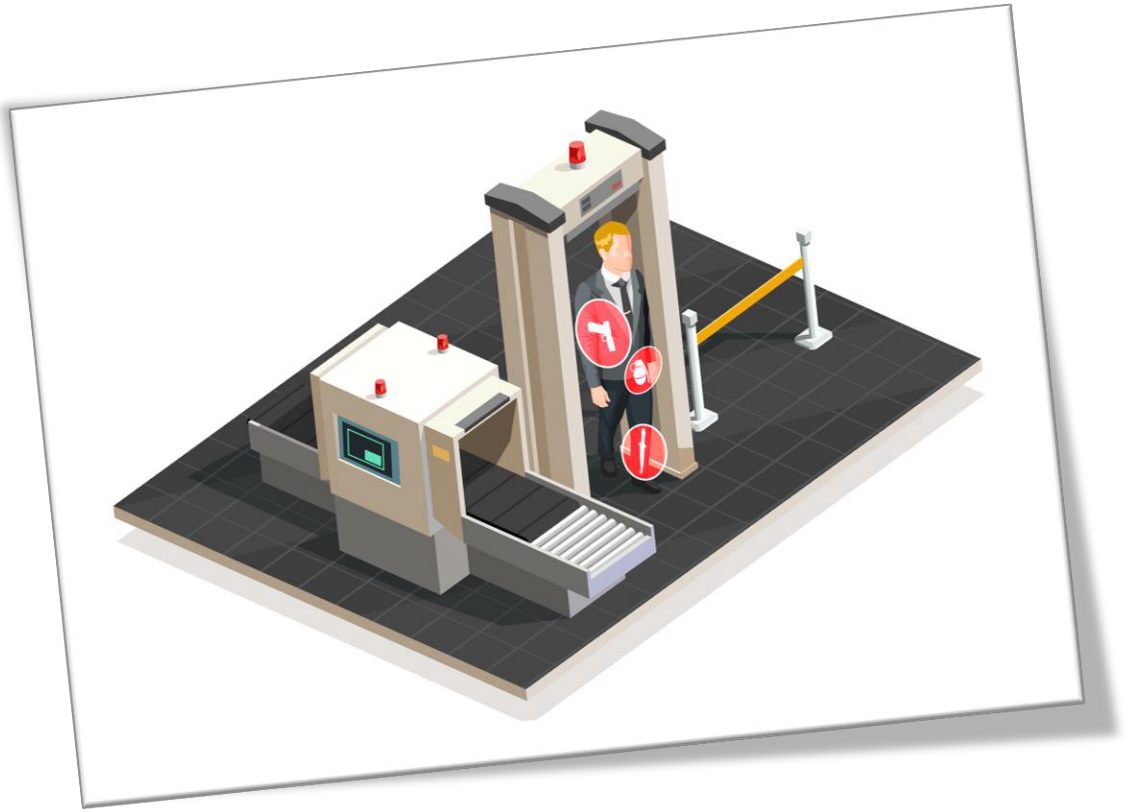
Based on Integration Type

Type	Description	Examples
Standalone IPS	Dedicated appliance for intrusion prevention	Trend Micro Tipping Point, Cisco IPS 4300
Integrated with NGFW (Next-Gen Firewall)	IPS capabilities built into next-gen firewalls	Cisco Firepower, Palo Alto NGFW, FortiGate
Cloud-based IPS	Delivered as part of cloud security services; monitors cloud-based workloads	AWS Guard Duty, Azure Defender for Cloud

Access Controls

Access Controls [1]

- **Access Controls**
- Access control is a **security technique** that regulates who or what can view or use resources in a computing environment. It enforces rules about **who is allowed to access what**, under what **conditions**, and with what **permissions**.



Access Controls [2]

❑Purpose:

- **Protect sensitive data** from unauthorized access, modification, or destruction.
- **Limit insider threats** by controlling user privileges.
- **Support compliance** with data protection laws (e.g., GDPR, HIPAA, PCI-DSS).
- **Enforce Principle of Least Privilege (PoLP)** to minimize exposure.

Types of Access Control [1]

- **Physical Access Control**
- Physical access control restricts access to physical locations (buildings, data centers, server rooms, etc.) using hardware-based mechanisms.
- **Purpose**
  - Prevent unauthorized individuals from **physically accessing** sensitive areas.
  - Protect **hardware**, **data**, and **personnel** from theft, damage, or tampering.
  - Mitigate risks such as **social engineering**, **tailgating**, and **physical attacks**.

Types of Access Control [2]

❑ Pros

- Provides **tangible barriers** against intruders.
- Often acts as a **first line of defense** before technical controls.
- Can integrate with security systems (e.g., surveillance, alarms).

❑ Cons

- **Bypassable** through human error (e.g., tailgating).
- Expensive to implement at scale (hardware, maintenance, monitoring).
- Doesn’t protect against **digital/remote threats**.

Examples

Method	Description
Key cards / RFID badges	Grant access to authorized users only.
Biometric scanners	Fingerprint or facial recognition at entrances.
Security guards	Manual verification of IDs and access rights.
Smart locks	Controlled through mobile apps or access systems.
Turnstiles or mantraps	Restrict entry to one person at a time.

Types of Access Control [3]

- **Logical (Technical) Access Control**
- Logical access control restricts access to **computer systems**, **data**, and **networks** using **software-based mechanisms** such as usernames, passwords, permissions, and encryption.
- **Purpose**
  - Prevent unauthorized **digital access** to systems, files, networks, or applications.
  - Enforce **authentication**, **authorization**, and **accountability**.
  - Protect **confidentiality**, **integrity**, and **availability** of data.

Types of Access Control [4]

❑Pros

- Enables **fine-grained control** over who can access what.
- Scalable across large numbers of users and systems.
- Easier to **audit and log** digital access events.

❑Cons

- Vulnerable to **cyberattacks** like phishing, brute force, or privilege escalation.
- Misconfiguration can expose sensitive data.
- Needs **regular updates** and **monitoring** to remain effective.

Example

Method	Description
Usernames and Passwords	Basic identity verification.
Multi-Factor Authentication (MFA)	Adds extra layer (e.g., token, app, biometric).
Access Control Lists (ACLs)	Specify what users or systems can access what resources.
Role-Based Access Control (RBAC)	Permissions based on job roles.
Group Policies (GPOs)	Control user rights and system behavior in Windows domains.
VPN Access Control	Only authorized devices/users can access internal networks remotely.

Summary

Feature	Physical Access Control	Logical Access Control
Definition	Controls access to physical spaces	Controls access to digital resources
Purpose	Prevent physical intrusion	Prevent unauthorized digital access
Pros	Strong visible deterrent; protects physical assets	Scalable, auditable, integrates with IAM systems
Cons	Costly, vulnerable to tailgating	Requires regular monitoring, vulnerable to cyberattacks
Example	RFID badge, biometric door lock	Passwords, MFA, file permissions

Access Control Models [1]

❑DAC (Discretionary Access Control)

❑Definition:

- Access is controlled by the **owner** of the resource, who decides who can access it and what they can do.

❑Purpose:

- Provide **flexibility** in managing access to files or systems at the discretion of the data owner.

❑Pros:

- Simple and user-friendly.
- Easy to manage on a small scale.

❑Cons:

- Vulnerable to insider threats and malware.
- Difficult to enforce organization-wide policy.

❑Example:

- A Windows user sets read/write permissions for a document and grants access to other users.

Access Control Models [2]

❑MAC – Mandatory Access Control

❑Definition:

- Access is controlled by a **central authority** based on predefined security labels (e.g., classified, secret, top secret).

❑Purpose:

- Enforce strict, non-discretionary security policies in **high-security environments**.

❑Pros:

- **Highly secure** and rigid
- Ideal for **classified information** and environments with high confidentiality needs

❑Cons:

- Very **inflexible**
- Difficult and expensive to manage
- Can **slow down operations** due to bureaucracy

❑Example:

- A government system where only users with “Top Secret” clearance can access classified documents.

Access Control Models [3]

❑RBAC (Role-Based Access Control)

❑Definition:

- Access is granted based on a user's **role** within an organization (e.g., Admin, HR, Finance).

❑Purpose:

- Simplify access control by grouping permissions into **roles** aligned with job functions.

❑Pros:

- **Scalable** and easy to manage
- Aligns well with **organizational structures**
- Reduces risk of **privilege creep**

❑Cons:

- May require complex role definitions
- Changes in user roles require careful updates
- **Too many roles** can complicate administration

❑Example:

- A finance department role grants access to payroll and accounting systems, but not HR files.

Access Control Models [4]

❑ABAC (Attribute-Based Access Control)

❑Definition:

- Access is based on a combination of **attributes** — user, resource, environment (e.g., user role, time of day, location).

❑Purpose:

- Provide **fine-grained, dynamic control** over access decisions.

❑Pros:

- Highly **dynamic and flexible**
- Enforces policies based on real-world conditions (e.g., time, location)
- Supports **zero trust** architecture

❑Cons:

- **Complex to implement and maintain**
- Requires a strong attribute management system
- **Harder to audit** due to dynamic nature

❑Example:

- A user from the HR department can access employee records **only during business hours and from a corporate device**.

Access Control Models [5]

❑Rule-Based Access Control

❑Definition:

- Access is granted or denied based on **predefined rules** set by administrators — often used for **system-wide policies**.

❑Purpose:

- Enforce **environmental** or **contextual** rules regardless of user identity.

❑Pros:

- Automates access control using logic.
- Useful for **firewalls**, **routing policies**, or **network access control**.

❑Cons:

- Lacks user-specific control or flexibility.
- Harder to audit at the user level.

❑Example:

- A firewall rule allows database access **only** during business hours, or a rule denies access from non-corporate IP addresses.

Summary

Model	Definition	Best Use Case	Pros	Cons	Example
DAC	Resource owner controls access	Small teams, personal systems	Easy to manage	Weak security	User sets file permissions
MAC	Admin-enforced with classification	Government, military	High security	Inflexible	SELinux labels
RBAC	Based on job roles	Enterprise environments	Scalable	Role explosion	AD groups and GPOs
ABAC	Based on attributes (user/resource/context)	Cloud, zero-trust	Very granular	Complex	Access if user=HR AND time=9–5
Rule-Based	Based on system rules	Firewalls, routers	Automated, policy-based	Lacks user context	Allow HTTP 80 only from LAN

Network Protocols & Encryption

Agenda

❑ Network Protocols & Encryption

- Secure protocols: HTTPS, SSH, SFTP, IPSec, SSL/TLS
- Insecure protocols: Telnet, FTP, HTTP – and how to replace them
- VPNs and tunneling protocols
- Encryption basics (symmetric/asymmetric, key management)

Network Protocols

❑Network Protocols

❑Network protocols are **standardized rules** and **formats** that govern how data is transmitted, routed, received, and interpreted across a network. These protocols enable **interoperability** between different devices and systems.

❑Purpose

- Ensure **reliable and secure communication** between devices.
- Define **how data packets are formatted, transmitted, and handled**.
- Facilitate **error checking**, **addressing**, **session control**, and **data flow**.
- Enable communication over **LANs**, **WANs**, and the **internet**.

Common Network Protocols

Protocol	Layer	Purpose	Secure Version
HTTP	Application	Web browsing	**HTTPS** (via TLS)
FTP	Application	File transfer	**FTPS/SFTP**
SMTP, POP3, IMAP	Application	Email transmission and access	Via **STARTTLS** or SSL/TLS
DNS	Application	Domain name resolution	**DNSSEC**, **DoH**
SNMP	Application	Network monitoring	SNMPv3 is secure
IP (IPv4/IPv6)	Network	Addressing and routing	**IPsec** for encryption

Secure protocols

HTTPS [1]

- **HTTPS** is the **secure version of HTTP** (HyperText Transfer Protocol). It uses **TLS (Transport Layer Security)** to **encrypt** data transmitted between a client (typically a web browser) and a web server.
  - HTTPS = HTTP + TLS (formerly SSL)
  - Operates on **port 443**
  - Uses **public key infrastructure (PKI)** for authentication and encryption
- **Purpose**
- Ensure **secure communication** over the web by:
  - **Encrypting** data in transit (confidentiality)
  - **Authenticating** the server via digital certificates
  - **Preventing tampering** (integrity)

การเปลี่ยนแปลง แก้ไข

- Protect users from **man-in-the-middle (MITM)** attacks, **eavesdropping**, and **data theft**

HTTPS [2]

❑ Pros

Benefit	Description
Data Encryption	Protects sensitive information (e.g., passwords, credit cards) during transmission
Authentication	Validates the identity of the website via SSL/TLS certificates
Data Integrity	Ensures data isn't modified in transit
Improves SEO	Google ranks HTTPS-enabled sites higher
Trust Indicator	Browsers show padlock icon for HTTPS, building user trust
Secure APIs & Mobile Apps	Essential for encrypting REST API or mobile app traffic

HTTPS [3]

❑ Cons

Limitation	Description
Complex Setup	Requires purchasing/installing/renewing SSL/TLS certificates (although free options like Let's Encrypt exist)
Performance Overhead	Slightly higher CPU usage due to encryption (mostly negligible with modern hardware and TLS 1.3)
False Sense of Security	HTTPS only secures data in transit — not the server or the data after it arrives
Certificate Management	Expired or misconfigured certificates can cause trust issues or outages

Summary

Feature	HTTPS
Protocol	HTTP secured with TLS encryption
Port	443 (vs. port 80 for HTTP)
Secure?	Yes – encrypts data in transit
Used for	Secure web browsing, online transactions
Example	https://www.example.com/login

SSH (Secure Shell) [1]

- **SSH** is a **cryptographic network protocol** used to securely access and manage network devices, servers, and systems over an **unsecured network**.
  - Operates on **TCP port 22** by default
  - Provides **secure remote login**, **command execution**, and **file transfer**
  - Uses **asymmetric encryption** (public/private key pair) for authentication and **symmetric encryption** for session data
- **Purpose**
  - Securely **log into remote systems**
  - Execute commands on remote machines
  - Transfer files securely (via **SCP** or **SFTP**)
  - **Tunnel** other protocols securely (SSH tunneling / port forwarding)
  - Replace **insecure protocols** like Telnet, FTP, and rlogin

SSH (Secure Shell) [2]

❑Pros

Benefit	Description
Strong Encryption	Encrypts data, credentials, and commands sent over the network
Authentication Options	Supports password and **key-based authentication** (more secure)
Secure File Transfer	Supports **SCP** and **SFTP** for encrypted file movement
Port Forwarding / Tunneling	Can securely tunnel other protocols (e.g., RDP, VNC)
Automation Support	Ideal for scripts, remote job execution, and server management
Cross-platform	Works on Windows, Linux, macOS, and embedded systems

SSH (Secure Shell) [3]

❑Cons

Limitation	Description
Target for Brute Force Attacks	If exposed to the internet, SSH is often scanned and attacked
Key Mismanagement	Lost or improperly stored keys can cause security or access issues
Configuration Complexity	Requires proper setup to harden (e.g., disabling root login, enforcing key auth)
Credential Leakage	If credentials or private keys are compromised, attackers can gain full control

Summary

Feature	SSH (Secure Shell)
Port	TCP 22
Security	Encrypted remote access, supports key-based auth
Used For	Remote administration, secure file transfers, tunneling
Pros	Encrypted, flexible, scriptable, secure login
Cons	Targeted by attackers, requires proper hardening
Example	ssh user@192.168.1.10 for remote server access

SFTP (Secure File Transfer Protocol) [1]

- **SFTP** is a **secure file transfer protocol** that operates over **SSH (Secure Shell)** to provide **encrypted file access, transfer, and management**.
  - It is **not the same** as FTP over SSL (that's FTPS).
  - Uses **TCP port 22** (same as SSH).
  - Provides file transfer capabilities similar to FTP, but with **built-in encryption and authentication** via SSH.
- **Purpose**
  - SFTP is used to:
  - **Securely transfer files** over untrusted networks (e.g., internet)
  - **Authenticate users** via SSH (passwords or key-based)
  - Support secure **upload/download**, **directory listing**, **file deletion**, and **permissions management**
  - Replace **insecure file transfer protocols** like FTP and RCP

SFTP (Secure File Transfer Protocol) [2]

❑ **Pros of SFTP**

Benefit	Description
Strong Encryption	Data and credentials are encrypted using SSH (end-to-end security)
Authentication Support	Supports **password** and **key-based** authentication
Firewall-Friendly	Only uses one port (TCP 22), unlike FTP which requires multiple
Scriptable & Automatable	Easily used in **cron jobs, backup scripts, and CI/CD pipelines**
Secure File Management	Supports file permissions, resuming transfers, and directory listing securely

SFTP (Secure File Transfer Protocol) [3]

❑ **Cons of SFTP**

Limitation	Description
Requires SSH Access	Needs SSH server running on the host, which may be overkill for simple file sharing
Key Management Complexity	Public/private key setup and rotation require discipline and security hygiene
Security Risk if Misconfigured	Improper SSH/SFTP configurations (e.g., root access or open internet exposure) can be exploited
No Built-in GUI	Native SFTP is command-line-based (though GUIs like WinSCP, FileZilla, or Cyberduck are available)

Summary

Feature	SFTP
Protocol	Secure File Transfer Protocol (over SSH)
Port	TCP 22
Encryption	Yes – via SSH
Used For	Secure file transfer, automated backups, remote file management
Pros	Encrypted, single port, scriptable, supports authentication
Cons	Needs SSH server, config complexity, key management overhead
Example	sftp user@host to transfer sensitive project files

IPsec (Internet Protocol Security) [1]

- **IPsec** is a **suite of protocols** used to **secure IP communications** by **authenticating and encrypting** each IP packet in a communication session.
  - Operates at **Layer 3 (Network Layer)** of the OSI model
  - Works with both **IPv4 and IPv6**
  - Can be used for **site-to-site** and **remote-access VPNs**
- **Purpose**
  - **Encrypt and authenticate** IP packets between devices
  - Secure **VPN tunnels** across untrusted networks (e.g., the internet)
  - Ensure **data confidentiality, integrity, and authenticity**
  - Protect against **packet sniffing**, **replay attacks**, and **IP spoofing**

IPsec (Internet Protocol Security) [2]

❑Key Features of IPsec:

Component	Description
AH (Authentication Header)	Provides data origin authentication and integrity, but **no encryption**
ESP (Encapsulating Security Payload)	Provides **encryption, authentication, and integrity**
IKE (Internet Key Exchange)	Negotiates and manages security keys and parameters
Transport Mode	Secures only the payload of the IP packet (used for host-to-host)
Tunnel Mode	Secures the entire IP packet (used for network-to-network or VPNs)

IPsec (Internet Protocol Security) [3]

❑Pros of IPsec

Benefit	Description
End-to-End Encryption	Secures all IP traffic, not just specific apps
Application-Agnostic	Works below the application layer, so no app changes are needed
Strong Authentication & Integrity	Uses cryptographic methods (HMAC, AES, SHA) to protect against tampering
Supports VPNs	Foundation of **site-to-site** and **remote-access VPNs**
Flexible Modes	Can be used in **transport** or **tunnel** mode depending on the use case

IPsec (Internet Protocol Security) [4]

❑Cons of IPsec

Limitation	Description
Complex Configuration	Requires careful setup (IPsec policies, key exchange, modes) and interoperability testing
Performance Overhead	Encryption and encapsulation can reduce network throughput or increase latency
Firewall/Router Compatibility	Some routers or firewalls may block or mishandle IPsec packets (especially NAT traversal)
Troubleshooting Difficulty	Encrypted traffic is hard to inspect or debug
May Not Work Well Over NAT	Needs **NAT Traversal (NAT-T)** in many real-world environments

Summary

Feature	IPsec (Internet Protocol Security)
Layer	Network Layer (Layer 3)
Security	Encryption, authentication, integrity
Key Protocols	AH, ESP, IKE
Modes	Transport mode, Tunnel mode
Common Use	VPNs (site-to-site or client-to-site)
Pros	Strong encryption, app-agnostic, end-to-end
Cons	Complex setup, performance impact, NAT issues
Example	Secure communication between two office networks via IPsec VPN

SSL/TLS [1]

- **SSL/TLS (Secure Sockets Layer / Transport Layer Security)**
- **SSL** and **TLS** are cryptographic protocols that provide **secure communication** over a computer network by encrypting the data exchanged between two endpoints (e.g., client and server).
  - **TLS** is the modern, secure successor to SSL.
  - **TLS 1.2 and 1.3** are the current secure versions; **SSL 2.0/3.0** and **TLS 1.0/1.1** are deprecated.
  - Operates at the **presentation** and **application layers** of the OSI model.
  - Commonly used in **HTTPS**, **email (SMTP, IMAP, POP3)**, **VoIP**, and **VPNs**.

SSL/TLS [2]

❑Purpose

- **Encrypt** data in transit to prevent eavesdropping.
- **Authenticate** servers (and optionally clients) using digital certificates.
- Ensure **data integrity** so it isn’t modified in transit.
- Protect against **man-in-the-middle (MITM)** attacks.

SSL/TLS [3]

❑ **Pros of SSL/TLS**

Benefit	Description
Strong Encryption	Protects sensitive data (e.g., passwords, credit card info) from being intercepted
Authentication	Confirms the identity of the server (and optionally the client) using digital certificates
Data Integrity	Ensures transmitted data is not altered in transit
Widely Supported	Built into all major browsers, operating systems, and web servers
Improves Trust & SEO	HTTPS sites using TLS show a padlock icon and are favored in search engine rankings

SSL/TLS [4]

❑ **Cons of SSL/TLS**

Limitation	Description
Certificate Management	Requires purchasing, installing, and renewing SSL/TLS certificates (unless using a free CA like Let's Encrypt)
Performance Overhead	Initial handshake and encryption add slight processing load (minimal with TLS 1.3)
Certificate Expiry	Expired or misconfigured certs can break access or cause browser warnings
False Sense of Security	TLS secures transmission but **not** the endpoint itself (a compromised server or phishing site can still use HTTPS)
Deprecated Versions Still in Use	SSLv2, SSLv3, and TLS 1.0/1.1 are insecure and must be disabled

Summary

Feature	SSL/TLS
Layer	Transport Layer (Layer 4)
Purpose	Secure data in transit (encryption + auth)
Successor	TLS replaces SSL (SSL is deprecated)
Encryption	Yes (e.g., AES, ChaCha20)
Authentication	Yes (via X.509 certificates)
Port	443 (HTTPS), 993 (IMAPS), 465 (SMTPS), etc.
Pros	Encrypts data, authenticates identity, widely supported
Cons	Certificate management, endpoint still vulnerable
Example	HTTPS website (https://bank.com/login)

Insecure protocols

Telnet [1]

- **Telnet (Telecommunication Network Protocol)**
- **Telnet** is an **application-layer protocol** used to **remotely access and manage devices** over a network using a command-line interface.
  - Defined in **RFC 854**
  - Operates on **TCP port 23**
  - Provides a **virtual terminal connection** to remote systems
  - Developed in the 1960s — now **considered obsolete and insecure**
- **Purpose**
  - **Remotely log into servers, routers, and switches**
  - Run **commands** and manage **network devices** or **UNIX/Linux systems**
  - Test **network connectivity** to specific ports (similar to nc or netcat today)

Telnet [2]

❑Pros of Telnet

Advantage	Description
Simple to Use	Lightweight and easy to configure for basic CLI access
Low Resource Usage	Minimal bandwidth and processing requirements
Good for Testing	Still sometimes used for **basic network service testing** (e.g., checking if a port is open)
Widespread Support	Supported on many older systems and devices

Telnet [3]

❑Cons of Telnet

Limitation	Description
No Encryption	All data, including **usernames and passwords**, is sent in **plaintext**
Easily Intercepted	Susceptible to **packet sniffing** and **MITM (Man-in-the-Middle)** attacks
No Authentication of Server	Users cannot verify they are connecting to a legitimate system
Outdated Protocol	Replaced by **SSH** in almost all modern secure environments
Disabled by Default on Most OSes	Considered a **security liability** if enabled on production systems

Summary

Feature	Telnet
Protocol	Text-based remote access
Port	TCP 23
Encryption	No – all data in plaintext
Use Today?	Strongly discouraged; legacy use only
Replacement	**SSH (Secure Shell)**
Migration	Disable Telnet service on all systems
Example	telnet server.com 23

FTP (File Transfer Protocol) [1]

- **FTP** is a **standard network protocol** used to **transfer files** between a client and a server over a **TCP/IP network**.
  - Defined in **RFC 959**
  - Operates on **TCP port 21** (control channel) and **TCP port 20** (data channel)
  - Works in **active** or **passive** mode for data connection
  - One of the **earliest Internet protocols**, developed in the 1970s
- **Purpose**
  - **Upload** and **download** files between computers and servers
  - Manage file directories on remote systems (list, rename, delete, etc.)
  - Facilitate **data exchange** for websites, backups, and application distribution

FTP (File Transfer Protocol) [2]

❑Pros of FTP

Advantage	Description
Simple File Transfers	Easy way to move files between client and server
Broad Compatibility	Supported on most operating systems, applications, and hardware
Efficient for Large Files	Handles large file transfers well compared to email or HTTP
Scriptable	Easily automated with batch files or command-line tools

FTP (File Transfer Protocol) [3]

❑ **Cons of FTP**

Limitation	Description
No Encryption	**Usernames, passwords, and file contents** are transmitted in **plaintext**
Vulnerable to MITM and Sniffing	Attackers on the network can easily intercept and read FTP traffic
Uses Multiple Ports	Requires opening additional ports for data connections, making firewall configuration complex
Lacks Strong Authentication	No support for secure tokens, certificates, or multi-factor authentication
Non-Compliant	Not compliant with most modern security standards (e.g., PCI-DSS, HIPAA, GDPR)

Summary

Feature	FTP
Port	TCP 21 (plus dynamic ports for data)
Encryption	None — all data in plaintext
Use Case	Legacy file transfer
Modern Use?	Not recommended
Secure Replacements	SFTP, FTPS, HTTPS
Migration	Replace automated FTP scripts with SFTP tools like sftp, WinSCP, or FileZilla (SFTP mode)
Example	ftp ftp.example.com to upload files

HTTP (HyperText Transfer Protocol) [1]

- **HTTP** is an **application-layer protocol** used to transmit **hypertext (web) data** between a **client** (typically a browser) and a **web server**.
  - Defined in **RFC 2616** (HTTP/1.1) and updated by newer RFCs (HTTP/2, HTTP/3)
  - Operates by default on **TCP port 80**
  - Stateless: each request-response cycle is independent
  - Foundation of data communication for the **World Wide Web**
- **Purpose**
  - **Transfer web content** such as HTML, CSS, JavaScript, images, etc.
  - Enable **client-server communication** for websites and web applications
  - Support **request-response** interactions (e.g., GET, POST, PUT, DELETE)
  - Serve as the **basis** for RESTful APIs and modern web services

HTTP (HyperText Transfer Protocol) [2]

❑Pros of HTTP

Advantage	Description
Widely Supported	Universally compatible with browsers, servers, and devices
Simple to Implement	No need for certificate management or TLS configuration
Low Overhead	Faster initial connections (no TLS handshake) — though negligible with HTTP/2+TLS
Transparent for Debugging	Easy to inspect HTTP requests/responses for testing (but not a benefit in production)

HTTP (HyperText Transfer Protocol) [3]

❑ **Cons of HTTP**

Limitation	Description
No Encryption	All data (including passwords, form entries, cookies) is transmitted in **plaintext**
Susceptible to MITM Attacks	Attackers can intercept or alter communications on open or compromised networks
No Authentication	Clients can't verify they're talking to the legitimate server (e.g., phishing risks)
Not Compliant with Security Standards	Not acceptable for use in handling sensitive data (e.g., under **PCI-DSS**, **HIPAA**, **GDPR**)
No Data Integrity	Attackers can modify content in transit without detection

Summary

Feature	HTTP
Port	TCP **80**
Encryption	None — all data in plaintext
Use Case	Basic web browsing (non-sensitive info)
Modern Use?	**Not recommended** for any production or sensitive data
Secure Replacements	**HTTPS (TLS-encrypted)**
Migration	Redirect all HTTP requests to HTTPS automatically
Example	http://example.com (no padlock in browser)

VPNs

VPNs (Virtual Private Networks) [1]

- A **VPN** creates a **secure, encrypted connection (tunnel)** over an untrusted network (usually the Internet) between a user/device and a remote network.
  - It allows users or devices to access **private/internal network resources** as if they were physically connected.
  - Utilizes **tunneling protocols** and **encryption** to ensure data confidentiality, integrity, and authenticity.
- **Purpose**
  - Enable **secure remote access** for users working off-site.
  - Connect **branch offices** to headquarters (**site-to-site VPN**).
  - **Protect data** in transit over public networks.
  - Bypass **geo-restrictions** and censorship (common in consumer VPN use).
  - Ensure **compliance** and **security** when accessing sensitive enterprise resources.

Common Tunneling Protocols

Protocol	Description	Port	Encryption	Use Case
IPsec	Network-layer security suite	UDP 500, 4500	Yes	Site-to-site and remote access VPNs
OpenVPN	Open-source, SSL/TLS-based VPN	TCP/UDP 1194 (configurable)	Yes	Remote access VPNs, highly configurable
WireGuard	Lightweight, modern VPN protocol	UDP (configurable)	Yes	Fast and secure VPNs, easy to deploy
L2TP/IPsec	Layer 2 Tunneling Protocol with IPsec for encryption	UDP 1701, 500, 4500	Yes	Legacy support, moderate security
SSTP	Microsoft protocol using HTTPS tunnel	TCP 443	Yes	Bypasses firewalls, integrates with Windows
PPTP	Point-to-Point Tunneling Protocol	TCP 1723	Weak	Obsolete, insecure (avoid)

VPNs (Virtual Private Networks) [2]

❑ **Pros of VPNs**

Benefit	Description
Secure Communication	Encrypts all traffic between client and server or between sites
Remote Access to Internal Networks	Employees can access corporate resources securely from anywhere
Firewall/NAT Traversal	Many VPNs (e.g., SSTP, OpenVPN) can bypass strict firewalls using common ports
Location Privacy	Masks your IP address and location
Compliance-Friendly	Helps meet security requirements for **HIPAA**, **PCI-DSS**, **GDPR**, etc.

VPNs (Virtual Private Networks) [3]

❑ **Cons of VPNs**

Limitation	Description
Performance Impact	Encryption and tunneling may slow down connection speeds
Complex Setup	Requires configuration of VPN servers, certificates, firewalls, and clients
Single Point of Failure	If the VPN server goes down, remote users may lose access
Credential or Key Misuse	Compromised credentials or keys can expose internal resources
False Sense of Security	VPNs protect traffic, but **not endpoint security** (e.g., malware can still spread)

Summary

Feature	VPNs & Tunneling Protocols
Function	Secure, encrypted communication over the Internet
Main Benefit	Protects data in transit; enables secure remote access
Common Protocols	IPsec, OpenVPN, WireGuard, SSTP, L2TP/IPsec, PPTP
Pros	Encryption, privacy, remote access, compliance
Cons	Setup complexity, performance, endpoint dependency
Example	Remote workers connecting to office via OpenVPN

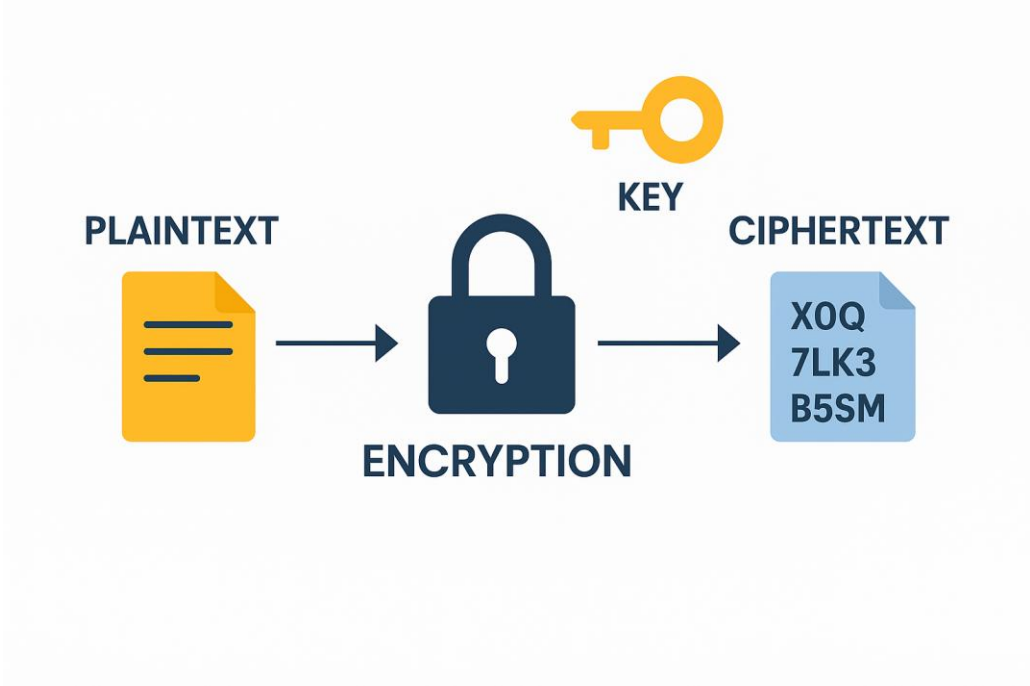
Encryption Basics

What Is Encryption?

❑ **Encryption** is the process of converting **plaintext** (readable data) into **ciphertext** (unreadable format) using an **algorithm** and a **key**. It protects data **confidentiality**, ensuring only authorized parties can access the original information.

❑ **Types of Encryption**

- Symmetric Encryption
- Asymmetric Encryption



Symmetric Encryption [1]

❑Symmetric Encryption

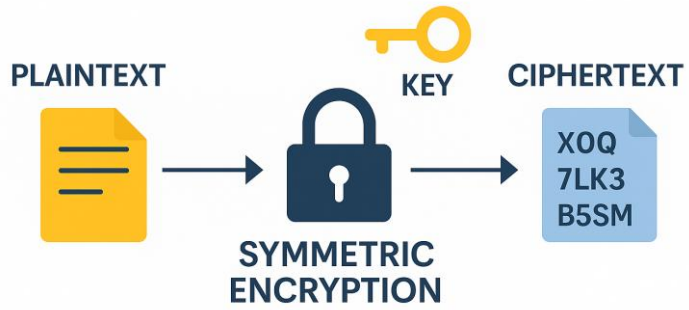
- **One key** is used for both **encryption and decryption**.

❑Features:

- Fast and efficient
- Ideal for encrypting large amounts of data (e.g., full-disk encryption)
- Requires secure key exchange to avoid interception

❑Common Algorithms:

- AES (Advanced Encryption Standard)
- DES (deprecated)
- 3DES (legacy, phased out)
- Blowfish



Symmetric Encryption [2]

❑ Pros:

- Fast and efficient (especially for large data volumes)
- Requires less computing power
- Ideal for **bulk encryption**

❑ Cons:

- **Key distribution problem**: both sender and receiver must **securely share the same key**
- Doesn't scale well for large networks (e.g., 1,000 users = 499,500 key pairs)

❑ Example Use Case:

- Encrypting a hard drive or database (e.g., AES-256 encryption for full-disk encryption)

Asymmetric Encryption [1]

❑Asymmetric Encryption

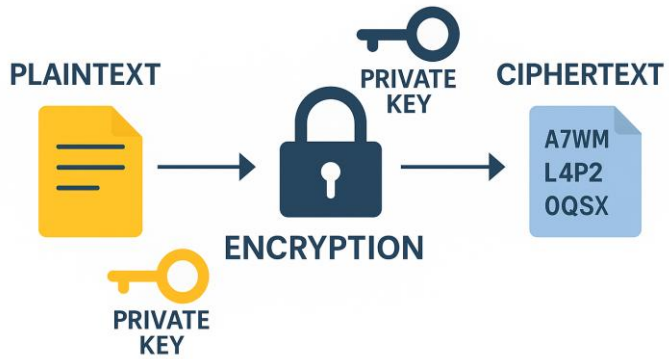
- Uses a **public key** to encrypt and a **private key** to decrypt.

❑Features:

- Eliminates the need to share a secret key
- Based on mathematical relationships (e.g., prime factorization, elliptic curves)
- Used for **key exchange**, **digital signatures**, and **authentication**

❑Common Algorithms:

- RSA
- ECC (Elliptic Curve Cryptography)
- DSA (for digital signatures)



Asymmetric Encryption [2]

❑Pros:

- **Solves the key distribution problem**
- Enables **digital signatures** and **authentication**
- Secure for **open communication channels**

❑Cons:

- **Slower** than symmetric encryption
- Not suitable for large data encryption (often used to exchange a symmetric key)

❑Example Use Case:

- Sending an encrypted email with **PGP/GPG**
- **TLS/SSL handshake** in HTTPS — asymmetric encryption is used to exchange a symmetric session key

Hybrid Encryption

- Many secure systems (e.g., HTTPS, VPNs) use **both**:
  - **Asymmetric encryption** to securely exchange a session key
  - **Symmetric encryption** to efficiently encrypt the actual data
  - Example: TLS (used in HTTPS) uses **RSA or ECC** to exchange keys and **AES** for the session

Summary

Feature	Symmetric Encryption	Asymmetric Encryption
Keys Used	1 (same key)	2 (public/private pair)
Speed	Fast	Slower
Security	Secure, but hard to manage at scale	More secure key exchange
Use Case	Data at rest, VPN payloads	Email, TLS key exchange
Challenge	Key distribution	Processing speed

Monitoring, Logging, and Incident Response

Agenda

❑ Monitoring, Logging, and Incident Response

- Network monitoring tools: Wireshark, Zeek, Snort
- Importance of centralized logging (SIEM)
- Setting up alerts and thresholds
- Incident response basics: detection, containment, recovery

Network Monitoring Tools

Network Monitoring

- **Network Monitoring** is the process of **continuously observing a network** to track **performance**, **availability**, and **security**. It involves collecting and analyzing data from devices like routers, switches, firewalls, servers, and endpoints.

- **Purpose**
  - Ensure **network uptime and performance**
  - Detect **failures**, **bottlenecks**, or **latency issues**
  - Identify **unauthorized access**, **intrusions**, or **malicious traffic**
  - Monitor **bandwidth usage** and application performance
  - Support **troubleshooting**, **capacity planning**, and **compliance**

Types of Network Monitoring

Type	Description	Tools
Performance Monitoring	Tracks metrics like latency, packet loss, throughput, jitter	PRTG, SolarWinds, Nagios
Availability Monitoring	Checks whether devices and services are online (ping, SNMP)	Zabbix, Pingdom
Traffic Monitoring	Analyzes traffic flow and bandwidth usage (NetFlow, sFlow)	ntopng, Wireshark (ไม่เหมาะกับเครือข่ายใหญ่ๆ) tshark (CLI)
Security Monitoring	Detects threats and anomalies (IDS/IPS, log analysis)	Zeek, Snort, Suricata
Log Monitoring	Collects and analyzes logs for patterns and incidents	ELK Stack, Graylog, Splunk
Application Monitoring	Monitors app-level performance (web, DNS, DB, etc.)	AppDynamics, Dynatrace

Benefits of Network Monitoring

Benefit	Description
Proactive Problem Detection	Identify and resolve issues before users notice
Performance Optimization	Tune networks for optimal bandwidth and reliability
Security Visibility	Detect suspicious or unauthorized traffic in real time
Compliance Support	Meet requirements like PCI-DSS, HIPAA, and ISO 27001
Reporting & Analytics	Create visual dashboards and historical trends
Capacity Planning	Understand growth patterns and plan infrastructure scaling

Network Monitoring Tools

Tool	Focus	Type
Wireshark	Packet analysis	Traffic monitoring
Zeek	Security event logging	Network security
Snort/Suricata	IDS/IPS	Intrusion detection
Nagios	Health checks	Performance/availability
Zabbix	Device and service monitoring	Performance
PRTG	All-in-one dashboard	Multi-layer monitoring
ntopng	Traffic flow and analytics	Bandwidth monitoring
ELK Stack (Elasticsearch, Logstash, Kibana)	Log aggregation & visualization	Log monitoring

Wireshark [1]

- Wireshark is a **GUI-based network protocol analyzer** used to **capture and inspect live packet data** from a network interface in real-time.
- **Purpose**
  - Troubleshoot network issues (latency, drops, errors)
  - Analyze protocol behavior (HTTP, TCP, DNS, etc.)
  - Inspect packet contents and flows
  - Understand and visualize **network traffic patterns**

Wireshark [2]

❑Pros

- Real-time packet capture and deep protocol inspection
- User-friendly **graphical interface** with advanced filters
- Supports a **wide range of protocols** (1000+)
- Great for **learning, auditing, and forensics**

❑Cons

- Not ideal for large-scale or continuous monitoring
- GUI can be **resource-intensive**
- Requires **manual analysis** (not automated)
- Sensitive data is visible — must be used **ethically and legally**

Zeek [1]

- Zeek is a **network security monitoring framework** that passively logs and analyzes network traffic to generate **high-level security event data**
- **Purpose**
  - Detect and log **network behaviors** and **security-relevant events**
  - Extract metadata (e.g., DNS queries, HTTP requests, TLS handshakes)
  - Integrate with SIEMs for **threat hunting and incident response**

Zeek [2]

❑Pros

- Focuses on **contextual, high-level metadata** instead of full packet capture
- Scalable for **large enterprise environments**
- Highly **scriptable and extensible**
- Low storage overhead compared to full PCAP capture

❑Cons

- Steep learning curve for configuration and scripting
- Not a real-time alerting engine (it's a **network recorder**, not a sniffer)
- Not beginner-friendly; better suited for **advanced analysts**

Snort [1]

- Snort is a **real-time Intrusion Detection and Prevention System (IDS/IPS)** that **analyzes packet headers and payloads** for signs of malicious activity using **signature-based detection**.
- **Purpose**
  - Detect and block **network-based threats** (e.g., exploits, malware, port scans)
  - Inspect traffic using **defined rule sets**
  - Actively respond to threats (in IPS mode)

Snort [2]

❑ Pros

- Real-time detection and alerting
- Huge community and regularly updated **rule sets** (e.g., from Emerging Threats, Snort.org)
- Can operate in IDS or **IPS mode** (inline traffic blocking)
- Lightweight and works on **low-resource devices**

❑ Cons

- Primarily **signature-based** — may miss **zero-day or unknown threats**
- Can generate **false positives** (needs fine-tuning)
- Doesn't provide **deep protocol behavior** like Zeek

Summary

Feature	**Wireshark**	**Zeek**	**Snort**
Type	Packet analyzer	Network security monitor (NSM)	IDS/IPS
Interface	GUI/CLI	CLI	CLI
Primary Use	Troubleshooting, learning	Network behavior logging	Threat detection
Real-time	Yes (manual)	No (logs events)	Yes (real-time alerting)
Extensible?	Filters	Scripts	Rules
Best For	Protocol analysis, live capture	Forensics, behavior analysis	Network threat detection

SIEM

SIEM [1]

- **SIEM** is a centralized platform that **collects, analyzes, correlates, and stores log and event data** from across your network infrastructure for **real-time threat detection, security monitoring, and compliance reporting**

Purpose of SIEM

- **Centralized Log Management:** Aggregate logs from firewalls, IDS/IPS, servers, endpoints, apps, etc.
- **Threat Detection & Correlation:** Use correlation rules to detect suspicious patterns (e.g., brute-force login attempts).
- **Alerting:** Notify admins/SOC when anomalies or known threats are detected.
- **Incident Response Support:** Provide forensic data for investigation and remediation.
- **Compliance Reporting:** Help meet regulatory standards (e.g., PCI-DSS, HIPAA, GDPR) with built-in templates.

SIEM [2]

❑Pros of SIEM

Advantage	Description
Centralized visibility	Single-pane-of-glass view across all systems and events
Real-time detection	Identify attacks (DDoS, privilege escalation, lateral movement) as they happen
Compliance support	Generate reports for auditors with minimal manual effort
Forensic analysis	Trace incident timelines and origins with historical log data
Automation and playbooks	Some SIEMs offer SOAR-like capabilities for automated response

SIEM [3]

❑Cons of SIEM

Disadvantage	Description
Cost	Licensing, storage, and scaling costs can be high
Complex setup	Requires proper tuning and configuration to reduce false positives
False positives	Without good correlation rules, noise overwhelms real alerts
Resource-intensive	Needs trained staff to operate and maintain effectively
Time to value	ROI takes time—especially during the initial learning and tuning phase

SIEM [4]

❑ **SIEM Tools Example**

Tool	Description
Splunk Enterprise Security	Powerful and flexible; large ecosystem; expensive
IBM QRadar	Great for correlation and compliance; enterprise-focused
Microsoft Sentinel	Cloud-native SIEM for Azure environments; scalable
LogRhythm	Mid-market focus; good balance of features and usability
Elastic SIEM (ELK Stack)	Open-source; customizable but needs more hands-on management
Graylog	Lightweight, open-source alternative with enterprise options
ArcSight (OpenText)	Mature, large-scale deployments; historically used by governments

Alerts and Thresholds

Alerts and Thresholds

- Setting up **alerts and thresholds** is one of the most important aspects of configuring a SIEM or any security monitoring system. Done well, it enables fast detection and response to real threats. Done poorly, it leads to **alert fatigue**, missed incidents, and wasted resources.

- **Define Your Objectives**
  - What are the **most critical assets and systems**?
  - What are the **biggest risks** (e.g., data exfiltration, privilege abuse)?
  - What regulations or compliance requirements apply?

User Activity Alerts

Event	Example Threshold
Login from a new geo-location	User logs in from two countries within an hour
Excessive failed login attempts	>10 failures in 5 minutes
Privilege escalation	Standard user suddenly gets admin privileges
Access outside working hours	Login between 12 AM – 5 AM

System & Network Alerts

Event	Example Threshold
High CPU/network usage	>85% CPU for more than 5 minutes
Unexpected service restarts	Multiple critical services restarting
New listening ports	Port opened that wasn’t used previously
Unauthorized software installation	Detected new binaries in sensitive systems

Security-Specific Alerts

Event	Example Threshold
Malware detection	Any malware signature match
IDS/IPS alerts	Severity = HIGH or CRITICAL
Suspicious PowerShell use	PowerShell calling encoded scripts
Multiple disabled security tools	AV, firewall, or EDR turned off on multiple machines

Data Exfiltration Alerts

Event	Example Threshold
Large file transfers to external IPs	>500MB sent to untrusted destination in <10 min
Unusual use of cloud storage	Uploads to Dropbox/Google Drive from internal hosts
Email exfiltration	Multiple large attachments sent to external emails

Incident response

Incident Response [1]

- **Incident Response (IR)** is a structured approach to **identify, manage, and mitigate security incidents or network disruptions**. It ensures an organization can effectively detect, contain, and recover from events such as cyberattacks, system failures, or data breaches with minimal damage and downtime.

- **Purpose**
  - **Minimize Impact**: Reduce the damage caused by an incident (data loss, downtime, financial cost)
  - **Restore Operations**: Quickly bring systems and services back to normal
  - **Preserve Evidence**: Maintain data integrity for forensic analysis or legal needs
  - **Comply with Regulations**: Meet standards like GDPR, HIPAA, PCI-DSS
  - **Improve Security Posture**: Learn from incidents and strengthen defenses

Incident Response [2]

❑ **Pros of Incident Response**

Advantage	Description
Faster Response Times	Defined playbooks and roles mean quicker actions
Reduced Risk Exposure	Limits lateral movement and data exfiltration
Improved Forensics	Preserves logs, timelines, and artifacts
Continuous Improvement	Post-incident reviews improve future responses
Compliance Friendly	Supports audits and security certifications
Clear Responsibility	Reduces confusion during high-stress events

Incident Response [3]

❑Cons of Incident Response

Disadvantage	Description
Requires Time & Resources	Building and maintaining IR plans takes effort
Potentially Costly	May need specialized tools, staff, or external IR teams
Needs Skilled Personnel	Not all IT staff are trained for IR (e.g., forensics, malware analysis)
False Positives	Poorly tuned systems may trigger unnecessary responses
Reputation Risk	Improper handling can lead to worse PR or legal consequences

Scenario: Ransomware Attack

Phase	Action
Detection	EDR alerts on file encryption behavior on a workstation. Suspicious .lock file extensions observed.
Containment	Infected machine is isolated from the network. User account is disabled. Network shares disconnected.
Eradication	Malware is removed. Full system scan conducted. Persistence mechanisms are checked.
Recovery	Clean backup is restored. Security patches applied. User accesses reinstated.
Lessons Learned	Email filtering and endpoint protection reviewed. Employees re-trained on phishing awareness. IR playbook updated.

Common Attacks & Defense Mechanisms

MITM (Man-in-the-Middle) Attack

- A **Man-in-the-Middle (MITM)** attack occurs when an attacker **intercepts and possibly alters communication between two parties** (e.g., client and server) **without their knowledge**. The attacker positions themselves "in the middle" to eavesdrop or manipulate the data being exchanged.

- **Purpose (from the attacker's perspective):**
  - **Eavesdropping**: Capture sensitive information like credentials, session cookies, or banking data
  - **Data Manipulation**: Alter or inject malicious data into communications
  - **Session Hijacking**: Steal authenticated sessions to impersonate users
  - **Credential Harvesting**: Collect usernames and passwords
  - **Traffic Redirection**: Redirect users to malicious websites

Example: Public Wi-Fi MITM Attack

❑Scenario:

- An attacker sets up a rogue access point (e.g., “Free_Coffee_WiFi”) in a coffee shop. Unsuspecting users connect, thinking it's legitimate.

❑MITM Tactics Used:

- **ARP Spoofing**: Redirects local LAN traffic through attacker's machine
- **SSL Stripping**: Downgrades HTTPS to HTTP to capture plaintext credentials
- **Traffic Interception**: Harvests usernames and passwords from login forms

❑Impact:

- Attacker steals credentials for email, social media, or even banking
- Users are unaware — everything “looks normal” in their browser

DNS Poisoning (DNS Spoofing)

- **DNS poisoning** is an attack where a malicious actor injects **false DNS records** into a DNS resolver's cache, causing users to be redirected to **fraudulent or malicious websites** without their knowledge.

- **Purpose (Attacker Goals):**
  - **Phishing / Credential Theft**: Redirect users to fake login portals
  - **Malware Distribution**: Serve malicious payloads via fake websites
  - **Traffic Interception**: Route traffic through attacker-controlled servers
  - **Denial of Service**: Disrupt service access by misdirecting requests
  - **Bypass Filtering**: Circumvent DNS-based security controls (e.g., parental controls, enterprise filtering)

Example Scenario

- **Example Scenario: Fake Banking Site via DNS Poisoning**
- Attacker targets a public or vulnerable DNS server.
- They inject a **fake record** for bank.com pointing to 5.5.5.5 (a malicious server).
- User tries to visit https://bank.com and gets redirected to a **lookalike phishing site**.
- Victim enters login credentials, which are captured by the attacker.
- Even if the user types the correct URL, DNS resolution sends them to the **wrong IP**.

ARP Spoofing

- **ARP spoofing** (also known as **ARP poisoning**) is a Layer 2 attack in which an attacker **sends forged ARP messages** on a local area network (LAN) to associate their **MAC address with the IP address of another device**, such as a **default gateway** or another host.

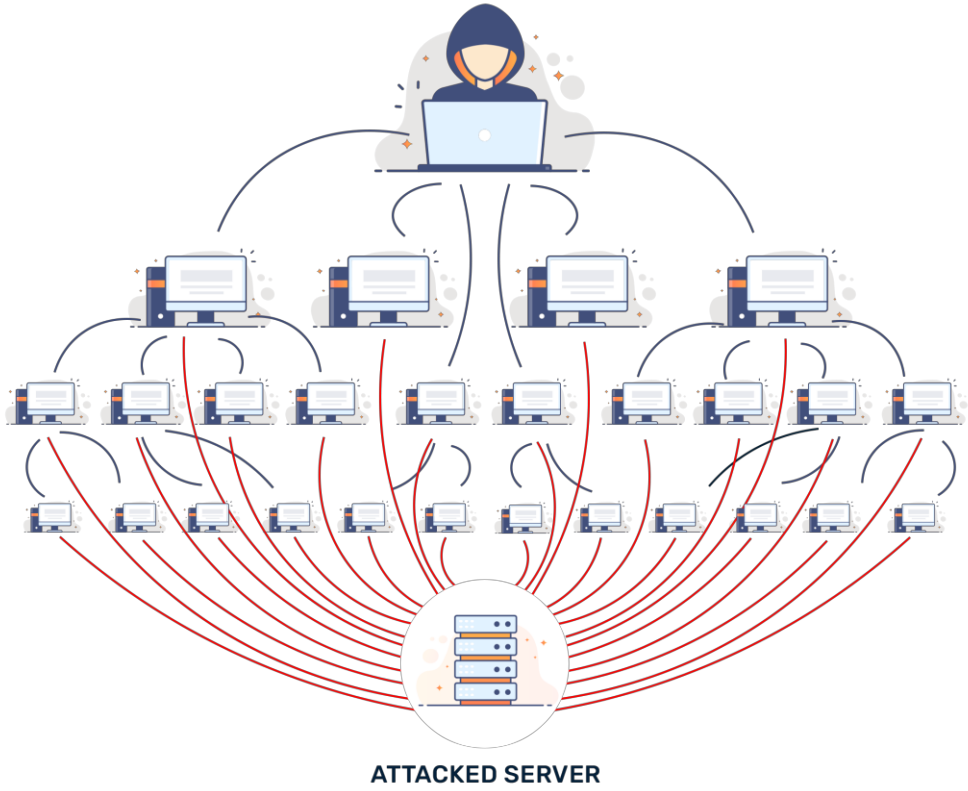
- **Purpose (Attacker Objectives):**
  - **Interception of Traffic**: Spy on data between devices (e.g., user ↔ internet)
  - **Impersonation**: Masquerade as another device to steal sessions or credentials
  - **Packet Manipulation or Injection**: Alter or inject malicious data into traffic
  - **Denial of Service (DoS)**: Disrupt communication by misrouting or black-holing packets
  - **Session Hijacking or Cookie Theft**: Capture active sessions from users

Example Scenario

- **Example Scenario: ARP Spoofing in a LAN**
- Attacker connects to a company's Wi-Fi or switches to an internal LAN.
- They run arpspoof to send fake ARP replies:
  - Claiming **their MAC** is associated with the **default gateway's IP**.
  - And also with **the victim's IP** (bidirectional spoofing).
- Now all traffic between the victim and the internet **routes through the attacker's system**.
- Attacker can:
  - **Sniff credentials**, even for HTTPS (if SSL stripping is used)
  - **Inject JavaScript** into web traffic
  - **Block or alter packets** for DoS

DDoS (Distributed Denial of Service)

- A **DDoS attack** is a type of cyberattack where **multiple systems flood a target (like a server, network, or website)** with excessive traffic or requests to **disrupt normal service availability**.
- DDoS attacks are "**distributed**" because they often use **botnets**—large networks of compromised devices—to launch attacks simultaneously from many geographic locations.



DDoS (Distributed Denial of Service)

- **Purpose (Attacker's Intent):**
- **Service Disruption**: Take down websites, applications, or entire networks
- **Extortion**: Demand ransom to stop attacks (ransom DDoS)
- **Distraction**: Divert attention from other attacks (e.g., data exfiltration)
- **Testing Defenses**: Probe infrastructure resilience
- **Ideological/Political Motivation**: Hacktivism (e.g., Anonymous)
- **Business Sabotage**: Cripple competitors (unethical/unlawful)

Example Scenario

- **Example Scenario: UDP Flood on Gaming Server**
- Attacker uses a botnet to send millions of UDP packets per second to a gaming server.
- Server allocates resources to process fake traffic, becomes overwhelmed.
- Legitimate users are disconnected or experience severe lag.
- Attacker demands payment to stop the attack (ransom DDoS).

Defense Strategies

NAC (Network Access Control)

❑ **Network Access Control (NAC)** is a cybersecurity approach that enforces security policies to control which devices and users can access a network. It checks the identity, security posture, and compliance status of devices before and during their connection to the network.

❑ **Purpose**

- **Control Access**: Ensure only authorized and compliant devices/users access the network.
- **Policy Enforcement**: Enforce company policies (e.g., antivirus, patches, OS version).
- **Contain Threats**: Quarantine non-compliant or suspicious endpoints automatically.
- **Visibility**: Gain insight into what devices are connected to your network and where.

Activities in Network Access Control

Activity	Description
Authentication	Verifies user or device identity using credentials or certificates (e.g., 802.1X, RADIUS).
Posture Assessment	Checks the security status of devices (e.g., AV installed, patched, no malware).
Authorization	Grants network access based on role, device type, location, time, etc.
Remediation/Quarantine	Non-compliant devices are redirected to a quarantine network or remediation portal.
Guest Management	Provides limited and temporary access to guests, often via a captive portal.
Policy Enforcement	Applies security policies dynamically to endpoints (e.g., VLAN assignment, ACLs).

Types of Network Access Control

NAC Type	Description	Best For
Pre-admission	Checks device before access	High-security environments
Post-admission	Monitors after access granted	Insider threat detection
Agent-based	Uses endpoint software agent	Managed endpoints (corporate laptops)
Agentless	No software; uses scanning	BYOD, IoT, guest devices
Inline	Enforces access in-path	Real-time enforcement
Out-of-Band	Makes decisions off-path	Scalable enterprise networks
Hardware-based	Dedicated appliances	On-premise, performance-critical deployments
Software-based	Virtual or cloud-deployed	Cloud-first or hybrid organizations

Examples / Use Cases

Scenario	NAC Action
Employee connects laptop to LAN port	NAC authenticates via 802.1X using AD credentials; checks AV status.
Visitor joins Wi-Fi network	Redirected to captive portal for guest registration; access is limited.
IoT device connects to network	Device is profiled and placed in IoT VLAN with minimal access.
Non-compliant machine	Redirected to quarantine VLAN with patch update instructions.

Tools (NAC Vendors & Solutions)

Tool	Description / Notes
Cisco Identity Services Engine (ISE)	Enterprise-grade NAC with deep integration into Cisco environments.
Aruba ClearPass	Powerful NAC solution, good for multi-vendor networks, BYOD, and IoT.
Forescout	Agentless visibility and control, good for unmanaged and IoT-heavy environments.
FortiNAC (Fortinet)	Integrates with Fortinet's ecosystem; offers visibility and automated response.
Microsoft NPS with 802.1X	Windows-based RADIUS/NAC server; often used in simpler deployments.
Portnox	Cloud-based NAC option for remote and hybrid environments.
Ruckus Cloudpath	Focused on secure onboarding and certificate-based NAC.
Sophos NAC (legacy)	Previously used in Sophos environments; now largely deprecated.

Network Segmentation

❑ **Network segmentation** is the practice of dividing a computer network into smaller, isolated segments (subnets, VLANs, zones) to improve **security**, **performance**, and **management**. It restricts traffic flow between parts of the network to enforce policies and minimize attack surfaces.

❑ **Purpose**

- **Limit the spread of malware or threats** (contain lateral movement)
- **Protect sensitive systems/data** by isolating them from less secure parts of the network
- **Enforce least privilege access** between systems/users
- **Improve performance and manageability** by reducing broadcast domains and traffic load

Types of Segmentation

Type	Description
Physical Segmentation	Completely separate hardware or cabling; rarely used except in high-security environments.
Logical Segmentation	Uses VLANs, subnets, and virtual interfaces over shared infrastructure.
Microsegmentation	Enforces policies at workload level (e.g., per app, VM, or container).
Application Segmentation	Allows or denies specific app-level communications regardless of IP or subnet.

Activities in Network Segmentation

Activity	Description
Define Zones	Identify and separate critical areas (e.g., HR, Finance, Production, IoT, Guest Wi-Fi).
Apply Access Control Policies	Set firewall rules or ACLs to restrict communication between zones.
Use VLANs and Subnets	Separate logical network segments using VLAN IDs and IP subnetting.
Implement Security Gateways	Use firewalls or layer 3 switches to filter traffic between segments.
Monitor Inter-Segment Traffic	Use IDS/IPS, NetFlow, or NDR to detect suspicious east-west traffic.
Microsegmentation	Apply even finer-grained segmentation at the workload or application level (e.g., per VM or container).

Examples / Use Cases

Scenario	Description
Guest Wi-Fi Segmentation	Isolate guest traffic from internal corporate network.
PCI Compliance	Isolate payment processing systems from general IT systems.
IoT Segmentation	Keep IoT devices (e.g., smart TVs, HVAC) separate from critical IT infrastructure.
Data Center Microsegmentation	Control traffic between VMs or containers at Layer 4 or Layer 7.
OT/ICS Segmentation	Protect operational technology (SCADA/PLC) from IT network threats.

Segmentation Tools & Technologies

Tool / Tech	Vendor / Use Case
VLANs	Layer 2 segmentation on switches (Cisco, Aruba, Juniper).
Subnets + ACLs	Layer 3 control, router/switch-based (Cisco, Palo Alto, etc.).
Next-Gen Firewalls	Palo Alto, Fortinet, Cisco ASA/Firepower — enforce policies between segments.
Cisco TrustSec	Role-based software-defined segmentation.
VMware NSX	Micro-segmentation inside virtualized environments.
Illumio	Agent-based micro-segmentation for cloud and hybrid.
Zero Trust Architectures	Often implement segmentation via identity and context.
SDN (e.g., Cisco ACI)	Software-defined segmentation in modern data centers.

Anomaly Detection

❑ **Anomaly detection** in cybersecurity refers to the process of identifying unusual patterns, behaviors, or activities that **deviate from the established baseline** of normal operations. These deviations may indicate a security threat, such as a cyberattack, insider threat, or system misconfiguration.

❑ **Purpose**

- **Identify novel or unknown threats** (zero-day, insider threats)
- **Detect behavioral deviations** in users, devices, or network activity
- **Enhance threat visibility** beyond traditional signature-based methods
- **Enable faster response** to suspicious activity through early warning

Activities in Anomaly Detection

Activity	Description
Baseline Profiling	Monitor and learn “normal” behavior over time (users, hosts, network traffic).
Behavior Monitoring	Continuously track activities to compare against established baselines.
Deviation Analysis	Identify patterns that fall outside expected norms (e.g., unusual login times, data access).
Alert Generation	Trigger security alerts or automated responses when anomalies are detected.
Correlation & Contextualization	Enrich anomaly data with threat intel, user identity, or system context.
Investigation & Response	Integrate with SIEM or SOAR for triage, investigation, and incident response.

Types of Anomaly Detection

Type	Detection Basis	Use Case	Pros	Cons
Statistical	Data deviations from norm	Network traffic, login attempts	Simple, fast	False positives, static
Rule-Based	Predefined rules/thresholds	Login failures, port scans	Easy to implement	Limited to known patterns
ML-Based	Learned patterns	Advanced persistent threats	Detects unknown threats	Data-heavy, opaque
Behavioral	User or entity behavior	Insider threat, data misuse	Context-aware	Needs baseline training
Network-Based	Traffic flows, protocols	Lateral movement, C2	Agentless	Limited visibility into identity
Host-Based	System/process behavior	Malware, privilege escalation	Deep endpoint insights	Requires agents
Application-Level	App usage patterns	API abuse, web attacks	App-layer protection	Needs app integration

Examples / Use Cases

Use Case	Description
User Login Anomalies	Detect logins from unusual locations, times, or devices (e.g., logins from two countries 5 minutes apart).
Lateral Movement Detection	Spot unexpected communication between devices that typically don't interact.
Data Exfiltration	Identify large or abnormal data transfers from sensitive servers or endpoints.
Compromised Account Detection	Flag users performing unusual actions like accessing high-value assets or changing permissions.
Malware Beaconing	Detect repetitive, low-frequency outbound communications to suspicious IPs.

Common Tools & Technologies

Tool / Category	Example	Description
UEBA (User and Entity Behavior Analytics)	**Microsoft Defender for Identity**, **Exabeam**, **Splunk UEBA**	Focuses on detecting abnormal user or device behavior.
NDR (Network Detection and Response)	**Darktrace**, **Vectra AI**, **ExtraHop Reveal(x)**	Analyzes network traffic for anomalies, lateral movement, or C2 behavior.
SIEM with Anomaly Detection	**IBM QRadar**, **Splunk**, **Elastic SIEM**, **Azure Sentinel**	Correlates logs and uses behavioral analytics to detect anomalies.
EDR/XDR Platforms	**CrowdStrike Falcon**, **SentinelOne**, **Microsoft Defender XDR**	Detects endpoint and system-level anomalies (processes, memory, file behavior).
AI/ML Engines	**Google Chronicle**, **Securonix**, **Devo**	Leverage ML models for high-volume anomaly detection and threat hunting.

Attack Scenario [1]

- **Case Study: Ransomware Attack via Compromised VPN Credentials**
- **Organization**
- **Industry:** Healthcare
- **Size:** ~2,000 employees
- **Network Stack:** Cisco ISE (NAC), Palo Alto Firewalls, VMware NSX (microsegmentation), Darktrace (anomaly detection), Splunk (SIEM)

Attack Scenario [2]

❑Initial Compromise

- An employee working remotely had **VPN credentials compromised** via a phishing email.
- The attacker logged into the corporate network **via VPN** using **valid credentials** — bypassing basic username/password-based access controls.

❑Detection Phase

❑**Anomaly detection** (Darktrace) flagged:

- A login from a **geographically unlikely location** (impossible travel)
- After login, the user account began **scanning internal file shares and databases** — behavior not typical for that role.
- **Lateral movement** was detected using SMB and RDP to other systems.

Response

❑Response Triggers

- **Cisco ISE (NAC)** integrated with Darktrace via pxGrid, automatically **moved the endpoint to a quarantine VLAN** using dynamic VLAN assignment.
- **Microsegmentation** via VMware NSX blocked east-west traffic between infected and uninfected systems.
- **Splunk SIEM** correlated the anomalous login, internal scanning, and endpoint alerts, escalating to the SOC team.

Incident Response Actions

Step	Action Taken
Isolate the Device	NAC moved the device to a **quarantine VLAN**, cutting off network access.
Block Lateral Movement	Microsegmentation policies dynamically **blocked internal communication**.
Disable Compromised Account	Identity team **disabled the compromised AD account**.
Log & Audit Investigation	SIEM logs were used to reconstruct the attack path and identify affected assets.
Remediation	Endpoint reimaged; MFA enforced for VPN access going forward.
Postmortem and Policy Update	Security awareness training updated; SOC tuned detection models for faster alerts.

Best Practices

Security Policies [1]

- Security policies are formal, documented rules and guidelines that define how your organization protects its information assets, responds to incidents, and manages user behavior.

Policy Type	Description
Acceptable Use Policy (AUP)	Defines permitted uses of corporate systems, devices, and networks.
Password Policy	Specifies requirements for password strength, MFA, change frequency, and reuse.
Data Classification Policy	Categorizes data (public, internal, confidential, restricted) to apply proper protections.
Access Control Policy	Enforces least privilege, RBAC, and access review protocols.

Security Policies [2]

❑ Security Policies to Implement

Policy Type	Description
Incident Response Policy	Defines roles, procedures, and SLAs for detecting, responding to, and recovering from incidents.
Bring Your Own Device (BYOD) Policy	Governs the use of personal devices on the corporate network.
Remote Work Policy	Addresses security requirements for off-site work (VPN, MFA, device hardening).
Patch Management Policy	Ensures timely updates and vulnerability remediation.
Third-Party Vendor Policy	Outlines requirements for external service providers, including due diligence and contract clauses.

Security Policy Best Practices

❑ Security Policy Best Practices

- **Align policies with frameworks**: Use NIST, ISO 27001, CIS Controls, etc.
- **Keep policies concise and understandable**: Avoid overly technical or legal language.
- **Enforce consistently**: Apply policies uniformly across all departments and users.
- **Review and update regularly**: At least annually or after significant changes.
- **Involve legal, HR, and IT**: Policies should reflect legal and operational requirements.
- **Audit compliance**: Monitor, log, and audit adherence to policies using tools and internal reviews.
- **Communicate and train**: Ensure all staff are aware of the policies and their responsibilities.

Employee Security Training [1]

- People are often the weakest link in cybersecurity. Training helps build a security-aware culture and reduces risks from phishing, social engineering, and careless behavior.

Topic	Why It Matters
Phishing Awareness	Teaches users how to spot suspicious emails, links, and attachments.
Password Hygiene	Emphasizes strong, unique passwords and use of password managers.
Social Engineering	Helps users recognize manipulation tactics (e.g., pretexting, baiting).
Device Security	Covers proper use of corporate and personal devices, screen locking, and secure disposal.

Employee Security Training [2]

Topic	Why It Matters
Remote Work Best Practices	VPN use, secure Wi-Fi, avoiding public networks.
Incident Reporting	Encourages prompt reporting of suspicious activity or potential breaches.
Data Handling & Privacy	Ensures compliance with GDPR, HIPAA, or other data regulations.
Insider Threat Awareness	Builds understanding of how insiders can compromise security (intentionally or not).

Training Program Best Practices

❑Training Program Best Practices

- **Make it role-based**: Tailor training for employees, IT staff, executives, and developers.
- **Use interactive content**: Simulations, quizzes, and gamified modules improve engagement.
- **Run phishing simulations**: Regularly test employee response to fake phishing emails.
- **Measure effectiveness**: Track completion rates, test scores, and incident reduction metrics.
- **Provide regular refreshers**: Conduct training at least annually and during onboarding.
- **Reward good behavior**: Recognize departments or users who report phishing or follow best practices.
- **Integrate with compliance requirements**: Map training to ISO, NIST, PCI-DSS, etc.

Combining Policies + Training

Weakness	Policy Response	Training Reinforcement
Phishing	Email security policy	Anti-phishing training & simulations
Weak passwords	Password policy	Password hygiene and MFA usage training
Shadow IT	Acceptable Use / BYOD policy	Training on approved tools and data risks
Insider threats	Access control, data use policy	Awareness of behavioral red flags
Remote work risks	Remote work & VPN policy	Secure home network and VPN training

Regular updates

- **Regular updates** are periodic changes made to **software, operating systems, firmware, or hardware drivers** to improve security, functionality, compatibility, or performance. These updates are typically released by software or hardware vendors and can be **automatic or manual**.

Regular updates

❑Purpose of Regular Updates

Purpose	Explanation
Enhance Security	Fix known vulnerabilities that can be exploited by cyber attackers (e.g., CVEs).
Improve Functionality	Add new features or enhance existing ones.
Fix Bugs	Resolve software errors, crashes, or performance issues.
Maintain Compatibility	Ensure interoperability with newer hardware, applications, or protocols.
Regulatory Compliance	Meet industry-specific security and privacy requirements (e.g., HIPAA, PCI DSS).

Regular updates

❑Types of Regular Updates

Type	Description
Security Updates	Address vulnerabilities (e.g., privilege escalation, remote code execution).
Feature Updates	Add new tools, services, or user interface improvements.
Stability Fixes	Correct bugs that may cause crashes or performance drops.
Driver/Firmware Updates	Improve compatibility and fix hardware-related bugs.
Service Packs / Cumulative Updates	Bundle multiple fixes and enhancements into one package.

Regular updates

❑Tools for Managing Regular Updates

Platform	Tool	Description
Windows	WSUS, SCCM, PDQ Deploy	Centralized control of Windows updates
macOS	Jamf, Munki	macOS patch and update management
Linux	Unattended-upgrades, YUM/DNF, APT	CLI-based or automated update managers
Cloud	AWS Systems Manager, Azure Update Manager	Patches VMs and services in cloud environments
Cross-platform	ManageEngine, Ivanti, Automox	Unified patch management for all OS types

Patch Management

- **Patch Management** is the process of **identifying, evaluating, acquiring, testing, and deploying software updates (patches)** to fix vulnerabilities, improve performance, and maintain the security of systems, applications, and devices.

Patch Management

❑Purpose of Patch Management

Purpose	Explanation
Security	Fix vulnerabilities before attackers exploit them (e.g., zero-days, CVEs).
System Stability	Prevent system crashes and improve operational reliability.
Performance Improvement	Enhance speed and responsiveness of software or hardware.
Compatibility	Ensure compatibility with other systems, software, and updates.
Regulatory Compliance	Meet legal/industry standards (e.g., HIPAA, PCI-DSS, ISO 27001).

Patch Management

❑ Activities in Patch Management

Activity	Details
Asset Inventory	Identify all systems, devices, and software needing patches.
Patch Notification	Subscribe to vendor bulletins or vulnerability feeds (e.g., NIST, CVE).
Testing	Test patches in staging environments to ensure compatibility.
Prioritization	Use CVSS scores and asset criticality to determine patching urgency.
Deployment	Deploy patches to production systems using automated or manual methods.
Verification	Confirm successful patch application and system functionality.
Documentation	Record patch details for auditing, tracking, and compliance.

Patch Management

❑Common Tools for Patch Management

Tool	Platform	Description
WSUS (Windows Server Update Services)	Windows	Manages updates for Microsoft products across an enterprise.
SCCM / MECM (System Center Configuration Manager)	Windows	Advanced patch and configuration management.
ManageEngine Patch Manager Plus	Cross-platform	Automates patching for Windows, Linux, macOS, and third-party apps.
Ivanti Patch Management	Cross-platform	Offers patch automation and vulnerability management.

Backup Strategies

- A **backup strategy** is a structured plan that defines **how, when, where, and what data is backed up**, in order to ensure that information can be restored in case of data loss, corruption, or disaster.
- **It includes:**
  - Backup frequency
  - Backup type
  - Storage location
  - Retention policies
  - Security controls

Backup Strategies

❑Purpose of a Backup Strategy

Objective	Explanation
Data Protection	Prevent permanent data loss due to accidental deletion or hardware failure.
Security Against Ransomware	Recover encrypted data without paying ransom.
Business Continuity	Ensure fast recovery of data and services during outages.
Compliance & Legal	Meet regulatory requirements (e.g., HIPAA, GDPR, ISO 27001).
Operational Resilience	Maintain productivity and reduce downtime.

Types of Backup

Type	Description
Full Backup	Copies **all data**. Easy to restore but slow and requires large storage.
Incremental Backup	Backs up **only data changed since the last backup** (full or incremental). Fastest, but restores take longer.
Differential Backup	Backs up **all data changed since the last full backup**. Restoration is faster than incremental.
Mirror Backup	Creates an exact replica. No versioning; if a file is deleted, it's deleted in the mirror too.
Image-Based Backup	Creates a snapshot of the entire system, including OS and configuration, for full system restore.
Cloud Backup	Stores data in the cloud (e.g., AWS S3, Google Cloud Storage, Azure). Offers scalability and offsite redundancy.
Hybrid Backup	Combines on-premises (local) and cloud backups for performance and resilience.
Continuous Data Protection (CDP)	Backs up data **in real time or near real time** to prevent loss between scheduled backups.

Backup Storage Options

Storage Type	Pros	Cons
Local (external drives, NAS)	Fast access, low cost	Risk of loss due to disasters or theft
Offsite physical (tape vaulting)	Good for long-term archival	Slow to restore, logistical challenges
Cloud storage	Scalable, accessible anywhere, offsite	Ongoing costs, depends on internet
Hybrid (local + cloud)	Combines speed of local with safety of cloud	Complex setup

Examples of Backup Strategies

- ❑ **3-2-1 Backup Strategy *(industry best practice)***
- ❑ **3** total copies of data
- ❑ **2** different storage media (e.g., disk + cloud)
- ❑ **1** offsite copy (e.g., cloud, remote data center)

Backup Tools & Solutions

Tool/Service	Key Features	Target
Veeam Backup & Replication	Full suite for VMs, cloud, physical	Enterprise
Acronis Cyber Protect	Backup + anti-malware	SMBs & enterprises
Backblaze	Simple and affordable cloud backup	Personal/SMBs
IDrive	Cloud backup with versioning	Personal/Business
AWS Backup / S3 / Glacier	Scalable cloud-based storage	Cloud-based workloads
Azure Backup	Integrated with Microsoft environments	Azure cloud, hybrid
Google Backup and DR	Backup for Google Cloud and hybrid systems	GCP & multi-cloud
Rubrik / Cohesity	Enterprise-grade, automation-focused	Large orgs/cloud-native

Disaster Recovery

- **Disaster Recovery (DR)** is a **set of policies, procedures, and tools** that enable the **restoration of critical IT systems, applications, and data** after a disruptive event such as:
  - Cyberattacks (e.g., ransomware)
  - Hardware/software failures
  - Natural disasters (e.g., fire, flood, earthquake)
  - Human errors or sabotage
  - Power outages

Disaster Recovery

❑ Purpose of Disaster Recovery

Goal	Explanation
Minimize Downtime	Quickly restore services and operations after a failure or attack.
Protect Critical Data	Recover lost or corrupted data.
Meet RTO and RPO	Ensure acceptable recovery time and data loss limits are met.
Maintain Business Continuity	Keep essential functions running during and after a disaster.
Ensure Compliance	Meet industry or legal requirements (e.g., HIPAA, GDPR, ISO 22301).
Mitigate Financial Loss	Reduce impact from outages and loss of revenue or reputation.

Types of Disaster Recovery

Type	Description
Cold Site	Basic infrastructure (power, space) only — systems must be set up from scratch.
Warm Site	Pre-installed hardware and OS — apps/data must still be restored.
Hot Site	Fully redundant systems — real-time replication; instant failover possible.
Disaster Recovery as a Service (DRaaS)	Cloud-based recovery of infrastructure/applications on demand.

Disaster Scenarios & Response Examples

Scenario	Response
Ransomware locks all file servers	Failover to clean DR environment using last known good backup
Fire damages on-prem data center	Spin up systems from hot site or DRaaS in the cloud
Cloud outage in primary region	Redirect traffic to secondary region via DNS or load balancer failover
Employee deletes critical database	Restore from backup within RPO window

Common Tools & Solutions for DR

Tool / Service	Purpose	Environment
Veeam Backup & Replication	Replication and DR orchestration	Physical, virtual, cloud
Acronis Cyber Protect	Backup, DR, and endpoint protection	Cross-platform
Zerto	Real-time replication and DR automation	Virtualized environments
Azure Site Recovery	DRaaS for Azure and on-prem systems	Microsoft / hybrid cloud
AWS Elastic Disaster Recovery (AWS DRS)	DRaaS for AWS & on-prem workloads	AWS
Commvault	Full-scale enterprise backup & DR	Data center & cloud
VMware Site Recovery Manager (SRM)	Disaster recovery for VMware vSphere environments	Virtualized infrastructure


Q&A