📘 INTRODUCTION TO CYBER SECURITY

Definition

Cyber Security refers to the practice of protecting computer systems, networks, digital devices, and data from unauthorized access, attacks, damage, or theft.
It ensures **confidentiality, integrity, and availability (CIA)** of information in the digital world.

In simple words:
➡️ *Cyber Security is the shield that protects digital systems from hackers and cyber threats.*

Key Features / Characteristics

1. Confidentiality

Only authorized users can access sensitive information.

2. Integrity

Information must remain accurate, unaltered, and trustworthy.

3. Availability

Systems and data should be accessible whenever needed.

4. Authentication

Verifying the identity of users (e.g., passwords, biometrics).

5. Authorization

Defining what each authenticated user can access or do.

6. Non-Repudiation

Ensuring that actions (transactions, messages) cannot be denied later.

7. Security Controls

Technical (firewalls), administrative (policies), and physical (CCTV).

8. Incident Response

Proper handling of security breaches and minimizing impact.

Advantages of Cyber Security

- Protects data from theft, loss, and unauthorized access.
- Prevents financial loss due to cyberattacks like ransomware.
- Builds trust for businesses (customers feel safe).
- Ensures business continuity and avoids downtime.
- Protects national security and critical infrastructure.

- Helps organizations comply with security laws (GDPR, IT Act).
- Reduces risks from emerging threats such as IoT attacks or AI-driven attacks.

Disadvantages / Limitations

- High cost of implementing advanced security systems.
- Requires continuous monitoring and updates.
- Skilled cybersecurity professionals are needed (skill shortage).
- Strong security may cause inconvenience (multi-factor authentication).
- Not 100% foolproof—no system is completely safe.
- May lead to slower system performance due to encryption or scanning.

Real-World Examples

✔ **Ransomware Attack (WannaCry 2017)**

Encrypted data of millions of devices worldwide; demanded payment.

✔ **Data Breach (Facebook–Cambridge Analytica)**

User data misused for political advertising.

✔ **Banking Malware (Zeus Trojan)**

Stole online banking credentials from users globally.

Importance of Cyber Security

Cyber Security is vital because almost every activity today depends on digital systems—banking, healthcare, education, government services, e-commerce, and personal communication.

1️⃣ **Protection of Sensitive Data**

Prevents unauthorized access to personal information, financial data, business secrets, and government records.

2️⃣ **Prevents Financial Loss**

Cyberattacks like ransomware, banking malware, and phishing cause massive monetary damage.
Strong security reduces these risks.

3️⃣ **Ensures Business Continuity**

Attackers can shut down business operations.
Cybersecurity ensures systems run smoothly without interruptions.

4️⃣ Maintains Trust and Reputation

Customers trust businesses that protect their data.
Loss of data → loss of customers → loss of brand value.

5️⃣ Secures Critical Infrastructure

Includes power grids, hospitals, airports, water systems—attacks can cause national-level disasters.

6️⃣ Protects National Security

Government institutions handle military and intelligence data.
Cyber warfare has become a real threat.

7️⃣ Supports Safe Digital Transformation

Organizations shift to cloud, IoT, AI-based solutions.
Security ensures safe adoption of new technologies.

8️⃣ Helps Comply with Laws & Regulations

Cybersecurity measures help organizations follow regulations like GDPR, HIPAA, IT Act, etc.

Challenges in Cyber Security

Despite advanced technologies, cyber security faces many challenges.

🔹 1. Increasing Number & Sophistication of Attacks

Attackers use new, complex methods like AI-driven attacks, zero-day exploits, ransomware-as-a-service.

🔹 2. Shortage of Skilled Cybersecurity Experts

There is a huge global gap in trained professionals.
Organizations struggle to find qualified security analysts.

🔹 3. Rapid Technological Changes

New technologies like **IoT, Cloud, Blockchain, AI** introduce new vulnerabilities faster than security can adapt.

🔹 4. Human Errors & Lack of Awareness

Employees often fall for phishing emails, weak passwords, misconfigurations, or accidental data leaks.

🔹 **5. Complex & Expanding Attack Surface**

More devices = more entry points.
Examples: phones, routers, smart TVs, IoT sensors, cloud accounts.

🔹 **6. Advanced Persistent Threats (APTs)**

Long-term, targeted attacks mostly organized by nation-states.
Very hard to detect and defend.

🔹 **7. Insider Threats**

Employees or contractors may misuse data intentionally or accidentally.

🔹 **8. Difficulty in Detecting Zero-Day Vulnerabilities**

Software bugs unknown to vendors can be exploited anytime.

🔹 **9. High Cost of Security Implementation**

Firewalls, SOCs, skilled professionals, and tools require heavy investment.

🔹 **10. Maintaining Privacy in a Connected World**

Data collection by apps and organizations increases privacy risks.

📘 CYBERSPACE

Definition

Cyberspace refers to the virtual environment created by interconnected computers, networks, software, the internet, digital devices, and data.
It is the **digital world** where online communication, transactions, social interaction, data exchange, and information processing take place.

In simple words:
➡️ *Cyberspace is the "world of the internet" where people, devices, and systems interact digitally.*

Key Characteristics / Features of Cyberspace

1️⃣ **Interconnected Network**

Billions of devices (phones, laptops, servers, IoT devices) connected globally.

2️⃣ **Virtual Environment**

It exists digitally—not physically—created through data and networking.

3️⃣ Global Accessibility

Anyone with internet access can interact in cyberspace from anywhere.

4️⃣ No Physical Boundaries

Unlike the real world, cyberspace has no geographical limitations.

5️⃣ High Speed Communication

Information travels instantly across networks.

6️⃣ Multiple Modes of Interaction

Messaging, emails, video calls, social media, cloud computing, online games.

7️⃣ Data-Driven

All activities (transactions, communication, browsing) rely on data exchange.

8️⃣ Dynamic & Ever-Changing

New technologies, platforms, threats, and users join every day.

Importance of Cyberspace

- Enables digital communication across the world.
- Supports e-commerce, online banking, digital payments.
- Provides platforms for social networking and collaboration.
- Powers cloud services, data storage, AI, IoT, and digital services.
- Critical for government operations, defense, healthcare, and education.
- Essential for the digital economy and global business operations.

Advantages of Cyberspace

- Enhances communication and connectivity.
- Supports online education and remote work.
- Helps businesses expand globally.
- Makes digital transactions faster and easier.
- Provides access to huge amounts of information.
- Allows real-time collaboration between people worldwide.

Disadvantages / Issues in Cyberspace

- Exposure to cybercrimes and cyberattacks.
- Privacy invasion due to data misuse.
- Spread of misinformation or fake news.
- Online addiction and psychological issues.
- Cyberbullying and harassment.
- Lack of global laws for cyber governance.
- Difficult to track criminals due to anonymity.

Real-World Examples of Cyberspace Activities

✔ **Google Search**

User sends a request → servers process → webpage delivered digitally.

✔ **Online Banking**

Transactions occur entirely in cyberspace.

✔ **Social Media Platforms (Instagram, Facebook, Twitter)**

All communication and data-sharing happen virtually.

✔ **Cloud Storage (Google Drive, Dropbox)**

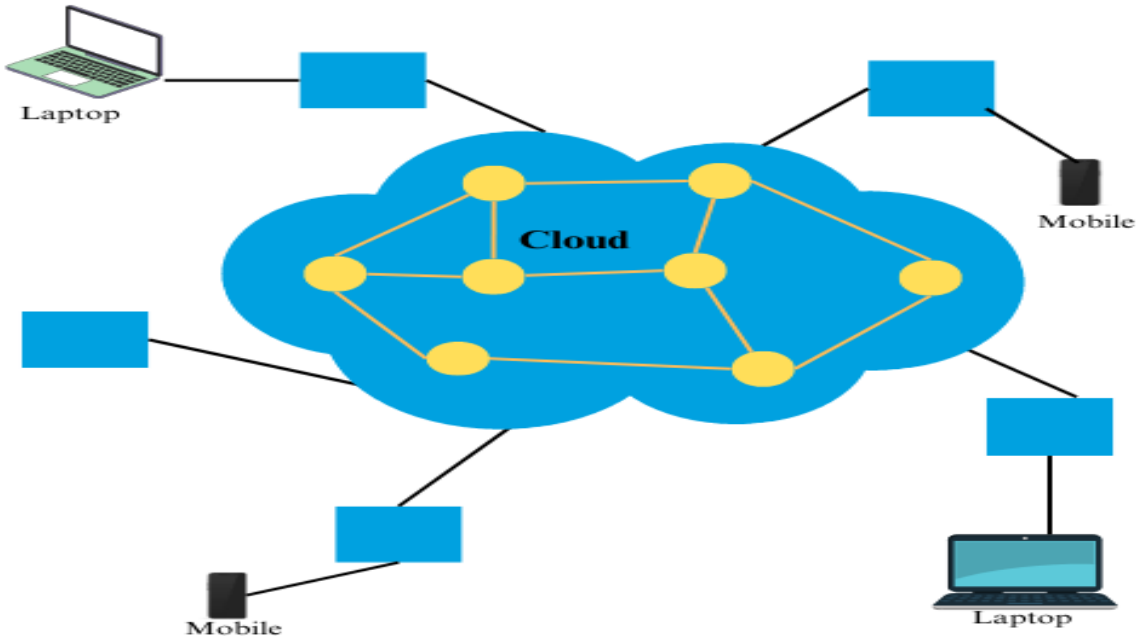
Data stored in cyberspace instead of local devices.

✔ **Online Gaming**

Players interact in a shared virtual environment

Q. **Why is cyberspace vulnerable to cyber threats?**

Answer:
Cyberspace is open, interconnected, and globally accessible, which creates many entry points for hackers.
Anonymity allows attackers to hide their identity.
Weak security practices and lack of international regulations also increase vulnerability.



📘 CYBER THREATS

Definition

A **cyber threat** is any harmful act that exploits vulnerabilities in computer systems, networks, or digital devices to steal data, damage systems, disrupt services, or gain unauthorized access.

In simple words:
➡️ *A cyber threat is a potential danger that can harm or attack digital systems.*

Why Cyber Threats Occur? (Causes)

- Financial gain
- Stealing confidential data
- Political motives (hacktivism)
- Cyber warfare by nations
- Revenge or personal motives
- Fun / challenge (script kiddies)
- Competition between businesses

Types of Cyber Threats (Detailed)

1️⃣ Malware (Malicious Software)

Software designed to damage or disrupt systems.

Examples:

Virus, Worm, Trojan, Spyware, Adware.

Key Features:

- Hides inside files
- Spreads from one system to another
- Can steal, delete, or modify data

2️⃣ Ransomware

Encrypts files and demands payment (ransom) for decryption.

Examples:

WannaCry, Locky, Petya.

Impact:

Hospitals, banks, and companies have faced shutdowns for days.

3️⃣ Phishing Attacks

Attackers send fake emails/messages pretending to be legitimate to steal passwords or banking info.

Example:

Fake bank email asking to “verify your account.”

4️⃣ Denial of Service (DoS) / DDoS Attack

Attackers overload a server with heavy traffic, causing shutdown.

Example:

DDoS on government website, preventing access.

5️⃣ Man-in-the-Middle (MitM) Attack

Attackers secretly intercept communication between two parties.

Example:

Intercepting data on public Wi-Fi.

6️⃣ SQL Injection Attack

Attackers exploit vulnerabilities in a database query to view or delete sensitive data.

Example:

“' OR '1'='1” bypasses login authentication.

7️⃣ Zero-Day Exploits

Attack exploiting an unknown software vulnerability before developers fix it.

Example:

Stuxnet worm used zero-day flaws in Windows.

8️⃣ Social Engineering

Manipulating people to reveal confidential information.

Examples:

Fake tech support calls, impersonation, tailgating.

9️⃣ Insider Threats

Threats from employees, contractors, or partners with access to systems.

Types:

- Malicious insider
- Negligent insider
- Compromised insider

🔟 Password Attacks

Attackers try to guess or crack passwords.

Methods:

- Brute force
- Dictionary attack
- Credential stuffing

1️⃣1️⃣ Cyber Espionage

Nation-states spying on other countries or organizations.

Example:

APT (Advanced Persistent Threat) groups like APT28, APT29.

1️⃣2️⃣ Botnets

Networks of infected devices controlled by a hacker.

Example:

Mirai botnet used millions of IoT devices for DDoS attacks.

Characteristics of Cyber Threats

- Hidden and difficult to detect
- Automated and fast-spreading
- Can be remote (global attacks)
- Exploit system vulnerabilities
- Target individuals, companies, governments
- Highly damaging and costly

Impact (Consequences) of Cyber Threats

- Data loss or corruption
- Financial loss
- Service downtime
- Damage to reputation
- Loss of customer trust
- Leakage of intellectual property
- National security risks
- Legal issues and penalties

Real-World Examples

- **WannaCry Ransomware (2017):** Hit 150+ countries & hospitals.
- **Equifax Data Breach (2017):** 147 million users affected.
- **Sony Pictures Hack (2014):** Data stolen due to political reasons.
- **Twitter Bitcoin Scam (2020):** Insider gained access & posted fake tweets.
- **Cambridge Analytica:** Misuse of personal data on Facebook.

Advantages & Disadvantages of Studying Cyber Threats

Advantages:

- Helps organizations prepare strong defenses.
- Improves risk management strategies.

- Enhances awareness and employee training.
- Protects businesses from financial and data loss.

Disadvantages:

- Constant need for updates and monitoring.
- Cyber attackers evolve faster than defenses.
- Requires skilled professionals.
- Not possible to eliminate threats completely.

📘 SQL Injection

SQL Injection (SQLi) is a cyberattack where an attacker inserts (injects) malicious SQL code into an input field (like login forms, search boxes, URL parameters) to manipulate a website's database.

In simple words:
➡️ *SQL Injection allows attackers to trick the database into revealing, modifying, or deleting data by entering malicious code.*

How SQL Injection Works (Simple Explanation)

1. A website takes input from the user (username, search query).
2. This input is included directly in an SQL query.
3. If the input is not validated or filtered, attackers insert harmful SQL code.
4. The database executes this malicious code.
5. Attackers gain unauthorized access or control.

Types of SQL Injection

1️⃣ In-band SQL Injection (Classic)

Attacker uses the same communication channel to attack and receive results.

Types:

- **Error-based SQLi:** Uses database error messages to extract information.
- **Union-based SQLi:** Uses UNION SQL operator to fetch additional data.

2️⃣ Inferential SQL Injection (Blind SQLi)

No direct output. Attacker infers data by analyzing website behavior.

Types:

- **Boolean-based Blind SQLi:** True/false responses reveal information.
- **Time-based Blind SQLi:** Database delays (sleep function) reveal information.

3️⃣ Out-of-band SQL Injection

Data is retrieved through different channels (like DNS or email).
Used when direct output is blocked.

What Attackers Can Do with SQL Injection

- Bypass login authentication
- View sensitive data (passwords, emails, credit card numbers)
- Modify or delete database records
- Gain admin access
- Take full control of the server
- Run system commands (advanced SQLi)
- Destroy entire databases

SQL Injection is one of the **top web vulnerabilities (OWASP Top 10).**

Real-World SQL Injection Incidents

✓ Sony Pictures Attack (2011)

Millions of user accounts leaked using SQL Injection.

✓ LinkedIn SQLi Vulnerability (2012)

Passwords of 6.5 million users stolen.

✓ Microsoft SQL Server Attacks

Thousands of websites defaced using automated SQLi bots.

📘 CYBERWARFARE

Cyberwarfare refers to the use of digital attacks by one nation/state (or state-sponsored groups) to damage, disrupt, or gain strategic advantage over another nation's computer systems, networks, or infrastructure.

In simple words:
➡️ *Cyberwarfare is war carried out in the digital world instead of physical battlefields.*

It targets:

- Military systems

- Government networks
- Power grids
- Banking systems
- Communication networks
- Transportation systems
- Critical infrastructure

Key Characteristics / Features

1 State-Sponsored

Carried out by governments or military agencies, not individuals.

2 Targets Critical Infrastructure

Power plants, gas pipelines, satellites, government servers.

3 Stealthy & Hidden

Attacks occur without physical presence; often undetected for long periods.

4 Highly Sophisticated

Utilizes zero-day exploits, malware, APTs (Advanced Persistent Threats).

5 Can Cause Real-World Damage

Disrupting power, hospital systems, and military operations.

6 Espionage & Surveillance

Stealing classified information, spying on other nations.

7 Offensive & Defensive Actions

Countries both **attack** and **defend** in cyberspace.

Types of Cyberwarfare

1 Espionage (Cyber Spying)

Stealing confidential military or political data.

Example: APT29 targeting government agencies.

2 Sabotage

Destroying or disabling critical infrastructure.

Example: Attacking power grids or nuclear plants.

3️⃣ Propaganda / Psychological Warfare

Spreading misinformation to influence public opinion.

Example: Fake news campaigns during elections.

4️⃣ Denial-of-Service Attacks

Shutting down government or military websites by massive traffic.

5️⃣ Cyber Terrorism

Digitally attacking systems to create panic or fear.

6️⃣ Data Manipulation

Changing data to mislead or disrupt military operations.

7️⃣ Attacks on Command and Control Systems

Targeting enemy communication networks to disrupt coordination.

Major Real-World Examples of Cyberwarfare

- **Stuxnet (2010)**
  - Most famous cyberwar attack.
  - Computer worm created by the US & Israel.
  - Targeted Iran's nuclear program and destroyed centrifuges.
  - Physical damage done through malware.
- **Russia–Estonia DDoS Attack (2007)**
  - Massive attack on Estonian government, media, and banking sites.
  - First large-scale cyberattack on a nation.
- **Ukraine Power Grid Attack (2015 & 2016)**
  - Malware "BlackEnergy" shut down electricity for hundreds of thousands.
  - Believed to be carried out by Russian state-sponsored hackers.

Goals of Cyberwarfare

- National security advantage
- Disrupt enemy communication
- Gather intelligence

- Destroy critical infrastructure
- Economic damage
- Influence political outcomes
- Military superiority

Advantages & Disadvantages

Advantages (from attacker perspective)

- No physical soldiers needed
- Low cost compared to traditional warfare
- Hard to trace the attacker
- Global impact with small resources
- Can disable enemy quickly

Disadvantages

- Ethical and legal issues
- Can escalate to full-scale war
- Civilians may be harmed indirectly
- Risk of backfiring if malware spreads globally
- Requires very high technical expertise

📘 CIA Triad

The **CIA Triad** is the core model of information security.
It stands for:

- **C – Confidentiality**
- **I – Integrity**
- **A – Availability**

These three principles ensure data is protected from unauthorized access, alteration, and unavailability.

In simple words:
➡️ *CIA Triad protects data privacy, correctness, and accessibility.*

Definition

The **CIA Triad** is a fundamental security framework that guides policies and controls to safeguard information.
It ensures that data is **kept secret (Confidentiality)**, **accurate (Integrity)**, and **accessible (Availability)**.

Components of CIA Triad (Detailed)

1️⃣ Confidentiality (C)

Ensures that information is accessible **only to authorized people**.

Goal: Prevent unauthorized access, viewing, copying, or sharing.

Methods to achieve Confidentiality:

- Encryption
- Strong passwords
- Multi-factor authentication (MFA)
- Access control lists (ACLs)
- Biometrics
- Data classification
- Firewalls

Example:

- ATM PIN ensures only you can access your bank account.
- Encrypting emails so only the receiver can read them.

2️⃣ Integrity (I)

Ensures data is **accurate, consistent, complete, and unaltered** unless modified by authorized users.

Goal: Protect data from unauthorized modification.

Methods to achieve Integrity:

- Hashing (SHA-256, MD5)
- Digital signatures
- Checksums
- Version control systems (Git)
- Backups
- Input validation

Example:

- Hash values verifying that a file wasn't tampered with.
- Digital signatures ensure messages are not altered in transit.

3️⃣ Availability (A)

Ensures data, systems, and services are **accessible whenever needed**.

Goal: Prevent downtime or service unavailability.

Methods to achieve Availability:

- Redundancy (backup servers, alternate power)
- Load balancing
- Regular system maintenance
- DDoS protection
- Disaster recovery plans
- Failover systems

Example:

- Google Search stays online even during heavy traffic due to load balancing.
- Hospitals must have 24/7 server availability.

Why the CIA Triad Is Important

- Forms the foundation of cybersecurity policies.
- Helps design secure systems and networks.
- Balances data security with functionality.
- Used globally in security audits, risk management, and compliance.

- **Confidentiality violation → Data breach exposing passwords**
- **Integrity violation → A hacker altering bank account balances**
- **Availability violation → DDoS attack taking a website offline**

📘 CYBER TERRORISM

Cyber Terrorism refers to the use of internet-based attacks by terrorists to create fear, cause disruption, damage critical infrastructure, steal sensitive information, or harm national security.

In simple words:
➡️ *Cyber Terrorism is terrorism carried out using computers, networks, and the internet instead of physical weapons.*

It aims to:

- Create panic
- Cause economic or social disruption
- Attack critical systems
- Spread propaganda
- Influence political decisions

Key Features / Characteristics

1️⃣ Intent to Create Fear or Panic

Goal is psychological impact, not just system damage.

2️⃣ Politically or Ideologically Motivated

Groups use cyberattacks to promote their beliefs.

3️⃣ Targeting Critical Infrastructure

Power grids, hospitals, transportation, banking systems.

4️⃣ Use of Highly Skilled Attackers

Cyber-terrorist groups have experts in hacking, encryption, and malware.

5️⃣ Anonymous & Remote Attacks

Difficult to trace due to global internet.

6️⃣ Low-Cost, High-Impact Attacks

Unlike physical terrorism, cyber terrorism can be done with minimal resources.

Objectives of Cyber Terrorism

- Disrupt government operations
- Destroy critical services
- Steal military or intelligence data
- Spread fear among the public
- Conduct digital propaganda
- Financial disruption (banks, stock markets)
- Recruit or radicalize individuals online
- Attack national infrastructure
- Influence political outcomes

Common Cyber Terrorism Techniques

1️⃣ Distributed Denial of Service (DDoS)

To shut down government or emergency websites.

2️⃣ Malware Attacks

Viruses, worms, ransomware targeting essential services.

3️⃣ Website Defacement

Replacing government websites with threatening messages.

4️⃣ Cyber Espionage

Stealing military or intelligence information.

5️⃣ Social Engineering & Phishing

Accessing sensitive systems or accounts.

6️⃣ **Cyber Propaganda**

Using social media to spread extremist ideologies.

7️⃣ **Attacks on SCADA / ICS**

Industrial systems controlling power, oil, water plants.

Real-World Examples

🔹 **1. ISIS Cyber Activities**

- Used social media for recruitment and propaganda.
- Hacked U.S. Central Command's Twitter account.

🔹 **2. Stuxnet (2010)**

- Although state-sponsored, it demonstrated how malware can damage nuclear facilities.
- Inspired terrorist groups to try similar attacks.

🔹 **3. 2007 Estonia Cyber Attack**

- DDoS attacks shut down government and banking services.
- Created national panic.

🔹 **4. Attack on Mumbai Electricity Grid Attempt (2020)**

- Suspected hacking attempt linked to foreign actors.
- Caused partial blackouts in Mumbai.

📘 CYBER SECURITY OF CRITICAL INFRASTRUCTURE

Critical Infrastructure (CI) refers to essential systems and assets whose destruction or disruption would severely impact national security, economy, public health, or safety.

Cyber Security of Critical Infrastructure is the practice of protecting these essential services from cyberattacks, ensuring they remain safe, reliable, and operational at all times.

In simple words:
➡️ *Protecting vital systems like power grids, water supply, transportation, defense, and hospitals from cyberattacks.*

Examples of Critical Infrastructure

- **Energy:** Power grids, nuclear plants, oil and gas pipelines
- **Transportation:** Railways, airports, metro, traffic control systems
- **Healthcare:** Hospitals, emergency response systems

- **Government:** Military networks, police databases
- **Communication:** Internet backbone, telecom networks
- **Water systems:** Water treatment plants, dams
- **Banking and Finance:** Online banking, ATMs, stock markets
- **Industrial Infrastructure:** SCADA and ICS systems in factories

Why Cybersecurity is Critical for These Systems?

1️⃣ National Security

A cyberattack can shut down military communication or cause power blackouts.

2️⃣ Public Safety

Hospitals, traffic systems, and water plants must run 24/7.

3️⃣ Economic Stability

Financial markets and banks depend heavily on secure digital systems.

4️⃣ Dependency on Technology

Most critical infrastructures use digital control systems like **ICS** (Industrial Control Systems) and **SCADA**.

5️⃣ Increasing Cyber Threats

State-sponsored attacks and terrorist activities are rising.

Components in Critical Infrastructure Security

1️⃣ ICS (Industrial Control Systems)

Used to monitor and control industrial operations.

2️⃣ SCADA (Supervisory Control and Data Acquisition)

Controls large-scale operations like power plants, pipelines.

3️⃣ PLC (Programmable Logic Controller)

Automates mechanical processes in industries.

These systems were originally designed for **functionality**, not **security**, making them vulnerable today.

Major Cyber Threats to Critical Infrastructure

1. Malware & Ransomware

Targets power plants, hospitals, industrial control systems.

2. DDoS Attacks

Overloads online services like emergency response networks.

3. Insider Threats

Employees with access may leak or misuse information.

4. Zero-Day Attacks

Unknown vulnerabilities in SCADA/ICS software.

5. Phishing / Social Engineering

Attackers trick employees into giving access.

6. SCADA-Specific Attacks

Like Stuxnet targeting centrifuge systems.

7. Supply Chain Attacks

Infecting software updates (e.g., SolarWinds attack).

Challenges in Securing Critical Infrastructure

- Legacy systems without modern security
- High dependency on SCADA/ICS for offline operations
- Lack of skilled cybersecurity professionals
- Complex networks with many vulnerabilities
- Remote operations increase entry points
- Governments and private sectors must coordinate
- Cyberattacks becoming more advanced (APT groups)

Security Measures to Protect Critical Infrastructure

1 Network Segmentation

Separate IT network and OT (operational technology) network.

2 Strong Authentication & Access Control

Least privilege principle.

3 Intrusion Detection & Monitoring

IDS, IPS, SIEM systems for real-time alerts.

4 Patch Management & Updates

Regular updates to remove vulnerabilities.

5️⃣ **Encryption of Communication**

Protects data in transit.

6️⃣ **Incident Response Plan**

Quick action during breaches to reduce impact.

7️⃣ **Backup & Disaster Recovery**

Ensures continuity during attacks.

8️⃣ **WAF & Firewalls**

Protect entry points and block malicious traffic.

9️⃣ **Employee Training**

To avoid phishing and social engineering attacks.

🔟 **Government Policies & National CERTs**

NCIIPC (India), CISA (USA), ENISA (EU).

Advantages of Securing Critical Infrastructure

- Protects national security
- Ensures smooth functioning of essential services
- Reduces economic losses
- Prevents large-scale panic
- Saves lives during emergencies
- Maintains trust in government & service providers

Disadvantages / Limitations

- High cost of specialized security
- Requires trained experts
- Difficult to upgrade legacy systems
- Threats evolve faster than defenses
- Hard to coordinate between govt & private sector

📘 CYBERSECURITY – ORGANIZATIONAL IMPLICATIONS

Organizational Implications of Cybersecurity refers to the effects, responsibilities, changes, and challenges that cybersecurity brings to an organization.

It includes how cybersecurity influences business operations, employees, finances, reputation, compliance, and long-term strategy.

In simple words:
➡️ *It means how cybersecurity impacts an organization's functioning, decision-making, and overall safety.*

Why Cybersecurity Matters to Organizations

Organizations depend on digital systems for:

- Customer data
- Financial transactions
- Online services
- Cloud platforms
- Internal communication
  Any cyberattack can disrupt the entire business.

Key Organizational Implications

1️⃣ Financial Impact

Cyberattacks result in:

- Loss of revenue
- Cost of restoring systems
- Legal penalties
- Compensation to customers
- Investment in new cybersecurity tools

Example: Ransomware can cause crores-worth losses

2️⃣ Operational Disruption

Cyber incidents can shut down operations (manufacturing, banking, transportation).

Example:
A DDoS attack can shut down an organization's website temporarily.

3️⃣ Damage to Reputation

Loss of customer trust leads to:

- Lower sales
- Bad publicity

- Long-term brand damage

Example: Facebook–Cambridge Analytica data misuse incident.

4️⃣ Legal & Regulatory Compliance

Organizations must comply with laws like:

- **GDPR**
- **IT Act (India)**
- **HIPAA**
- **ISO 27001**
- **PCI-DSS**

Non-compliance → penalties + lawsuits.

5️⃣ Strategic & Management Changes

Cybersecurity becomes part of:

- Business strategy
- Risk management
- Board-level decisions
- Budget planning
- Vendor management

6️⃣ Need for Skilled Workforce

Organizations must hire:

- Security analysts
- Network security engineers
- Incident response teams
- SOC (Security Operations Center) members

Skill shortage increases costs.

7️⃣ Organizational Culture Shift

Cybersecurity awareness becomes mandatory for all employees.
Training programs must be conducted regularly.

8️⃣ Supply Chain & Third-Party Risks

A weak vendor can expose the entire organization.

Example: SolarWinds supply-chain attack (2020).

9️⃣ Increased Use of Technology & Tools

Organizations must adopt:

- Firewalls
- Endpoint security
- DLP systems (Data Loss Prevention)
- Encryption
- SIEM (Security Information & Event Management)
- Zero-Trust Architecture

🔟 Data Privacy Requirements

Organizations must protect customer and employee data.
Misuse or theft → legal consequences + loss of trust.

Positive Implications (Benefits)

- Better risk management
- Stronger internal controls
- Higher customer trust
- Competitive advantage ("We are secure")
- Improved business continuity
- Compliance with global standards

Negative Implications (Challenges)

- Increased cost of cybersecurity tools
- Need for trained professionals
- Complex security policies
- Interruption during security audits
- Frequent updating of systems
- Requires cultural change

📘 HACKERS

A **Hacker** is an individual who uses technical knowledge, programming skills, and understanding of computer networks to access systems or data—either legally (**ethical hacking**) or illegally (**malicious hacking**).

In simple words:
➡️ *A hacker is someone who finds weaknesses in computer systems.*

Hackers can be **good**, **bad**, or **neutral**, depending on their motives.

Key Characteristics of Hackers

- Expert knowledge of computers & networks
- Understand vulnerabilities, security gaps
- Strong problem-solving skills
- Use programming languages (Python, C, Java, JS)
- Skilled at penetration testing, exploiting bugs
- Creative and curious mindset
- Can work anonymously

Positive Roles of Hackers

- Strengthening cybersecurity
- Bug bounty programs
- Penetration testing
- Securing critical infrastructure
- Discovering zero-day vulnerabilities
- Protecting data and networks

Negative Roles of Hackers

- Data breaches
- Financial fraud
- Identity theft
- Cyber terrorism
- Ransomware attacks
- Espionage
- Reputation damage

- System shutdowns

📘 CRACKERS

Both **Hackers** and **Crackers** deal with computer systems and software, but their intentions and actions are very different.

- **Hackers** → Can be ethical or unethical
- **Crackers** → Always malicious; break security and software protections

Understanding the difference is crucial in cybersecurity.

A **Cracker** is a malicious individual who breaks into systems, software, or networks **with the intention to steal, damage, or bypass security protections**, especially **software piracy**.

In simple words:
➡️ *Crackers break security systems purely for illegal purposes.*

Common activities of Crackers:

- Breaking passwords
- Cracking paid software to remove license protection
- Creating keygens or serial key generators
- Distributing pirated software
- Breaking encryption algorithms
- Defacing websites
- Stealing confidential data
- Injecting malware into cracked software

Goal:

To **damage**, **steal**, **bypass**, or **illegally modify software or systems**.

Example : Crackers (Always Malicious)

✓ Software Crackers

Remove license keys from paid apps and turn them into “free” versions.

✓ Password Crackers

Use tools like John the Ripper, Hydra to break into accounts.

✓ Serial Key/Keygen Makers

Develop programs that generate illegal activation keys.

✓ **Website Crackers**

Deface websites, destroy data, inject backdoors.

Why Crackers Are Dangerous

- Spread pirated software with malware
- Cause massive financial losses to companies
- Damage websites and servers
- Steal credentials and money
- Break encryption and authentication
- Can create backdoors for future attacks
- Reduce trust in digital systems

Feature	Hackers	Crackers
Intent	Can be ethical or unethical	Always malicious
Focus	Security research, penetration testing	Breaking into systems/software
Legality	White hats work legally	Illegal activity
Motives	Security improvement, learning, curiosity	Theft, piracy, destruction
Skill Level	High technical expertise	Moderate to high
Activities	Finding vulnerabilities, fixing bugs	Password cracking, software piracy
Outcome	Improves cybersecurity	Causes damage or illegal distribution
Ethical Value	Can be positive	Always negative

Major Types of Hackers

Below are the **most widely recognized types**:

1️⃣ **White Hat Hackers (Ethical Hackers)**

Ethical, legal hackers who work to **secure systems**.

Features:

- Authorized to test systems
- Prevent cyberattacks
- Perform penetration testing and vulnerability assessments
- Improve defense mechanisms

Example:

- Bug bounty hunters
- CEH (Certified Ethical Hacker) professionals

2️⃣ Black Hat Hackers (Malicious Hackers)

Illegal hackers who access systems to **steal or damage data**.

Features:

- Exploit vulnerabilities for money, revenge, or power
- Create malware, ransomware
- Sell stolen data on the dark web

Example:

- Ransomware gangs
- Credit card thieves

3️⃣ Grey Hat Hackers

A mix between white hat & black hat.

Features:

- Access systems without permission
- No harmful intent, but unethical behavior
- Often reveal vulnerabilities publicly or to the organization

Example:

- Hackers who reveal website bugs without malicious activity

4️⃣ Script Kiddies

Amateurs who use ready-made tools or scripts created by others.

Features:

- Low skill level
- Use simple tools to perform DDoS or website defacement

- Often motivated by thrill or attention

Example:

- Young users running pre-built hacking tools

5️⃣ Hacktivists

Hackers motivated by political, social, or religious beliefs.

Features:

- Use hacking as a form of protest
- Deface websites, leak information
- Spread digital campaigns

Example:

- Anonymous hacking government websites

6️⃣ State-Sponsored / Nation-State Hackers

Highly skilled hackers working for a **government**.

Features:

- Conduct cyber espionage
- Attack enemy nations
- Target military, nuclear, and critical infrastructure

Examples:

- APT28 (Russia)
- Lazarus Group (North Korea)

7️⃣ Cyber Terrorist Hackers

Hackers who use digital attacks to create **fear**, **panic**, or **violence**.

Features:

- Target critical infrastructure
- Aim to disrupt society
- Spread radical propaganda

Examples:

- Terrorist organizations hacking power grids, government websites

8️⃣ Insider Hackers

Employees or contractors who misuse internal access.

Features:

- Know internal systems very well
- Steal or leak data intentionally or accidentally
- Very hard to detect

Examples:

- Disgruntled employee leaking sensitive company data

9️⃣ Elite Hackers

Highly skilled, top-level hackers.

Features:

- Discover zero-day vulnerabilities
- Create advanced exploits
- Break into highly secure systems

Examples:

- Hackers behind APTs and nation-level attacks

🔟 Whistleblower Hackers

Expose corruption or illegal activities within organizations.

Features:

- Leak sensitive documents
- Aim for transparency but still break security

Example:

- Edward Snowden (leaked NSA documents)

📘 CYBER CRIMES

A **Cyber Crime** is any unlawful activity performed using computers, networks, the internet, or digital devices with the purpose of harming individuals, organizations, or governments.

In simple words:
➡️ *Cyber Crime is a crime committed in digital space using technology.*

Cyber crimes may involve:

- Stealing data

- Hacking systems
- Online fraud
- Identity theft
- Financial scams
- Spreading malware
- Cyber bullying
- Online harassment

Characteristics of Cyber Crime

- Happens in virtual/digital world
- Can be anonymous and hard to trace
- Global in nature (cross-border)
- Highly scalable (target thousands easily)
- Uses advanced tools & techniques
- Causes financial, psychological, and national-level harm

Causes of Cyber Crimes

- Financial gain
- Revenge
- Curiosity
- Lack of strong laws/enforcement
- Weak security systems
- Human error (phishing clicks)
- Increasing digital dependence
- Anonymity on the internet
- Global cyber-criminal networks

Impact of Cyber Crimes

On Individuals

- Financial loss
- Mental stress
- Loss of privacy

- Reputation damage
- Identity theft

On Organizations

- Data breaches
- Business downtime
- Loss of customer trust
- Lawsuits & fines
- Damage to reputation

On Nation

- Threat to national security
- Critical infrastructure disruption
- Economic loss
- Spread of misinformation
- Loss of public trust

Prevention of Cyber Crimes

For Individuals

- Use strong passwords & 2FA
- Avoid clicking unknown links
- Use updated antivirus/firewalls
- Don’t share OTP or banking details
- Verify before online transactions
- Avoid pirated software
- Use secure Wi-Fi networks

For Organizations

- Regular security audits
- Employee training
- Firewalls, IDS, IPS
- Data encryption
- Backup & recovery plans

- Implement zero-trust architecture
- Access control policies

For Government

- Strong cyber laws
- National cybersecurity strategies
- CERT-In coordination
- Cyber police cells
- Awareness programs

Advantages of Studying Cyber Crime

- Helps understand cyber threats
- Improves online safety
- Helps in cybersecurity career
- Reduces financial and data loss
- Helps organizations strengthen security

📘 CYBER-ATTACKS

A **Cyber-Attack** is a deliberate attempt by hackers or malicious actors to damage, disrupt, steal, or gain unauthorized access to computer systems, networks, or data.

In simple words:
➡️ *A cyber-attack is an attempt to break into or harm a digital system.*

Cyber-attacks target:

- Individuals
- Businesses
- Banks
- Government
- Critical infrastructure

Objectives of Cyber Attacks

- Steal sensitive data
- Financial gain
- Destroy or damage systems

- Spy on organizations or nations
- Take control of systems
- Spread fear or misinformation
- Deface websites
- Disrupt services (downtime)

Categories of Cyber Attacks

Cyber-attacks can be broadly divided into:

- **Active Attacks**: Intentionally modifying, damaging, or attacking systems
- **Passive Attacks**: Monitoring or stealing information without altering data

Consequences of Cyber-Attacks

Impact on Individuals

- Identity theft
- Loss of money
- Privacy invasion
- Emotional stress

Impact on Organizations

- Data breach
- Financial loss
- Reputation damage
- Service downtime
- Legal penalties

Impact on Nation

- Threat to national security
- Disruption of power grids, transportation, military systems
- Economic slowdown
- Public fear

Prevention of Cyber-Attacks

- Strong passwords & multi-factor authentication
- Firewalls, antivirus, and IDS/IPS

- Regular patching & system updates
- Employee awareness (avoid phishing)
- Data encryption
- Network monitoring
- Zero-trust architecture
- Backup & disaster recovery plans
- Secure development practices (SDLC)

📘 VULNERABILITIES

A **Vulnerability** is a weakness, flaw, or gap in a system, software, network, or device that can be exploited by attackers to gain unauthorized access, steal data, or disrupt operations.

In simple words:
➡️ *A vulnerability is a security weakness that allows hackers to attack a system.*

Key Characteristics of Vulnerabilities

- Exists due to errors or weaknesses
- Can be exploited by cyber-attacks
- May appear in hardware, software, humans, or processes
- Creates risk of data theft or system damage
- Requires patching or mitigation
- Does not cause harm on its own, but allows attacks

Causes of Vulnerabilities

1️⃣ **Poor software coding / design flaws**

Unsecured code, no input validation.

2️⃣ **Misconfigurations**

Weak passwords, open ports, exposed databases.

3️⃣ **Unpatched Software**

Outdated apps or operating systems.

4️⃣ **Human Errors**

Phishing clicks, weak passwords, accidental data exposure.

5️⃣ **Lack of Encryption**

Unprotected data in storage or transmission.

6 Using insecure third-party components

Compromised plugins or libraries.

7 Network Weaknesses

Unsecured Wi-Fi, no firewall.

8 Insider neglect

Employees failing to follow security policies.

Types of Vulnerabilities (Detailed)

1 Software Vulnerabilities

Flaws in software code or logic.

Examples:

- Buffer Overflow
- SQL Injection vulnerability
- Cross-Site Scripting (XSS)
- Hardcoded passwords

2 Network Vulnerabilities

Weaknesses in network configuration or design.

Examples:

- Open ports
- Weak or no firewall
- Unsecured Wi-Fi
- ARP spoofing vulnerabilities

3 Hardware Vulnerabilities

Weaknesses inside hardware components.

Examples:

- Meltdown and Spectre CPU vulnerabilities
- Unprotected USB ports
- Faulty firmware

4 Operating System Vulnerabilities

Flaws in Windows, Linux, macOS OS environments.

Examples:

- Privilege escalation bugs
- Zero-day OS vulnerabilities
- SMB protocol vulnerabilities

5️⃣ Human Vulnerabilities

People are often the weakest link.

Examples:

- Falling for phishing attacks
- Weak passwords like “123456”
- Sharing credentials
- Lack of training

6️⃣ Physical Vulnerabilities

Weaknesses in physical access security.

Examples:

- Unlocked server rooms
- Lost devices (laptops, USB drives)
- No CCTV monitoring

7️⃣ Configuration Vulnerabilities

Systems not configured correctly.

Examples:

- Default passwords
- Misconfigured cloud buckets (AWS S3 exposed)
- Excessive user permissions

8️⃣ Zero-Day Vulnerabilities

Unknown to the vendor; no patch available.

Examples:

- Stuxnet exploited zero-day Windows vulnerabilities.

Risks Associated with Vulnerabilities

- Data leakage
- Unauthorized access
- Loss of privacy
- Ransomware infections
- Service outages
- Financial loss
- Reputational damage
- National security threats

How to Prevent & Manage Vulnerabilities

1️⃣ Patch Management

Regular updates for OS, software, firmware.

2️⃣ Vulnerability Assessment & Penetration Testing (VAPT)

Scanning and testing systems for weaknesses.

3️⃣ Strong Configuration Management

Disable unused ports, change default passwords.

4️⃣ Use Firewalls & IDS/IPS

Block malicious traffic and detect intrusions.

5️⃣ Encryption

Protect data in transit and storage.

6️⃣ Employee Training

Avoid phishing, use strong passwords.

7️⃣ Zero-Trust Architecture

Never trust — always verify.

8️⃣ Regular Backup

Even if attacked, data can be restored.

📘 MALWARE THREATS

Malware (short for *malicious software*) is any software or code intentionally designed to damage, disrupt, steal, or gain unauthorized access to computer systems, networks, or devices.

In simple words:

➡️ *Malware is harmful software that attacks your computer or data.*

Malware is one of the **most common cyber threats** faced worldwide.

Characteristics of Malware

- Designed to harm systems or steal information
- Works secretly without user permission
- Can replicate and spread automatically
- Delivered through email attachments, downloads, USBs
- Can disable security tools
- Varies in complexity and damage level

Types of Malware (Detailed)

1️⃣ Virus

A program that attaches itself to legitimate files and spreads when the files are executed.

Features:

- Needs a host file
- Can delete or corrupt data

Example:

ILOVEYOU virus (2000)

2️⃣ Worm

Malware that self-replicates and spreads through networks without a host file.

Features:

- Spreads quickly
- Causes network congestion

Example:

SQL Slammer worm

3️⃣ Trojan Horse

Appears as a legitimate software but performs malicious actions in the background.

Features:

- Doesn’t replicate

- Creates backdoors

Example:

Emotet Trojan

4️⃣ Ransomware

Encrypts user data and demands ransom to restore access.

Features:

- Extremely damaging
- Often delivered via phishing

Examples:

WannaCry, Petya, Locky

5️⃣ Spyware

Secretly monitors user activities and collects personal data.

Features:

- Tracks keystrokes (keyloggers)
- Sends data to hackers

Example:

DarkComet keylogger

6️⃣ Adware

Displays unwanted ads, redirects browsers, or collects data.

Example:

Ad-injecting browser extensions

7️⃣ Rootkits

Malware designed to hide itself and other malicious processes.

Features:

- Hard to detect
- Gains high-level system access

Example:

Sony BMG rootkit (2005)

8️⃣ Botnets

A network of infected computers controlled remotely by attackers.

Features:

- Used for DDoS attacks
- Controlled by C2 (command & control) server

Examples:

Mirai botnet, Zeus botnet

9️⃣ Fileless Malware

Operates in RAM without saving on disk.

Features:

- Extremely hard to detect
- Leaves no traces

Example:

PowerShell-based attacks

🔟 Logic Bomb

Code that triggers when a specific condition is met.

Example:

Employee sets malware to activate after being fired.

How Malware Spreads

- Phishing emails
- Malicious downloads or attachments
- Pirated software
- USB/External drives
- Compromised websites
- Weak passwords
- Public Wi-Fi attacks
- Social engineering

Effects of Malware Attacks

On Individuals

- Loss of personal data

- Identity theft
- Financial fraud
- System slowdown
- Data encryption (ransom)

On Organizations

- Data breaches
- Operational downtime
- Financial loss
- Reputation damage
- Legal penalties
- Theft of intellectual property

On Nation / Government

- Attacks on critical infrastructure
- Data theft by foreign actors
- National security threats

How to Prevent Malware Threats

Technical Measures

- Install antivirus/anti-malware
- Update operating system & apps
- Use firewalls
- Enable ransomware protection
- Backup data regularly
- Disable macros in documents
- Use email filters
- Network segmentation

User-Level Measures

- Don’t click unknown links
- Avoid pirated software
- Don’t open suspicious attachments

- Use strong passwords
- Avoid using unknown USB drives
- Use secure Wi-Fi only

Organizational Measures

- Employee awareness training
- Incident response plans
- Endpoint security solutions
- Zero-trust architecture
- Regular vulnerability scans & patches

📘 SNIFFING

Sniffing is a cyber technique in which an attacker intercepts, captures, and analyzes data packets traveling through a network.

In simple words:
➡️ *Sniffing is "listening" to network traffic to steal information.*

It can be used for **legitimate** purposes (network monitoring) or **malicious** purposes (stealing passwords, credit card data).

What Attackers Can Capture Through Sniffing

- Usernames & passwords
- Credit card details
- Cookies & session tokens
- Email content
- Chat messages
- Files being transferred
- Network configuration details
- Website browsing data

Types of Sniffing

1️⃣ Passive Sniffing

- Listens quietly to network traffic.
- Works on **hub-based** or broadcast networks.
- Hard to detect.

Example:

Capturing unencrypted traffic on a Wi-Fi network.

2️⃣ Active Sniffing

- Attacker injects packets or manipulates traffic.
- Used in **switched networks**, where traffic is not broadcast.

Examples:

- ARP Poisoning
- MAC Flooding
- DHCP Spoofing
- DNS Poisoning

Techniques Used in Sniffing Attacks

1️⃣ ARP Poisoning

Manipulating ARP tables to redirect traffic through the attacker.

2️⃣ MAC Flooding

Flooding a switch with fake MAC addresses to force it into hub mode.

3️⃣ DHCP Spoofing

Sending fake DHCP responses to control network settings.

4️⃣ DNS Spoofing

Redirecting users to fake websites by altering DNS responses.

5️⃣ Man-in-the-Middle (MitM)

Intercepting communication between two systems.

6️⃣ Packet Injection

Adding malicious packets into legitimate communication.

Popular Sniffing Tools

✓ Wireshark — most widely used packet analyzer

✓ Tcpdump — command-line traffic capture

✓ Cain & Abel — password sniffing

✓ Ettercap — ARP poisoning and sniffing

✓ Kismet — Wi-Fi sniffing

✓ Nmap — network exploration + packet capture

✓ Aircrack-ng — wireless sniffing and cracking

Indicators of Sniffing on a Network

- Unusual network slowdowns
- Duplicate ARP responses
- Unknown MAC addresses on the network
- Strange TLS/SSL certificate warnings
- Unauthorized device detected
- DNS response anomalies

Impact of Sniffing Attacks

- Loss of sensitive data
- Identity theft
- Financial fraud
- Unauthorized account access
- Compromise of login credentials
- Corporate data breach
- Man-in-the-middle attacks

Prevention & Countermeasures

🔒 **1. Use Encryption**

- HTTPS
- SSL/TLS
- VPN
- WPA3/WPA2 for Wi-Fi

🔒 **2. Static ARP Entries**

Prevents ARP spoofing.

🔒 **3. Switch Port Security**

Restricts number of MAC addresses on a port.

🔒 **4. Strong Wi-Fi Security**

Use WPA3, disable open networks.

🔒 5. Use Intrusion Detection Systems (IDS)

Detect ARP poisoning and rogue devices.

🔒 6. Network Segmentation

Limits access to sensitive areas.

🔒 7. Disable Unused Ports

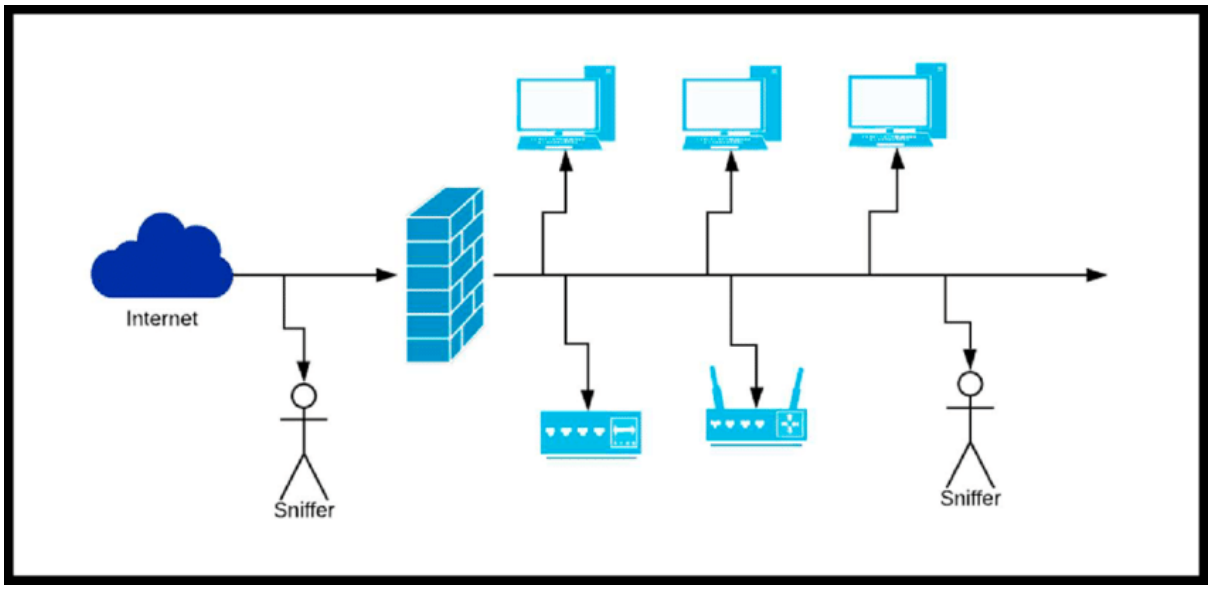
Reduces attack surface.

🔒 8. Employee Awareness

Avoid connecting to open Wi-Fi; beware of fake hotspots.

Sniffing vs Spoofing

Sniffing	Spoofing
Captures traffic	Fakes identity
Passive/active	Always active
Steals data	Misleads systems
Example: packet sniffing	Example: DNS spoofing



📘 GAINING ACCESS

Gaining Access is a phase of a cyberattack or ethical hacking process where the attacker exploits vulnerabilities to enter a system, network, application, or account.

In simple words:

➡️ *Gaining Access means breaking into a system by exploiting its weaknesses.*

This stage comes **after reconnaissance & scanning**, and before **maintaining access** and **covering tracks**.

Purpose of Gaining Access

- Steal sensitive data
- Modify or delete information
- Install malware/backdoors
- Take control of user accounts
- Perform privilege escalation
- Hijack entire systems
- Interrupt services

Attackers use technical and social techniques to gain unauthorized access.

Steps Involved in Gaining Access

1️⃣ Identify vulnerabilities
2️⃣ Choose exploitation method
3️⃣ Execute the exploit
4️⃣ Gain entry to the system
5️⃣ Escalate privileges (root/admin)
6️⃣ Establish backdoor access

Techniques Used for Gaining Access

1️⃣ Password Attacks

Attackers try to break passwords to access user accounts.

Methods:

- Brute-force attack
- Dictionary attack
- Credential stuffing
- Password cracking tools (John the Ripper, Hydra)
- Guessing weak passwords

Example:

Trying “admin/admin” on routers.

2️⃣ Exploiting Vulnerabilities

Taking advantage of weak code or outdated software.

Examples:

- SQL Injection
- Buffer Overflow
- Cross-Site Scripting (XSS)
- Missing patches
- Zero-day vulnerabilities

Tools:

Metasploit, SQLmap, Burp Suite.

3️⃣ Social Engineering

Tricking people to reveal information or perform actions.

Methods:

- Phishing emails
- Fake tech support calls
- Impersonation
- Baiting (USB drops)

Example:

Sending an email asking for login details.

4️⃣ Malware Attacks

Using malicious software to break into systems.

Types:

- Trojans
- Keyloggers
- Remote Access Trojans (RATs)
- Spyware
- Ransomware

Example:

Malicious email attachment that installs a Trojan.

5️⃣ Network-Based Attacks

Targeting network protocols or devices.

Methods:

- ARP poisoning
- DNS spoofing
- Man-in-the-Middle (MitM)
- Sniffing passwords
- Exploiting open ports

6️⃣ Session Hijacking

Stealing session tokens to access user accounts without passwords.

Examples:

- Stealing cookies
- Using unsecured Wi-Fi

7️⃣ Exploiting Misconfigurations

Gaining access due to poor security configurations.

Examples:

- Default credentials
- Open S3 buckets
- Improper permissions
- Unrestricted ports

8️⃣ Physical Access Attacks

Entering secure premises or accessing devices physically.

Examples:

- Plugging USB malware
- Stealing laptops
- Shoulder surfing

📘 ESCALATING PRIVILEGES

Privilege Escalation is a cyberattack technique where an attacker, after gaining initial access to a system, attempts to increase their access rights from a **low-level user** to **administrator/root-level privileges**.

In simple words:
➡️ *Privilege escalation means turning small access into full control.*

Privilege escalation helps attackers:

- Install malware
- Create backdoors
- Modify/delete data
- Access sensitive files
- Take over entire systems

It is one of the **most important phases** of hacking.

Why Attackers Perform Privilege Escalation

- Initial access usually has limited permissions
- Admin privileges allow full system control
- Helps in maintaining persistent access
- Enables lateral movement across networks
- Allows disabling security tools
- Helps execute high-level commands

Types of Privilege Escalation

1️⃣ Vertical Privilege Escalation (Privilege Elevation)

Attacker increases privilege level.
Example:
Normal user → Administrator

Example Scenario:

A student account on a system becomes an admin by exploiting a flaw.

2️⃣ Horizontal Privilege Escalation

Attacker stays at the same level but gains access to another user's data or functions.

Example Scenario:

A normal user accessing another user's email or bank account.

Techniques Used for Privilege Escalation

1️⃣ Exploiting Software Vulnerabilities

- Vulnerable kernel drivers
- Buffer overflow
- Unpatched OS
- Local privilege escalation bugs
- DLL hijacking

Tools:

Metasploit, PowerUp, LinPEAS, WinPEAS.

2️⃣ Password Cracking

- Weak or reused passwords
- Dumping password hashes
- Cracking using John the Ripper or Hashcat
- Credential stuffing

3️⃣ Misconfigurations

- Incorrect file/system permissions
- Sudo misconfigurations
- World-writable files
- Weak ACL (Access Control List)

4️⃣ Token/Session Hijacking

- Stealing administrator session tokens
- Using cookies to access admin accounts

5️⃣ Security Bypass Techniques

- UAC bypass (Windows)
- Disabling antivirus or firewall
- Jailbreak/rooting exploits

6️⃣ Using Malware

- Trojans that provide backdoor access

- Rootkits designed for privilege escalation

7️⃣ Social Engineering

- Tricking admin into revealing credentials
- Fake “IT Support” calls
- Phishing admin accounts

8️⃣ Kernel Exploits

- Vulnerabilities at OS kernel level
- Gives complete system control
- Used in advanced cyberattacks

Practical Examples of Privilege Escalation

✓ Sudo Misconfiguration (Linux)

User allowed to run commands like sudo vi, enabling root shell.

✓ Sticky Keys Exploit (Windows)

Replacing sethc.exe with cmd.exe allows admin-level shell at login screen.

✓ Pass-the-Hash Attack

Attacker uses hashed credentials to authenticate without knowing the password.

✓ Dirty COW Vulnerability (Linux)

A famous privilege escalation bug allowing root access.

✓ Unpatched Windows Vulnerability (CVE-2016-5195)

Exploited to gain system-level privileges.

Consequences of Successful Privilege Escalation

- Complete system compromise
- Loss of sensitive data
- Ransomware installation
- Full administrative control
- Lateral movement to other systems
- Permanent backdoors created
- Security tools disabled

- Total organizational collapse (in large networks)

Defending Against Privilege Escalation

1️⃣ Patch & Update Systems

Fix OS/software vulnerabilities regularly.

2️⃣ Least Privilege Principle (PoLP)

Users should only have necessary permissions.

3️⃣ Strong Password Policies

- Enforce complex passwords
- Multi-factor authentication (MFA)

4️⃣ Secure Configurations

- Fix Sudo & ACL permissions
- Restrict file access
- Disable unnecessary services

5️⃣ Monitor Logs & Alerts

Detect unauthorized privilege changes.

6️⃣ Use Endpoint Security

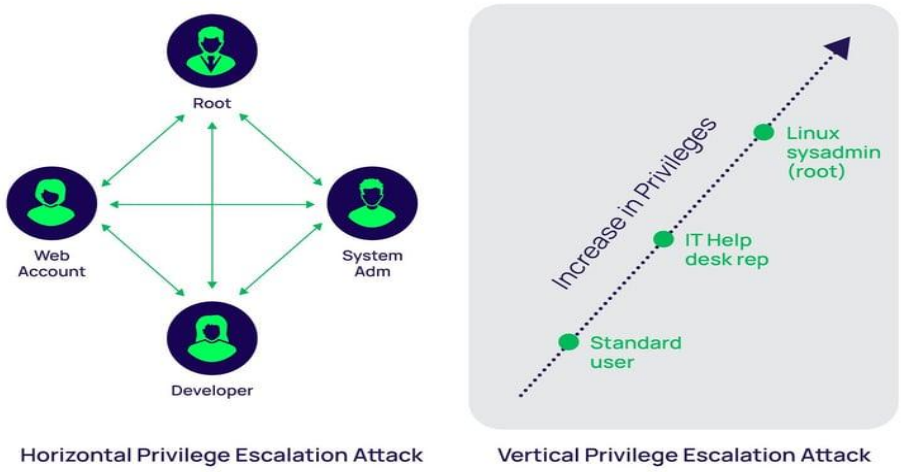
EDR, antivirus, anti-malware solutions.

7️⃣ Segmentation & RBAC

- Role-Based Access Control
- Separate admin accounts from regular accounts

8️⃣ Disable Default Accounts

Prevent attackers from using built-in admin accounts.



Horizontal Privilege Escalation Attack Vertical Privilege Escalation Attack

📘 EXECUTING APPLICATIONS

Executing Applications is a phase in cyberattacks (and ethical hacking) where attackers run malicious programs, scripts, payloads, or commands on a compromised system after gaining access.

In simple words:
➡️ *Executing applications means running malicious software or code on a target system to control or damage it.*

This is usually done **after the attacker gains entry** and **before maintaining access** or **covering tracks**.

Why Attackers Execute Applications?

Attackers execute applications to:

- Take full control of the target system
- Install backdoors or malware
- Steal sensitive data
- Capture keystrokes (keyloggers)
- Download additional payloads
- Spread inside the network
- Disable security systems
- Launch further attacks (pivoting / lateral movement)

How Attackers Execute Malicious Applications

Below are the common methods:

1️⃣ Remote Code Execution (RCE)

Exploiting vulnerabilities to run code remotely on a system.

Example:

Log4j (Log4Shell) allowed attackers to execute commands on servers.

2️⃣ Command Injection

Injecting OS commands through vulnerable input fields.

3️⃣ Exploiting Software Vulnerabilities

- Buffer overflow
- Unpatched applications
- Misconfigured services

Example:

Using Metasploit to run payloads on vulnerable apps.

4️⃣ Running Malicious Script Files

Attackers upload & run:

- .exe files
- .bat (batch) scripts
- .sh (shell scripts)
- Python scripts
- PowerShell commands

5️⃣ Trojan Execution

Attacker convinces user to install a fake application that opens a backdoor.

Example:

Fake “game installer” that installs a RAT.

6️⃣ Droppers and Downloaders

Small malware that downloads bigger malware once executed.

7️⃣ Keyloggers / Spyware

Executed to record keystrokes and user activity.

8️⃣ Remote Access Trojans (RATs)

Allow remote control of the infected system.

Examples:

- DarkComet
- NjRAT

9️⃣ Macros in Documents

Malicious Microsoft Word/Excel macros execute malware when opened.

🔟 Execution via Social Engineering

User is tricked into double-clicking an infected file.

📘 HIDING FILES

Hiding Files is a post-exploitation technique used by attackers to conceal malicious files, scripts, logs, or tools on a compromised system so the user or security software cannot detect them.

In simple words:
➡️ *Hiding files means storing malicious or unauthorized files in a way that they remain invisible or undetectable.*

This helps attackers:

- Stay inside the system longer
- Avoid detection
- Maintain persistence
- Hide malware or backdoors
- Hide evidence of intrusion

Why Attackers Hide Files

- To avoid antivirus/monitoring tools
- To maintain long-term access
- To hide stolen data
- To hide malware payloads
- To cover tracks in an investigation
- To bypass digital forensics

Techniques Used for Hiding Files

1️⃣ File Attribute Manipulation

Changing file attributes to hide them from standard view.

2️⃣ Storing Files in Unusual Locations

Attackers hide files in rarely checked directories:

- System folders (System32)
- Temp folders
- Recycle Bin
- Hidden partitions
- Device drivers directory
- Inside registry keys (Windows)

3️⃣ File Name Tricks

Using misleading or confusing names like:

- svch0st.exe (looks like svchost.exe)
- .jpg.exe
- Unicode characters to hide extensions

4️⃣ NTFS Alternate Data Streams (ADS) (Windows Only)

Hide data *inside* another file without changing its size.

5️⃣ Steganography

Hiding data inside images, audio, or video files.

Example:

Hiding malware inside a .jpg image file.

Tools:

- Steghide
- OpenStego

6️⃣ Encryption and Obfuscation

Attackers encrypt malicious files so antivirus cannot read them.

Techniques:

- Base64 encoding
- Code obfuscation
- File packing (UPX)

7️⃣ Using Rootkits (Most Dangerous)

Rootkits hide files, processes, and registry entries from the OS.

Example:

Kernel-level rootkit hiding malware in Linux or Windows.

8️⃣ Hiding in Memory (Fileless Malware)

No file is stored on disk; malware runs only in RAM → extremely hard to detect.

9️⃣ Browser Storage & Cookies

Attackers hide malicious scripts or tokens inside:

- HTML5 local storage
- Browser cache
- Cookies

Prevention of File Hiding Attacks

- Use updated antivirus/EDR
- Disable autorun for USB
- Block execution in Temp folders
- Restrict admin/root privileges
- Use application whitelisting
- Enforce strict access permissions
- Patch OS & software
- Regular system audits

📘 COVERING TRACKS

Covering Tracks is the final stage of a cyberattack (or ethical hacking process) where attackers hide their activities, eliminate evidence, and remove traces of intrusion to avoid detection.

In simple words:
➡️ *Covering tracks means hiding all evidence so nobody knows a system was hacked.*

This ensures the attacker:

- Stays undetected
- Avoids forensic investigation
- Maintains long-term access
- Escapes legal or administrative consequences

Why Attackers Cover Tracks

- To avoid being identified
- To prevent triggering alarms
- To safely ex-filtrate (steal) data
- To remain in the system longer (persistence)
- To hide backdoors and malware
- To continue future attacks silently

When Does Covering Tracks Occur?

It is the **last phase** in the Ethical Hacking cycle:

1. Reconnaissance
2. Scanning
3. Gaining Access
4. Escalating Privileges
5. Maintaining Access
6. **Covering Tracks** ← (Final Phase)

Techniques Used to Cover Tracks

1️⃣ Clearing System Logs

Attackers delete or modify:

- Security logs
- Application logs
- Event viewer logs (Windows)
- Syslog files (Linux)
- Web server logs (Apache, Nginx)

2️⃣ Clearing Browser & History Logs

Attackers delete:

- Cookies
- Cache
- Browsing history
- Download history

3️⃣ File Time-Stamp Modification ("Timestomping")

Changing creation, modification, or access times of malicious files.

Example tools:

- Metasploit Timestomp
- Touch command in Linux (touch -t)

4️⃣ Rootkits (Most Dangerous)

Rootkits hide:

- Malicious files
- Processes
- Network connections
- Registry entries

This makes detection extremely difficult.

5️⃣ Deleting Malware or Exploit Tools

Attackers remove their tools:

- Scripts
- RATs
- Scanners
- Payload files

6️⃣ Covering Tracks in Network

- Clearing ARP tables
- Removing SSH histories
- Clearing PowerShell logs
- Replacing log entries with fake ones

7️⃣ Using Encrypted or Obfuscated Communication

Hides command-and-control (C2) activity.

8️⃣ Creating Fake Accounts or Deleting Created Accounts

Used to confuse investigators.

9️⃣ Using Steganography

Hide stolen data inside media files so evidence is concealed.

1 0 Disabling or Manipulating Security Systems

- Turning off antivirus logs
- Disabling SIEM alerts
- Killing monitoring processes

Impact of Covering Tracks

- Delays investigation
- Forensic teams fail to identify attack path
- Attackers stay inside system longer
- More systems get infected
- Increased financial & data loss
- No clear evidence for legal action

How to Detect and Prevent Track Covering

1 Use Centralized Logging (SIEM)

Even if logs are deleted locally, copies exist in central servers.

Tools:

- Splunk
- ELK Stack
- QRadar
- Azure Sentinel

2 Enable Log Integrity and Tamper Protection

- Digitally signed logs
- Immutable storage
- WORM (Write Once Read Many) logs

3 File Integrity Monitoring (FIM)

Detects unauthorized changes.

Tools:

- Tripwire
- OSSEC

4 Use Anti-Rootkit Tools

- GMER
- Malwarebytes Anti-Rootkit
- Kaspersky TDSSKiller

5️⃣ **Enable Security Alerts**

Monitor:

- Log deletions
- Timestomping
- Suspicious PowerShell commands

6️⃣ **Restrict Admin Privileges**

Least Privilege Principle → reduces attacker's ability to wipe traces.

7️⃣ **Network Monitoring**

Detect hidden traffic, encrypted C2 channels.

8️⃣ **Regular Auditing & Backups**

Investigators can cross-verify logs.

📘 WORMS

A **Computer Worm** is a type of **self-replicating malware** that spreads automatically across computers and networks **without needing a host file or user action**.

In simple words:
➡️ *A worm is malware that spreads on its own through networks, infecting as many devices as possible.*

Key Characteristics of Worms

- Self-replicating (copies itself)
- Does **not** require a host file (unlike viruses)
- Spreads through networks automatically
- Can overload networks due to massive replication
- Often used to deliver payloads (ransomware, Trojans)
- Works silently in the background
- Hard to detect in early stages

How Worms Spread

Worms usually take advantage of:

✓ **Network vulnerabilities**

Unpatched systems, weak firewall rules.

✓ **Email attachments**

Sending copies of themselves to all contacts.

✓ **USB drives**

Auto-run features in removable media.

✓ **File sharing services**

P2P networks, shared drives.

✓ **Internet scanning**

Scanning for devices with weak security.

Types of Worms

1️⃣ **Email Worms**

Spread by sending infected emails and attachments.

Example:
ILOVEYOU worm.

2️⃣ **Internet Worms**

Scan the internet for vulnerable devices.

Example:
Code Red worm.

3️⃣ **File-Sharing Worms**

Spread through P2P file sharing networks.

4️⃣ **Instant Messaging Worms**

Spread through apps like Skype, WhatsApp, Slack.

5️⃣ **Network Worms**

Target corporate or local networks.

Example:
SQL Slammer.

6️⃣ **Mobile Worms**

Infect smartphones and spread via Bluetooth/SMS.

7️⃣ Payload-Carrying Worms

Worms that carry extra malware like ransomware, Trojans, or backdoors.

Symptoms of a Worm Infection

- Sudden system slowdown
- High CPU/Network usage
- Unexpected files/folders
- Emails sent without user's knowledge
- Network congestion
- Pop-ups or unusual behavior

How to Prevent Worm Attacks

1️⃣ Regular Updates & Patch Management

Fix vulnerabilities before worms exploit them.

2️⃣ Strong Firewall Configuration

Block unnecessary ports (like SMB ports for WannaCry).

3️⃣ Antivirus & Anti-Malware Software

Detects known worms.

4️⃣ Disable Auto-Run Features

Prevents USB-based worm spread.

5️⃣ Network Segmentation

Limits worm movement across internal networks.

6️⃣ User Awareness

Don't open suspicious attachments or downloads.

7️⃣ Email Filtering

Blocks worm-loaded attachments.

8️⃣ IDS/IPS Deployment

Detects abnormal network scanning behavior.

Difference: Worms vs Viruses

Feature	Worm	Virus
Needs host file	❌ No	✓ Yes
Spreads automatically	✓ Yes	❌ No
Network spread	✓ Fast	❌ Slow
Self-replicating	✓ Yes	✓ Yes
User action needed	❌ No	✓ Usually yes

📘 TROJANS

A **Trojan** or **Trojan Horse** is a type of malware that disguises itself as a legitimate file or program but performs malicious actions once executed.

In simple words:
➡️ *A Trojan looks harmless, but once you install it, it gives attackers access to your system.*

Unlike viruses and worms:

- A Trojan **does NOT replicate** itself.
- It requires the user to **download or run** it.

Characteristics of Trojans

- Pretends to be useful or harmless
- Needs user interaction to run
- Does not self-replicate
- Difficult to detect due to disguise
- Often used for remote control
- Can steal sensitive data
- Can disable security tools

How Trojans Spread

- Fake software downloads
- Email attachments (invoice.pdf.exe)
- Pirated software and cracked games

- Malicious ads and pop-ups
- Social engineering (“Urgent Update Found”)
- Fake mobile apps (Android Trojans)
- USB drives containing Trojan installers

Types of Trojans (Detailed)

1️⃣ Remote Access Trojan (RAT)

Allows attackers full remote control of the victim’s system.

Capabilities:

- Access files
- Turn on camera/microphone
- Keylogging
- Screen recording

Examples:
DarkComet, NjRAT, BlackShades

2️⃣ Trojan Downloader

Downloads additional malware from the internet.

Example:
Downloader.Agent

3️⃣ Trojan Dropper

Installs or drops multiple other malicious files onto the system.

4️⃣ Banking Trojan

Steals banking details, passwords, OTPs, PINs.

Examples:
Zeus, Dridex, SpyEye

5️⃣ Backdoor Trojan

Creates hidden access points for attackers.

Used for:

- Future entry
- Bypassing authentication
- Persistent control

6️⃣ Trojan Spy / Information Stealer

Steals data:

- Keystrokes
- Screenshots
- Clipboard data
- Browser passwords

Examples:
Keylogger.Trojan, FormGrabber

7️⃣ Ransomware Trojan

Disguised as a useful file → encrypts data after execution.

Example:
WannaCry (spread using trojan-like behavior)

8️⃣ Rootkit Trojan

Hides other malware and prevents detection.

9️⃣ DDoS Trojan

Infects machines to turn them into botnets for attacks.

🔟 Mobile Trojans

Attack smartphones; steal SMS, contacts, banking apps.

📘 VIRUSES

A **Computer Virus** is a type of malicious software that attaches itself to a legitimate file, program, or system and replicates when the infected file or program is executed.

In simple words:
➡️ *A virus is malware that infects files and spreads when those files are run.*

Viruses cannot spread on their own — they need:
✓ A **host file**
✓ **User action** (like opening a file)

Key Characteristics of Viruses

- Requires a host file to attach to
- Needs user action to execute and spread
- Self-replicating once activated

- Can corrupt or delete data
- Can display unwanted messages or slow the system
- Can spread via USB, email attachments, downloads
- Often used for destruction rather than stealth

How Viruses Spread

- Infected email attachments
- Downloading pirated/cracked software
- USB drives or removable media
- Infected documents (Word/Excel)
- Malicious websites
- File sharing networks
- Shared folders in LAN networks

How a Virus Works (Lifecycle)

1. Virus attaches to a legitimate file/program
2. User executes the infected file
3. Virus activates and replicates
4. Virus infects other files or memory areas
5. Virus delivers payload (damage, corruption, theft)
6. Virus continues spreading to other systems

Types of Computer Viruses (Detailed)

1️⃣ File Infector Virus

Infects executable files (.exe, .com).

Example:

Cascade virus.

2️⃣ Boot Sector Virus

Infects the boot sector or Master Boot Record (MBR).

Example:

Michelangelo virus.

3️⃣ Macro Virus

Targets applications like MS Word, Excel using VBA macros.

Example:

Melissa virus.

4️⃣ Polymorphic Virus

Changes its code on every infection to evade antivirus detection.

Example:

Storm Worm.

5️⃣ Metamorphic Virus

Rewrites its own code from scratch each time it infects.

Example:

Zmist virus.

6️⃣ Resident Virus

Resides in memory and infects files as soon as they are opened.

Example:

Randex, CMJ.

7️⃣ Non-Resident Virus

Does not stay in memory; spreads when the infected program is run.

8️⃣ Multipartite Virus

Infects both boot sector and executable files.

9️⃣ Overwriting Virus

Overwrites data in files, making them unusable.

Example:

Trivial.88.

🔟 Browser Hijacker Virus

Modifies browser settings, redirects to malicious sites.

Symptoms of Virus Infection

- Slow system performance
- Frequent crashing or freezing
- Unexpected pop-ups

- Files disappearing or corrupted
- Programs opening/closing automatically
- High CPU usage
- Browser redirection
- Unknown processes running

Prevention of Virus Attacks

✓ 1. Install Antivirus/Anti-malware

Detects and removes viruses.

✓ 2. Update OS & Applications

Unpatched software is vulnerable.

✓ 3. Avoid Pirated Software

Cracked software is a major virus source.

✓ 4. Scan USBs Before Use

Disable autorun features.

✓ 5. Avoid Suspicious Attachments

Don't open unknown email files.

✓ 6. Use Firewalls

Blocks suspicious connections.

✓ 7. Enable Macro Security

Prevents macro viruses (Word/Excel).

✓ 8. Regular Backups

Recover data if infected.

📘 BACKDOORS

A **Backdoor** is a hidden method of bypassing normal authentication or security controls to gain unauthorized access to a system, network, or application.

In simple words:
➡️ *A backdoor is a secret entry point that allows attackers to enter a system whenever they want.*

Backdoors may be created by:

- **Hackers** → for long-term hidden access
- **Malware** (Trojan/RAT) → to control infected devices
- **Developers** → intentionally for debugging (dangerous if exposed)

Key Characteristics of Backdoors

- Hidden from users and security tools
- Allows remote, unauthorized access
- Operates silently in the background
- Often installed after gaining initial access
- Can survive reboots using persistence methods
- Used to deliver further attacks
- Difficult to detect and remove

How Backdoors Are Installed

Attackers may use:

✓ Malware (Trojan, RAT)

Once executed, it opens a secret communication channel.

✓ Exploiting vulnerabilities

Using RCE (Remote Code Execution) to drop backdoor files.

✓ Credential theft

Using stolen passwords to create hidden admin users.

✓ Web server exploit

Upload backdoor scripts (e.g., PHP shells).

✓ Insider threats

Employees may install unauthorized remote-access tools.

✓ Supply chain attack

Malicious code inserted into software updates.

Types of Backdoors

1️⃣ Application Backdoor

Placed inside legitimate software to bypass authentication.

Example: Hidden admin login in a web application.

2️⃣ System Backdoor

Modifies OS files, services, registry to provide stealth access.

Example: Hidden Windows user accounts.

3️⃣ Hardware/Firmware Backdoor

Built into hardware components or firmware.

Example: Compromised BIOS or IoT device firmware.

4️⃣ Remote Access Trojan (RAT)

Most common type of backdoor used by attackers.

Capabilities:

- File access
- Keylogging
- Camera access
- Remote shell

Examples:
NjRAT, DarkComet, Gh0st RAT

5️⃣ Web Shell Backdoor

Script uploaded to a server providing remote command execution.

Examples:

- c99.php
- r57.php
- China Chopper web shell

6️⃣ Hardcoded Backdoor

Developer intentionally inserts a backdoor for testing, but forgets to remove it.

Example:
Hardcoded “admin/admin” credentials in IoT devices.

7️⃣ Encrypted/Stealth Backdoor

Uses encryption to hide communication with attacker’s server.

How Backdoors Work (Step-by-Step)

1. Attacker gains initial access (phishing, exploit, malware)
2. Installs a backdoor hidden in system files or registry
3. Backdoor communicates with command & control (C2) server
4. Attacker maintains persistent, secret access
5. Executes commands, steals data, installs more malware
6. Covers tracks so the user doesn’t know

📘 ETHICAL HACKING

Ethical Hacking is the authorized process of identifying security weaknesses in computer systems, networks, or applications by using the same tools and techniques as malicious hackers — but with legal permission and for defensive purposes.

In simple words:
➡️ *Ethical hacking means hacking legally to improve security.*

Ethical hackers are also called:
✓ White Hat Hackers
✓ Security Analysts
✓ Penetration Testers

Goals of Ethical Hacking

- Identify vulnerabilities before attackers exploit them
- Strengthen system and network security
- Prevent data breaches
- Improve incident response
- Test real-world attack scenarios
- Ensure compliance (ISO, PCI-DSS, GDPR)

Who Is an Ethical Hacker?

An **Ethical Hacker** is a cybersecurity professional trained to test and secure systems legally.

Ethical hackers must:

- Follow rules of engagement
- Protect organizational data
- Report findings responsibly
- Never misuse their skills
- Obtain written permission

Five Phases of Ethical Hacking (CEH Model)

1️⃣ Reconnaissance (Footprinting)

Gathering information about the target before the attack.

Examples:

- Whois lookup
- Google hacking
- DNS enumeration

2️⃣ Scanning

Identifying open ports, services, and vulnerabilities.

Tools:

- Nmap
- Nessus
- OpenVAS

3️⃣ Gaining Access

Exploiting vulnerabilities to enter the system.

Methods:

- SQL Injection
- Password attacks
- Buffer overflow

4️⃣ Maintaining Access

Installing backdoors or persistence to stay connected.

Tools:

- RATs
- Rootkits

5️⃣ Covering Tracks

Hiding evidence of hacking.

Methods:

- Log deletion
- Timestomping
- File hiding

Advantages of Ethical Hacking

- Helps secure systems proactively
- Protects against cybercriminals
- Reduces risk of data breaches
- Improves organizational security posture
- Ensures regulatory compliance
- Builds trust with customers
- Identifies vulnerabilities before exploitation

Disadvantages / Risks

- Testing may unintentionally break systems
- Requires highly skilled professionals
- Costly for organizations

- Sensitive data exposure risk
- If misused, hacker may turn malicious

Scope of Ethical Hacking (Major Areas)

Ethical hacking has a **very wide scope** because almost every digital system can be hacked. Below are the core areas:

1️⃣ Network Security

Ethical hackers test:

- Routers, switches, firewalls
- Open ports & services
- Network protocols (TCP/IP, DNS, DHCP)
- VPN security
- Wi-Fi networks

Goal:
Prevent unauthorized access and network attacks.

2️⃣ Web Application Security

Testing the security of:

- Websites
- Web servers
- API endpoints
- Cloud-based apps

Common attacks tested:

- SQL Injection
- Cross-Site Scripting (XSS)
- CSRF
- Broken Authentication

3️⃣ System Security

Testing operating systems:

- Windows
- Linux
- macOS

Checks include:

- Privilege escalation
- Misconfigurations
- Password weaknesses
- Backdoors

4️⃣ Wireless Security

Testing Wi-Fi networks:

- WPA2/WPA3 cracking
- Evil twin attacks
- Rogue access points

Goal:
Prevent unauthorized wireless access.

5️⃣ Mobile Application Security

Testing Android/iOS apps for:

- Data leakage
- Insecure API calls
- Root/jailbreak vulnerabilities
- Authentication issues

6️⃣ Cloud Security

Ethical hacking in:

- AWS
- Azure
- Google Cloud
- SaaS, PaaS, IaaS

Tests include:

- Misconfigured buckets
- Access control flaws
- Key mismanagement

7️⃣ Database Security

Testing:

- SQL/NoSQL database configuration
- Database injection attacks
- Weak credentials

8️⃣ IoT & Embedded Device Security

Ethical hacking of:

- Smart home devices
- Cameras
- Sensors
- Industrial IoT
- Medical IoT

Goal:
Prevent device hijacking & data leaks.

9️⃣ Social Engineering Testing

People are the weakest link.

Testing includes:

- Phishing simulations
- Impersonation attempts
- Baiting attacks

🔟 Critical Infrastructure Security

Testing:

- Power grids
- Transportation systems
- Water supply systems
- Oil and gas control systems

Goal:
Prevent cyberwarfare & national-level attacks.

Types of Ethical Hackers

1. WHITE HAT HACKERS

Definition

White Hat Hackers are ethical hackers who use hacking skills legally, with proper authorization, to test and improve the security of systems.

➡️ *They hack to protect*, not to harm.

Characteristics

- Work with permission
- Purpose is to secure systems
- Follow laws and ethical guidelines
- Found in companies, cybersecurity firms, government agencies
- Follow Rules of Engagement (RoE)
- Report vulnerabilities responsibly
- Never misuse data

What They Do

- Penetration testing
- Vulnerability assessments
- Security audits
- Network and application testing
- Fixing vulnerabilities
- Creating secure configurations
- Participating in bug bounties

Examples

- Security engineer testing a bank's website
- HackerOne bug bounty researchers
- Ethical hackers working for Microsoft, Google, NASA

Advantages

- Prevent cyberattacks
- Strengthen system security
- Help companies achieve compliance (PCI-DSS, ISO)

Disadvantages

- Require high skill & experience
- Testing may sometimes crash systems
- Time-consuming and costly

2. BLACK HAT HACKERS

Definition

Black Hat Hackers are malicious hackers who break into systems without permission, intending to steal data, cause damage, or gain financial benefit.

➡️ *They hack illegally and with harmful intent.*

Characteristics

- No authorization
- Intent is malicious: theft, fraud, sabotage
- Use malware, Trojans, exploits
- Violate laws (cybercriminals)
- Remain anonymous
- Sell stolen data on dark web
- Hide tracks using rootkits

What They Do

- Launch malware attacks
- Data breaches
- Ransomware attacks
- Identity theft
- Stealing bank credentials
- Corporate espionage
- Website defacement

Examples

- Hackers behind WannaCry ransomware
- Cybercriminals stealing credit card data
- APT groups performing espionage

Advantages (From learning perspective only)

(Not ethical advantages)

- Helps cybersecurity professionals understand real-world attacks
- Drives improvement in defensive measures

Disadvantages

- Illegal and punishable
- Cause financial and reputational loss
- Endanger personal privacy & national security

3. GREY HAT HACKERS

Definition

Grey Hat Hackers fall between White Hat and Black Hat.
They may break into systems without permission, but not with malicious intent.
They usually report the vulnerability—but sometimes expect rewards.

➡️ *They hack first and ask permission later.*

Characteristics

- No legal authorization
- No harmful intent, but still illegal
- Sometimes violate ethics accidentally
- Try to help organizations by reporting vulnerabilities
- May perform "ethical but illegal" hacks

What They Do

- Scan websites for vulnerabilities without permission
- Report discovered bugs to the company
- Participate in research
- Sometimes exploit vulnerabilities to prove their point

Examples

- Hacker finds a bug in a company website without permission, reports it later
- Independent researchers who do unauthorized scans

Advantages

- They often discover serious vulnerabilities early
- Help improve security (informally)

Disadvantages

- Still illegal because they lack permission
- Can unintentionally harm systems
- Organizations may interpret actions as attacks
- May lead to legal trouble even with good intentions

Other Types of Hackers

There are generally 7 types of Hackers, after the main 3 types, they are:

- **Script Kiddies**: They are the most dangerous people in terms of hackers. A Script kiddie is an unskilled person who uses scripts or downloads tools available for hacking provided by other hackers. They attempt to attack computer systems and networks and deface websites. Their main purpose is to impress their friends and society. Generally, Script Kiddies are juveniles who are unskilled about hacking.
- **Green Hat Hackers**: They are also amateurs in the world of hacking but they are bit different from script kiddies. They care about hacking and strive to become full-blown hackers. They are inspired by the hackers and ask them few questions about. While hackers are answering their question they will listen to its novelty.
- **Blue Hat Hackers**: They are much like the white hat hackers, they work for companies for security testing of their software right before the product launch. Blue hat hackers are outsourced by the company unlike white hat hackers which are employed by the (part of the) company.
- **Red Hat Hackers:** They are also known as the eagle-eyed hackers. Like white hat hackers, red hat hackers also aims to halt the black hat hackers. There is a major difference in the way they operate. They become ruthless while dealing with malware actions of the black hat hackers. Red hat hacker will keep on attacking the hacker aggressively that the hacker may know it as well have to replace the whole system.
- **State/Nation Sponsored Hackers:** State or Nation sponsored hackers are those who are appointed by the government to provide them cybersecurity and to gain confidential information from other countries to stay at the top or to avoid any kind of danger to the country. They are highly paid government workers.
- **Hacktivist:** These are also called the online versions of the activists. Hacktivist is a hacker or a group of anonymous hackers who gain unauthorized access to government's computer files and networks for further social or political ends.
- **Malicious Insider or Whistleblower**: A malicious insider or a whistleblower could be an employee of a company or a government agency with a grudge or a strategic employee who becomes aware of any illegal activities happening within the organization and can blackmail the organization for his/her personal gain.

📘 THREATS AND ATTACK VECTORS

A **Cyber Threat** is any potential danger or harmful event that can exploit a vulnerability and cause damage to data, systems, or networks.

In simple words:

➡️ *A threat is anything that can cause harm to a system.*

An **Attack Vector** is the path, method, or technique used by cyber attackers to exploit vulnerabilities and deliver threats.

In simple words:

➡️ *An attack vector is the route attackers use to enter a system.*

Examples: phishing email, open port, weak password.

Relationship Between Threat, Vulnerability, and Attack Vector

Concept	Meaning	Example
Threat	Potential harm	Malware, hacker
Vulnerability	Weakness	Weak password
Attack Vector	Method used to exploit	Brute-force attack

➡️ Threat uses an **attack vector** to exploit a **vulnerability**.

Types of Cyber Threats

Below are the major categories:

1️⃣ Malware Threats

Includes viruses, worms, Trojans, ransomware, spyware.

Example:

WannaCry ransomware attack.

2️⃣ Social Engineering Threats

Tricking users into revealing information.

Examples:

Phishing, vishing, baiting.

3️⃣ Network-based Threats

Target network protocols and devices.

Examples:

DDoS attacks, sniffing, spoofing, MITM.

4️⃣ Web Application Threats

Attacks on web apps due to coding flaws.

Examples:

SQL injection, XSS, CSRF.

5️⃣ Insider Threats

Employees or trusted individuals causing harm.

Types:

- Malicious insider
- Negligent insider
- Compromised insider

6️⃣ Advanced Persistent Threats (APT)

Highly skilled, long-term cyberattacks by nation-states.

Examples:

Stuxnet, SolarWinds attack.

Types of Attack Vectors

Attack vectors are the **specific methods** used to deliver threats.

1️⃣ Phishing Emails

Most common attack vector.

Used to:

- Steal passwords
- Deliver malware
- Trick users

2️⃣ Malicious Attachments / Downloads

Files that install malware when opened.

3️⃣ Weak or Stolen Credentials

Attackers use:

- Brute-force
- Password spraying
- Credential stuffing

4️⃣ Unpatched Software

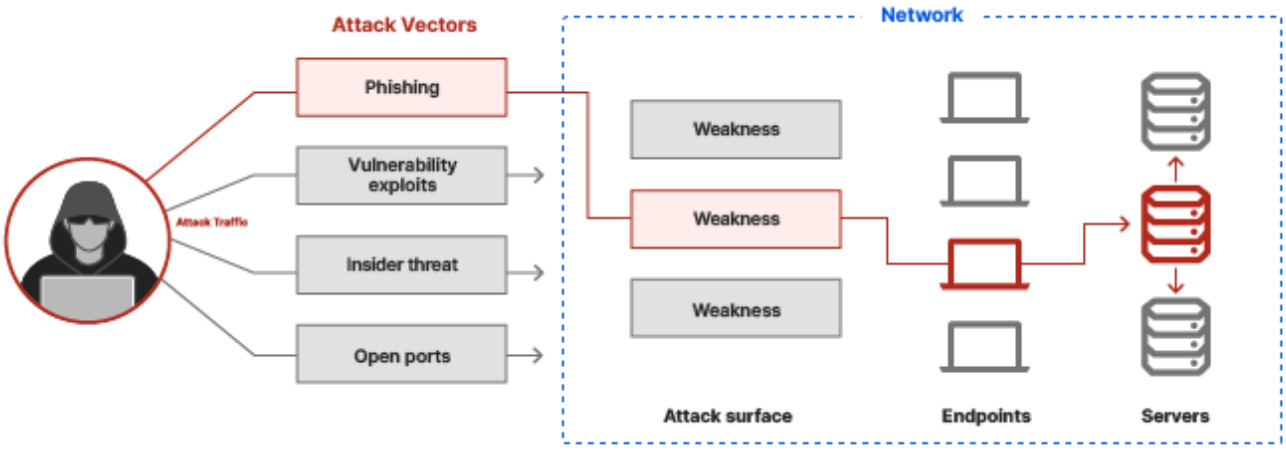
Exploit kits attack outdated software.

5️⃣ Social Engineering Techniques

Impersonation, pretexting, baiting.

6️⃣ Insecure Networks (Public Wi-Fi)

Used for MITM, sniffing attacks.



📘 INFORMATION ASSURANCE (IA)

Information Assurance (IA) refers to the practice of managing risks related to the use, processing, storage, and transmission of information by ensuring its **Confidentiality, Integrity, Availability, Authentication, and Non-repudiation (CIANA model).**

In simple words:
➡️ *Information Assurance ensures information is safe, reliable, and available, even in the presence of threats.*

It focuses not only on preventing attacks but also on **maintaining trust and resilience** in information systems.

Key Components of Information Assurance

1️⃣ Risk Management

Identifying, analyzing, and reducing risks.

2️⃣ Security Controls

Technical, administrative, and physical protections.

3️⃣ Business Continuity & Disaster Recovery

Ensuring operations continue even after attacks or failures.

4️⃣ Compliance & Legal Requirements

Following standards and regulations:

- ISO 27001
- GDPR
- HIPAA
- NIST

5️⃣ Policy Development

Creating rules for:

- Password usage
- Data handling
- Incident response
- Access permissions

6️⃣ Incident Detection & Response

Monitoring systems and reacting to cyber incidents.

7️⃣ Training & Awareness

Educating employees to avoid human errors.

Benefits of Information Assurance

- Reduces cyber risk
- Ensures business continuity
- Protects sensitive data
- Builds customer trust
- Ensures compliance with laws
- Prevents financial and reputational loss
- Enhances organizational resilience

Challenges in Information Assurance

- Constantly evolving threats
- Human errors
- Managing complex systems
- Cost of implementation
- Insider threats
- Need for continuous monitoring
- Difficulty maintaining availability and security simultaneously

Information Assurance Lifecycle

1. Identify assets

Know what needs protection.

2. Assess risks

Identify vulnerabilities & threats.

3. Develop IA strategy

Create policies and controls.

4. Implement controls

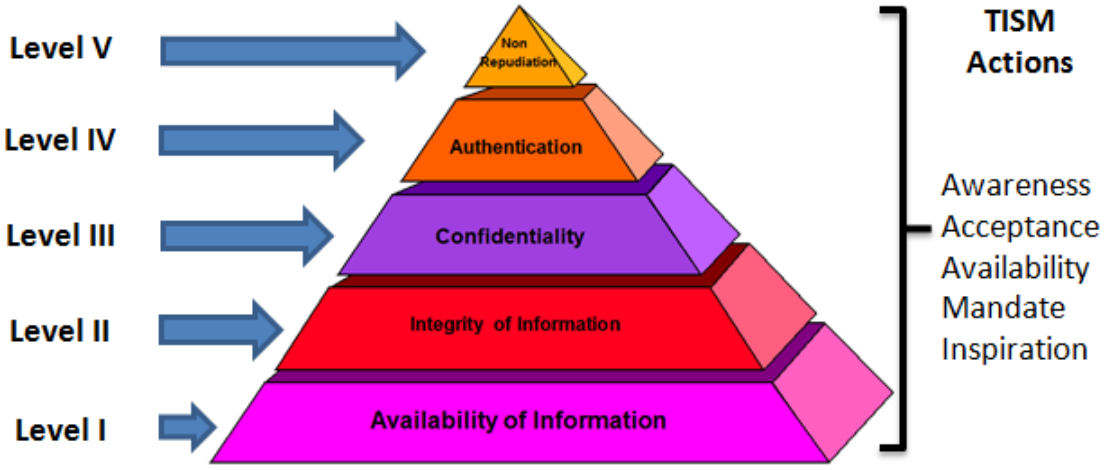
Firewalls, encryption, access control.

5. Monitor and detect incidents

Continuous monitoring.

6. Respond and recover

Incident response & disaster recovery.



📘 THREAT MODELLING

Threat Modelling is a structured process used to identify, analyze, prioritize, and address potential security threats, vulnerabilities, and attack paths in a system.

In simple words:

➡️ *Threat modelling helps us to think like an attacker and find weaknesses before they do.*

It is used during **design, development, and testing phases** of software and network systems.

Purpose of Threat Modelling

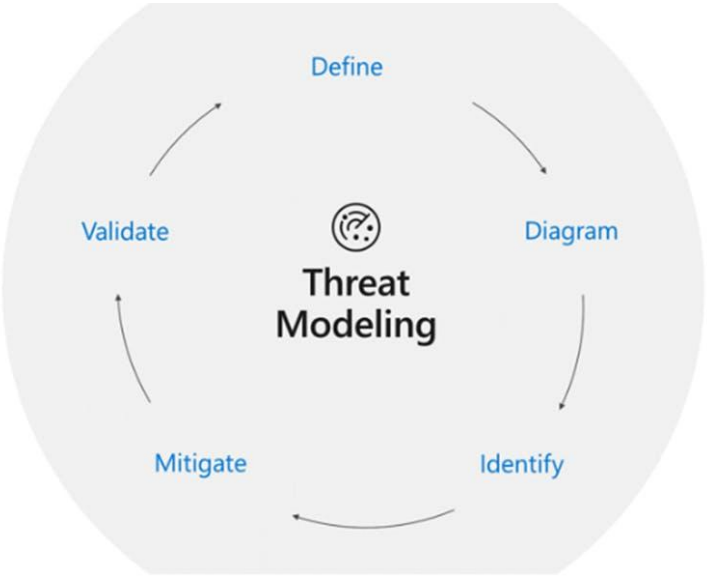
- Identify potential threats early
- Evaluate the security posture of a system
- Prioritize high-risk areas
- Reduce cost of fixing vulnerabilities later
- Improve design and architecture of systems
- Understand how attackers might attack
- Define effective mitigation strategies
- Ensure compliance with standards (ISO 27001, NIST, OWASP)

When Is Threat Modelling Done?

- During software development (SDLC)
- Before deploying new applications
- When making architectural changes
- While designing network infrastructure

- During cloud migration
- For IoT, mobile apps, APIs

Steps in Threat Modelling (Standard Process)



Step 1: Identify Security Objectives

What are you trying to protect?

Example: Protect user login, prevent data leakage.

Step 2: Create Architecture Overview

Understand system components:

- Servers
- Databases
- APIs
- Data flows
- Trust boundaries

Usually represented using **DFD (Data Flow Diagrams)**

Step 3: Identify Threats

Think like an attacker.

Use frameworks like STRIDE or MITRE ATT&CK.

Step 4: Identify Vulnerabilities

Find weaknesses in code, network, configuration.

Step 5: Assess Risks

Calculate:

- Likelihood of attack
- Impact of attack
- Business consequences

Step 6: Prioritize Threats

Fix the highest-risk threats first.

Step 7: Mitigation & Controls

Apply:

- Encryption
- Authentication
- Input validation
- Firewall rules
- Patching
- Logging

Step 8: Review & Iterate

Threat modelling is **continuous**, not one-time.

📘 ENTERPRISE INFORMATION SECURITY ARCHITECTURE (EISA)

Enterprise Information Security Architecture (EISA) is a **strategic framework** that defines how security controls, policies, technologies, and processes should be designed, implemented, and managed across an entire organization.

In simple words:
➡️ *EISA is the blueprint for securing the entire enterprise—its data, systems, networks, people, and operations.*

It ensures that business goals and security requirements are aligned.

Purpose of EISA

- Provide a **structured approach** for securing enterprise assets
- Create **alignment** between security and business objectives
- Reduce security gaps and risks

- Ensure consistency in security policies and controls
- Support compliance and governance requirements
- Enable secure digital transformation
- Enhance long-term scalability and resilience

Why EISA is Needed

- Growing complexity of IT environments
- Increase in cyber threats
- Need for centralized governance
- Integration of cloud, IoT, and mobile systems
- Regulatory pressures (GDPR, HIPAA, ISO 27001)
- To ensure enterprise-wide *standardization*

EISA Development Process (High-Level Steps)

Step 1: Understand Business Goals

Align security with business objectives.

Step 2: Identify Assets & Risks

Classify systems and data based on sensitivity.

Step 3: Define Security Requirements

Confidentiality, integrity, availability, compliance needs.

Step 4: Design Security Architecture

Plan network, cloud, IAM, encryption, monitoring, etc.

Step 5: Implement Security Controls

Apply technical and administrative security measures.

Step 6: Monitor, Audit & Improve

Continuous assessment and adaptation.

Benefits of Enterprise Information Security Architecture

✓ 1. Consistency Across the Organization

Ensures uniform security controls across all departments.

✓ 2. Reduced Risk

Proactively identifies vulnerabilities and threats.

✓ 3. Better Governance

Supports policies, compliance, and audits.

✓ 4. Scalability

EISA grows with the business.

✓ 5. Cost Efficiency

Prevents duplication, simplifies security operations.

✓ 6. Enhances Incident Response

Clear architecture supports faster detection & response.

✓ 7. Supports Digital Transformation

Secure adoption of cloud, mobility, IoT.

Challenges in Implementing EISA

- Complex for large organizations
- Requires top-level management support
- Needs skilled architects
- Integration issues with legacy systems
- Continuous updates needed
- Time and cost constraints

Example Scenario of EISA in Real Use

Company: A multinational e-commerce company

Security Needs:

- Protect customer data (card, personal info)
- Secure APIs and microservices
- Maintain PCI-DSS compliance

EISA Implementation Includes:

- Zero-trust network
- Multi-factor authentication
- Secure payment gateways
- Cloud security controls
- SOC for continuous monitoring

- Backup and disaster recovery

📘 VULNERABILITY ASSESSMENT & PENETRATION TESTING (VAPT)

VAPT (Vulnerability Assessment and Penetration Testing) is a combination of two security services used to identify security weaknesses in systems, networks, and applications.

It ensures both:

- **Finding vulnerabilities** (assessment)
- **Exploiting them safely** (penetration testing)

1️⃣ Vulnerability Assessment

A process of identifying, scanning, and analyzing **known vulnerabilities** in systems, applications, or networks.

➡️ *Focus: Finding weaknesses, not exploiting them.*
➡️ Objective: **Breadth over depth** (identify as many issues as possible).

2️⃣ Penetration Testing (Pentesting)

A controlled cyberattack performed by authorized professionals to **exploit vulnerabilities** and determine the real-world impact.

➡️ *Focus: Exploitation.*
➡️ Objective: **Depth over breadth** (test how far an attacker can go).

Why VAPT Together?

Neither VA nor PT alone is sufficient.

- VA finds weaknesses but cannot show actual damage
- PT shows impact but may miss some vulnerabilities

Together → Complete security evaluation.

Steps in Vulnerability Assessment

Step 1: Asset Identification

Identify systems, servers, applications.

Step 2: Vulnerability Scanning

Use automated tools to detect weaknesses.

Step 3: Vulnerability Analysis

Evaluate:

- Severity

- Impact
- Likelihood

Step 4: Prioritization

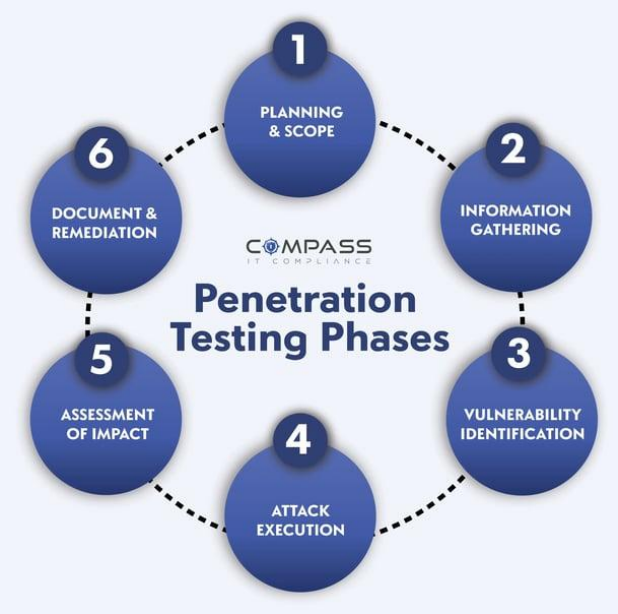
Rank vulnerabilities (Critical, High, Medium, Low).

Step 5: Reporting & Remediation

Provide:

- Risk ratings
- Fix recommendations

Steps in Penetration Testing



1️⃣ Planning & Reconnaissance

Gather information (OSINT, footprinting).

2️⃣ Scanning & Enumeration

Identify open ports and vulnerabilities.

3️⃣ Gaining Access

Exploit vulnerabilities (SQL injection, weak passwords).

4️⃣ Maintaining Access

Install backdoors or persistence (simulates real attackers).

5️⃣ Covering Tracks (Optional)

Hide evidence (ethical testers don’t harm logs unless allowed).

6️⃣ Reporting

Document exploit steps, evidence, and remediation.

Types of Penetration Testing

1. Black-Box Testing

Tester has **no internal knowledge**.

Simulates real-world attack.

2. White-Box Testing

Tester has **full access** (source code, credentials).

Finds deep architectural flaws.

3. Grey-Box Testing

Tester has **partial knowledge**.

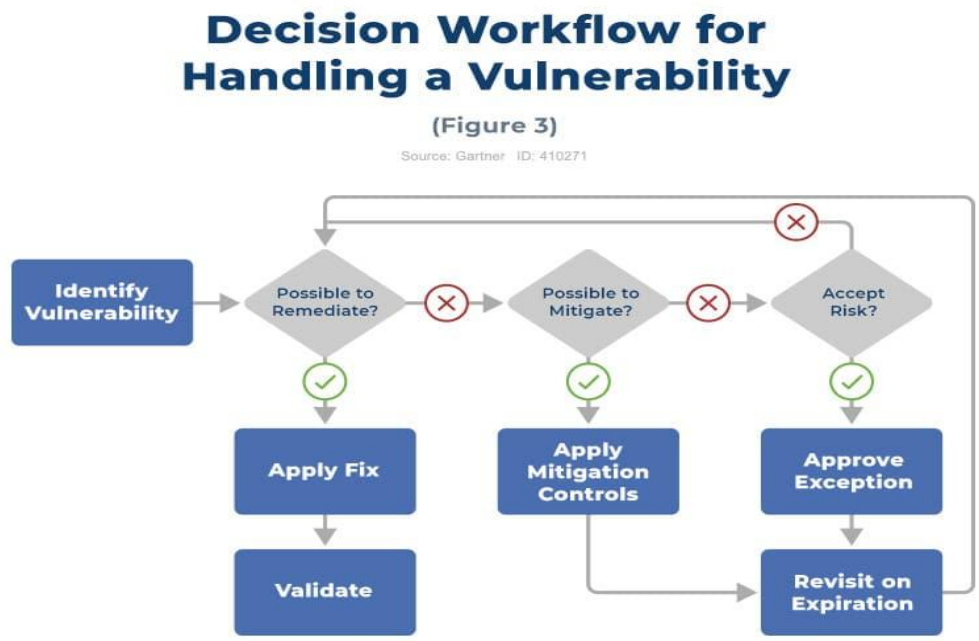
Balanced approach.

4. Network Penetration Testing

Tests internal/external networks.

5. Web Application Penetration Testing

Focus on OWASP vulnerabilities (XSS, SQLi).



Benefits of VAPT

- Detect vulnerabilities before attackers do
- Prevent data breaches
- Strengthen security posture
- Ensure compliance (PCI-DSS, ISO 27001, GDPR)
- Improves customer trust
- Cost-effective security improvement
- Identifies real-world exploitability of issues

Challenges / Limitations of VAPT

- Time-consuming in large networks
- Requires skilled professionals
- Cannot guarantee 100% security
- May cause temporary system disruption
- Tools may detect false positives/negatives
- Scope must be clearly defined

Feature	Vulnerability Assessment (VA)	Penetration Testing (PT)
Objective	Find and list vulnerabilities	Exploit vulnerabilities to determine impact
Focus	Breadth (many issues)	Depth (few issues but deeply tested)
Approach	Mostly automated scanning	Manual, creative exploitation
Risk Level	Low	Medium (because exploitation occurs)
Outcome	Report of vulnerabilities with severity	Proof-of-concept attacks, impact demonstration
Testing Scope	Wide	Narrow and deep
Skill Requirement	Moderate	High (ethical hacker expertise)
Tools Used	Nessus, OpenVAS, Qualys, Nmap	Burp Suite, Metasploit, SQLMap
Time Required	Short	Longer
Real-World Attack Simulation	No	Yes
Fix Validation	Not always tested	Yes, often verifies if exploit is patched
Cost	Lower	Higher

Social Engineering is the technique of manipulating people into revealing confidential information or performing actions that compromise security.

In simple words:
➡️ *Attackers hack humans, not machines.*

Social engineering relies on **psychological manipulation**, not technical hacking.

Main Types of Social Engineering Attacks

1️⃣ Phishing

Definition:

Fraudulent emails/messages that appear legitimate, tricking users into clicking malicious links or revealing credentials.

Targets:

- Passwords
- Banking details

- Personal information

Example:

Email pretending to be from a bank asking to "verify account."

2️⃣ Spear Phishing

Definition:

A **targeted** phishing attack aimed at a specific individual or organization.

Characteristics:

- Personalized messages
- Higher success rate

Example:

Email to a company employee pretending to be the CEO.

3️⃣ Whaling

Definition:

A form of spear phishing targeting **top executives** like CEOs, CFOs.

Goal:

High-value financial fraud.

4️⃣ Vishing (Voice Phishing)

Definition:

Social engineering via **phone calls**.

Example:

Scammer pretending to be bank support asking for OTP/PIN.

5️⃣ Smishing (SMS Phishing)

Definition:

Phishing via text messages (SMS).

Example:

"Your package is on hold. Click here to update details."

6️⃣ Baiting

Definition:

Offering something irresistible (free item/download) to trick victims.

Physical Example:

- USB drive labeled “Confidential – Salaries” left in parking lot.

Online Example:

- Free movie download with hidden malware.

7️⃣ Pretexting

Definition:

Attacker creates a **fake scenario (pretext)** to trick users into giving information.

Example:

Attacker pretending to be:

- HR asking for employee data
- IT support requesting login details
- Police officer demanding identity proof

8️⃣ Tailgating (Piggybacking)

Definition:

Entering a restricted area by **following an authorized person**.

Example:

Attacker carrying boxes so employee holds door open.

📘 INSIDER ATTACK

An **Insider Attack** occurs when someone **within an organization**—such as an employee, contractor, partner, or former staff—misuses their authorized access to compromise the confidentiality, integrity, or availability of systems, data, or networks.

In simple words:
➡️ *An insider attack happens when someone from inside the organization causes harm intentionally or accidentally.*

Insiders are dangerous because they already have **trust, access, and knowledge** of the system.

Characteristics of Insider Attacks

- Performed by users with **legitimate access**
- Hard to detect due to trusted identity
- Often cause more damage than external attackers

- Motivated by personal benefit, revenge, negligence, or coercion
- Can be accidental or intentional

Why Insider Attacks Are Difficult to Detect

- Insiders already have legitimate access
- Activities look normal at first glance
- Attackers know security weaknesses
- They understand system architecture
- Lack of monitoring for internal behavior

Prevention and Defense Against Insider Attacks

1 Zero Trust Model

Never trust any user blindly.
Continuous verification is required.

2 Least Privilege Principle

Give only the minimum required access.

3 Privileged Access Management (PAM)

Monitor and control admin-level privileges.

4 User Behavioral Analytics (UBA / UEBA)

Detect abnormal behavior using AI.

5 Monitoring & Logging

Track file access, login activity, data transfers.

6 Data Loss Prevention (DLP) Tools

Prevent copying or sending sensitive data.

7 Employee Awareness Training

Reduce negligent insider threats.

8 Background Checks

Verify employee history before hiring.

9 Access Review and Revocation

Remove access immediately after resignation or role change.

10 Enforce Strong Policies

- Acceptable use policy
- Password policy
- Remote access policy

Social Engineering Targets

Attackers choose targets based on **access level, privileges, and psychological vulnerability**.

Below are the main categories:

1️⃣ Employees (General Staff)

Why targeted?

- Lack of security awareness
- Easy to trick via phishing emails
- Access to internal systems

Example attacks:

- Phishing emails
- USB baiting
- Fake tech support calls

2️⃣ High-Value Individuals (VIPs / Executives)

Also called **Whaling Targets**.

Why targeted?

- Access to sensitive data
- Approval authority for financial transactions

Example attacks:

- CEO fraud
- Business Email Compromise (BEC)

3️⃣ IT Administrators / Technical Staff

Why targeted?

- High system privileges
- Ability to install software or modify configurations

Example attacks:

- Privilege escalation via phishing

- Fake "patch update" requests

4️⃣ New Employees

Why targeted?

- Unaware of company processes
- More likely to trust others

Example attacks:

- Pretexting: “I’m from IT, give me your login.”

5️⃣ Help Desk / Customer Support Staff

Why targeted?

- Trained to be helpful
- Often share temporary passwords or reset access

Example attacks:

- Impersonation: fake customer seeking password reset

6️⃣ Financial/Accounts Department

Why targeted?

- Handle money and transactions

Example attacks:

- Invoice fraud
- CEO payment instructions

7️⃣ Physical Security Guards / Receptionists

Why targeted?

- First point of entry
- Often trust visitors

Example attacks:

- Tailgating
- Fake repairman entry

8️⃣ Customers or External Users

Why targeted?

- Weak personal cybersecurity

- Easy to impersonate or scam

Example attacks:

- Smishing (SMS fraud)
- Fake customer service calls

9️⃣ Suppliers / Third-Party Vendors

Why targeted?

- Lower security than the main organization
- Provide backdoor access

Example attacks:

- Watering hole attacks
- Supply chain attacks

Defence Strategies Against Social Engineering

1️⃣ Security Awareness Training

Most effective defense.

Includes:

- Phishing simulation
- Password awareness
- Safe internet practices
- Recognizing scams
- Reporting suspicious activity

2️⃣ Zero Trust Security Model

Never trust; always verify.

Features:

- Continuous authentication
- Device verification
- Least privilege access
- Strong identity management

3️⃣ Email & Web Filtering

Blocks:

- Malicious attachments
- Phishing links
- Fake domain names

Tools:

- Secure Email Gateways
- Anti-spam filters

4️⃣ Multi-Factor Authentication (MFA)

Even if password is stolen → attacker cannot log in.

Methods:

- OTP
- Biometrics
- Smart cards

5️⃣ Strong Password and Access Policies

Includes:

- Password rotation
- Complex passwords
- Avoiding password reuse
- Disabling unused accounts

6️⃣ Data Loss Prevention (DLP)

Prevents:

- Unauthorized data transfer
- Copying to USB
- Sending sensitive info outside network

7️⃣ Behaviour Monitoring / UEBA

User and Entity Behavior Analytics detects:

- Unusual logins
- Sudden large data transfers
- abnormal privilege use

8️⃣ Physical Security Controls

Includes:

- Smart ID badges
- Biometric entry
- CCTV cameras
- Security guards

Prevents:

- Tailgating
- Impersonation

📘 CYBER FORENSICS

Cyber Forensics, also called **Digital Forensics**, is the scientific process of collecting, analyzing, preserving, and presenting digital evidence from electronic devices to investigate cybercrimes.

In simple words:
➡️ *Cyber forensics is the method of finding and proving what happened on a digital device.*

Cyber forensics helps law enforcement, organizations, and courts in solving digital crimes.

Objectives of Cyber Forensics

- Identify, collect, and preserve digital evidence
- Recover deleted, hidden, or encrypted data
- Trace cybercriminal activities
- Maintain chain of custody for legal acceptance
- Support legal proceedings
- Prevent future cyberattacks
- Understand how the attack happened

Characteristics of Digital Evidence

Digital evidence must be:

✓ **Authentic – from a legitimate source**

✓ **Accurate – exact representation of original data**

✓ **Reliable – obtained using proper methods**

✓ **Complete – includes all relevant information**

✓ **Admissible – acceptable in court**

Types of Cyber Forensics

1️⃣ Computer Forensics

Deals with PCs, laptops, hard drives.

Includes:

- File recovery
- OS analysis
- Registry analysis

- Malware investigation

2️⃣ Mobile Forensics

Deals with smartphones and tablets.

Includes:

- Call logs, SMS, WhatsApp data
- GPS data
- App usage
- Deleted file recovery

3️⃣ Network Forensics

Analyzes network packets and traffic.

Includes:

- Packet capture
- Intrusion detection
- Session reconstruction
- Tracking unauthorized activity

4️⃣ Cloud Forensics

Investigates crimes involving cloud platforms.

Challenges:

- Multi-tenancy
- Remote storage
- Shared responsibility model

5️⃣ Email Forensics

Used for phishing, BEC fraud, spam, and impersonation investigation.

Includes:

- Header analysis
- Metadata
- IP tracking

6️⃣ Malware Forensics (Malware Analysis)

Investigates malicious software behavior.

Includes:

- Static and dynamic analysis
- Payload identification

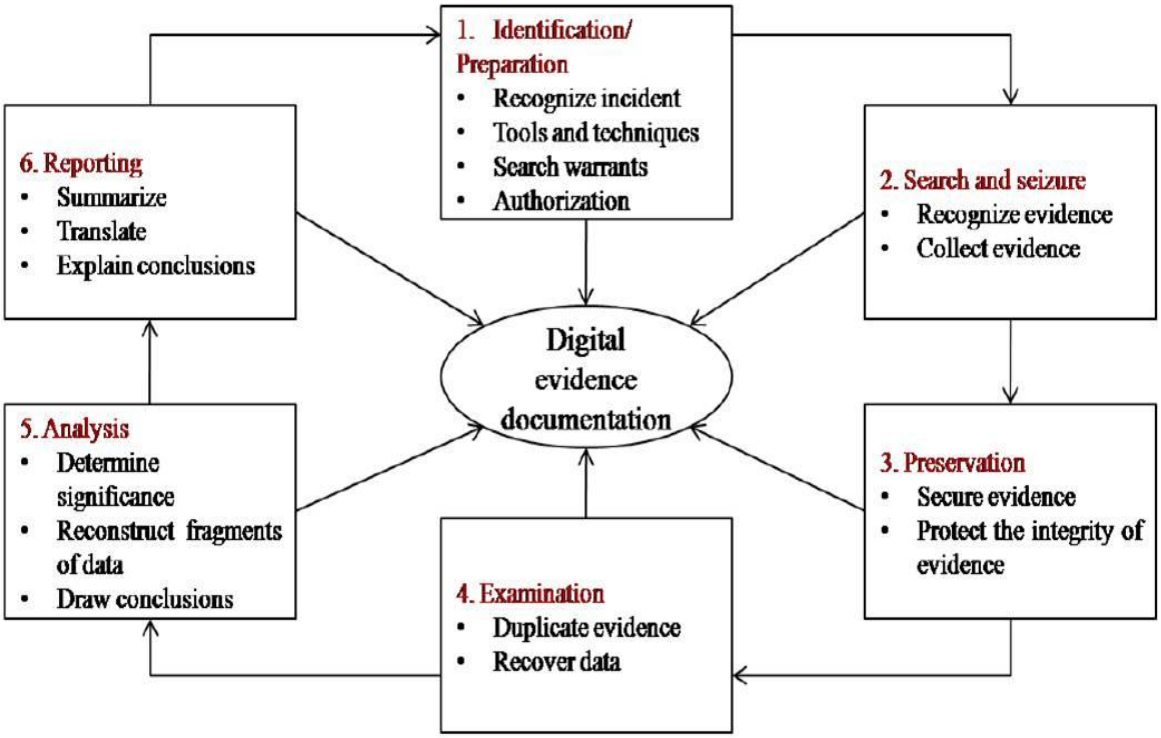
7 Database Forensics

Investigates database logs, queries, and breaches.

8 IoT Forensics

Deals with smart devices, sensors, CCTV systems.

Cyber Forensics Process (Stages)



1 Identification

Recognize incident and potential evidence.

Example:

- Logs
- Disk drives
- Emails
- Network traces

2 Preservation

Protect evidence from being altered.

Includes:

- Disk imaging
- Write blockers
- Maintaining chain of custody

3️⃣ Collection

Gather digital evidence in a legally acceptable way.

Tools:

- EnCase
- FTK Imager
- dd (Linux)

4️⃣ Examination

Extract relevant information using forensic techniques.

Examples:

- Recover deleted files
- Analyze registry entries
- Search keywords

5️⃣ Analysis

Interpret results and discover what happened.

Includes:

- Timeline analysis
- Attack mapping
- Identifying suspects

6️⃣ Documentation / Reporting

Prepare a structured report for legal or organizational use.

7️⃣ Presentation

Present findings in court or to management in a clear, understandable way.

Common Tools Used in Cyber Forensics

Computer Forensics

- EnCase
- FTK (Forensic Toolkit)
- Autopsy / Sleuth Kit
- X-Ways

Network Forensics

- Wireshark
- tcpdump
- NetworkMiner

Mobile Forensics

- Cellebrite UFED
- Oxygen Forensic Suite

Malware Analysis

- IDA Pro
- OllyDbg
- Cuckoo Sandbox

Log Analysis Tools

- Splunk
- ELK Stack

Real-World Cyber Forensics Cases

✓ Sony Pictures Hack (2014)

Digital forensics traced attack back to Lazarus Group (North Korea).

✓ BTK Serial Killer Case

A deleted file and metadata from a floppy disk led to identification of the criminal.

✓ Email Fraud Cases

Header analysis often reveals fraudsters' IP addresses.

✓ Financial Cybercrime Investigations

Forensics identifies transaction trails and attackers behind banking fraud.

Importance of Cyber Forensics in Organizations

- Helps in incident response

- Supports compliance requirements
- Prevents future cyberattacks
- Identifies insider threats
- Builds trust with customers
- Supports legal resolution
- Strengthens cybersecurity posture

Challenges in Cyber Forensics

- Huge amount of digital data
- Strong encryption
- Anti-forensic techniques
- Cloud-based storage complexity
- Jurisdiction issues (data stored in other countries)
- Rapidly evolving technology
- Maintaining chain of custody

📘 COMPUTER EQUIPMENT AND ASSOCIATED STORAGE MEDIA

Computer equipment and storage media are essential components used to **process, store, and retrieve digital data**.
In **cyber forensics**, understanding these devices is crucial because they often contain **digital evidence**.

Computer Equipment (Digital Devices)

Computer equipment includes all hardware devices capable of storing or processing electronic information.

1️⃣ Personal Computers (Desktop Systems)

Components:

- CPU
- RAM
- Hard drives (HDD/SSD)
- Motherboard

Forensic Relevance:

- User files, browsing history, logs, deleted data.

2️⃣ Laptops / Notebooks

Characteristics:

- Portable, contain battery
- Often include SSDs (harder to recover deleted data)

Forensic Relevance:

- Emails, chat logs, documents, Wi-Fi logs.

3️⃣ Servers

Used for:

- Website hosting
- Databases
- Enterprise applications

Forensic Relevance:

- Large-scale logs, user accounts, system events, audit trails.

4️⃣ Mobile Devices (Smartphones/Tablets)

Features:

- Internal flash storage
- Apps, messages, GPS, call logs

Forensic Relevance:

- Key evidence in modern investigations.

5️⃣ Network Devices

Includes:

- Routers
- Switches
- Firewalls
- Access points

Forensic Relevance:

- Logs, routing tables, NAT records.

6️⃣ IoT Devices

Examples:

- Smart TVs
- CCTV cameras
- Wearables
- Home assistants

Forensic Relevance:

- Video files, GPS logs, voice recordings.

7️⃣ Peripheral Devices

Includes:

- Printers
- Scanners
- External webcams

Forensic Relevance:

- Cached print jobs, scanned images

8️⃣ Removable Storage Devices

Explained in detail below (as associated storage media).

Associated Storage Media

Storage media refers to devices used to **store digital data**, temporarily or permanently.

They are classified as:

- **Primary Storage** (volatile – for processing)
- **Secondary Storage** (non-volatile – for long-term storage)
- **Removable Storage**

A. Primary Storage (Volatile)

1️⃣ RAM (Random Access Memory)

Characteristics:

- Volatile (data lost on shutdown)
- Stores active processes and data

Forensic Relevance:

- Can contain passwords
- Network session data

- Running malware behavior

2️⃣ Cache Memory

Types:

- CPU cache (L1, L2, L3)
- Browser cache

Forensic Relevance:

- Frequently accessed data
- Browser cache reveals user activity

B. Secondary Storage (Non-Volatile)

1️⃣ Hard Disk Drives (HDD)

Characteristics:

- Magnetic storage
- Stores OS, applications, user files
- Large capacity

Forensic Relevance:

- High chance of recovering deleted files
- Contains long-term logs and browsing history

2️⃣ Solid State Drives (SSD)

Characteristics:

- Flash-based
- Faster than HDD
- TRIM feature makes deleted data recovery harder

Forensic Relevance:

- Harder to recover deleted data
- Still contains important user information

3️⃣ Hybrid Drives (SSHD)

Combination of HDD + SSD.

4️⃣ Optical Media

Types:

- CD
- DVD
- Blu-ray

Characteristics:

- Read-only or rewritable

Forensic Relevance:

- Often used to store pirated content, backup files.

5️⃣ Magnetic Tapes

Characteristics:

- High-capacity archival storage
- Used in enterprises for backup systems

Forensic Relevance:

- Contains historical data backups.

C. Removable Storage Media

1️⃣ USB Flash Drives

Features:

- Portable, easy to hide
- Common in insider attacks

Forensic Relevance:

- Data exfiltration
- Malware transfer
- Hidden encrypted containers

2️⃣ Memory Cards

Types:

- SD, microSD, CF
- Used in cameras, phones, drones

Forensic Relevance:

- Photos, videos, app data

3️⃣ External Hard Drives

Features:

- Large portable storage
- Often used for backups

Forensic Relevance:

- Important evidence in data theft cases

4️⃣ Cloud Storage (Logical Storage)

Examples:

- Google Drive
- OneDrive
- Dropbox
- AWS S3

Forensic Relevance:

- Files stored remotely
- Sync logs
- Access history

5️⃣ Network Attached Storage (NAS)

Shared storage devices connected to LAN.

Forensic Relevance:

- Centralized logs
- User folders
- Encrypted storage

Forensic Importance of Computer Equipment & Storage Media

- Digital evidence resides on these devices
- Helps reconstruct user activities
- Stores logs and metadata
- Contains deleted or hidden data
- Provides clues for cybercrime investigations

Challenges in Digital Forensics Related to Storage Media

- Encryption makes data recovery difficult

- SSD TRIM destroys deleted data
- Cloud storage jurisdiction issues
- Huge storage capacity causes long analysis time
- Anti-forensic techniques (wiping, obfuscation)
- Volatile memory requires immediate capture

Key Responsibilities of a Forensics Investigator

1 Securing the Crime Scene (Digital or Physical)

- Identify devices involved
- Isolate systems from network (prevent data loss)

2 Identification of Evidence

- Determine which devices, logs, or storage media contain relevant evidence
- Includes computers, mobiles, cloud accounts, networks

3 Preservation of Evidence

- Prevent tampering, corruption, or modification
- Use write blockers
- Create forensic images
- Maintain chain of custody

4 Collection of Evidence

- Collect data systematically
- Follow legal procedures
- Extract logs, files, memory dumps

5 Examination and Recovery

- Recover deleted, hidden, damaged, or encrypted data
- Analyze artifacts (browser history, logs, registry, emails)
- Extract timestamps and metadata

6 Analysis of Evidence

- Reconstruct timeline
- Identify attack methods
- Detect malware behavior

- Determine who performed the action
- Trace IPs, network traffic

7️⃣ Documentation

- Record every step followed
- Maintain investigation logs
- Prepare structured reports

8️⃣ Reporting

- Prepare detailed technical and non-technical reports
- Provide evidence summary
- Include screenshots, logs, findings

9️⃣ Presenting Evidence in Court

- Testify as an expert witness
- Explain technical findings in simple language
- Demonstrate authenticity and integrity of evidence

🔟 Collaboration

Work with:

- Law enforcement agencies
- Cybersecurity teams
- Legal teams
- IT administrators

📘 COLLECTING NETWORK-BASED EVIDENCE

Network-based evidence refers to digital data collected from network devices, traffic flows, logs, and communication channels that can help forensic investigators analyze cyber incidents.

In simple words:
➡️ *It is evidence collected from the network to find how an attack happened, who did it, and what was affected.*

Why Network-Based Evidence Is Important

- Tracks attacker behavior
- Identifies unauthorized access

- Helps reconstruct timeline of attack
- Provides proof of data exfiltration
- Detects malware communication (C2 servers)
- Supports incident response
- Essential for legal investigations

Challenges in Collecting Network Evidence

- High volume of logs
- Encrypted traffic (HTTPS, VPN)
- Dynamic IP addresses (DHCP)
- Distributed cloud environments
- Limited log retention period
- Attackers deleting logs
- Multi-jurisdiction issues (cloud forensics)

Steps in Collecting Network-Based Evidence

Step 1: Identify Relevant Network Sources

Determine which devices contain needed logs:

- Firewalls
- Routers
- SIEM systems
- DNS servers
- Cloud logs

Step 2: Preserve Network Evidence

Methods:

- Export logs using read-only access
- Enable logging at the time of incident
- Save PCAP files immediately

Integrity Protection:

- Hashing (MD5/SHA-256)
- Time-stamping

Step 3: Collect Real-Time Data (Live Capture)

Tools:

- Wireshark
- tcpdump
- Tshark

Useful for:

- Ongoing attack observation

Step 4: Collect Stored Logs

Obtain logs from:

- SIEM (Splunk, ELK)
- Servers
- Firewalls
- Routers

Step 5: Correlate Data

Combine logs from various sources to reconstruct the attack timeline.

Step 6: Analyze Evidence

Look for:

- Suspicious IPs
- Failed login attempts
- Port scans
- Malware traffic patterns
- Data exfiltration signs

Step 7: Document Everything

Record:

- Tools used
- Time of collection
- Hash values
- Source of logs

Step 8: Present Findings

Create:

- Timeline
- Attack flow
- Evidence screenshots
- Summary report

📘 WRITING COMPUTER FORENSICS REPORTS

A **Computer Forensics Report** is a formal document that summarizes the findings of a digital investigation. It contains collected evidence, analysis performed, tools used, results obtained, and conclusions drawn by the forensic investigator.

In simple words:
➡️ *It explains what happened, how it happened, what evidence was found, and what it means—written in a clear, logical, and legally acceptable manner.*

Importance of a Forensics Report

- Provides legally admissible documentation
- Communicates findings to law enforcement or management
- Maintains chain of custody
- Supports expert testimony in court
- Records investigation steps for future reference
- Ensures transparency and accuracy of the investigation

Below is the typical structure used in professional and legal settings:

1️⃣ Cover Page

Includes:

- Case title
- Investigator name
- Organization
- Case number
- Date of report

2️⃣ Executive Summary

High-level overview for non-technical readers.

Includes:

- What happened?
- What was found?
- Overall conclusion

3️⃣ Introduction / Background

Describes:

- Purpose of investigation
- Requester of investigation
- Scope and limitations

4️⃣ Authorization

Shows permission to conduct investigation.

Includes:

- Legal authority
- Search warrant (if applicable)
- Consent forms

5️⃣ Evidence Description

List all evidence collected:

- Devices (laptop, mobile, USB)
- Serial numbers
- Evidence tags
- Physical condition

6️⃣ Chain of Custody Records

Tracks who handled the evidence, when, and where.

Essential for legal admissibility.

7️⃣ Tools and Software Used

Examples:

- EnCase
- FTK
- Autopsy
- Wireshark

- Cellebrite

Also mention:

- Tool version numbers
- Hashing methods (MD5/SHA-256)

8️⃣ Methodology

Explain step-by-step how the investigation was conducted.

Includes:

- Imaging process
- Data recovery techniques
- Keyword searches
- Timeline creation
- Log analysis

9️⃣ Findings / Analysis

This is the **core of the report**.

Includes:

- Deleted files recovered
- Logs extracted
- Email analysis
- Malware behavior
- User activity timeline
- Screenshots and evidence items
- Hash values

Should contain **facts only**, no speculation.

🔟 Conclusion

Summarizes:

- Interpretation of findings
- Whether misconduct occurred
- Final opinion based on evidence

1️⃣1️⃣ Recommendations

(Optional)
Suggest improvements:

- Security controls
- Policies
- Preventive measures

1️⃣2️⃣ Appendices

Includes:

- Screenshots
- Hash logs
- Raw data
- Tool outputs

📘 AUDITING & PLANNING AN AUDIT AGAINST AUDIT CRITERIA

Auditing is a systematic process of evaluating an organization's systems, policies, controls, and operations to determine whether they meet defined standards, regulations, and best practices.

In simple words:
➡️ *Auditing checks whether things are done correctly, safely, and according to rules.*

In cybersecurity/IT, auditing ensures that security controls are effective and compliant.

Purpose of Auditing

- Verify compliance with standards (ISO 27001, GDPR, PCI-DSS)
- Evaluate effectiveness of internal controls
- Detect weaknesses and vulnerabilities
- Ensure operational efficiency
- Identify risks
- Recommend improvements
- Provide assurance to management and stakeholders

Step-by-Step Process: Planning an Audit

1️⃣ Define Audit Objectives

Determine what the audit aims to achieve.

Examples:

- Ensure compliance with ISO 27001
- Verify access control implementation
- Check data protection practices

2️⃣ Identify and Establish Audit Criteria

Criteria may include:

- Organizational policies
- International standards (ISO, NIST)
- Regulatory requirements
- Contracts
- Internal procedures

3️⃣ Determine Audit Scope

Scope should define:

- Departments covered
- Geographic locations
- Systems and processes included
- Time period being audited

Example scope:
"Audit user access controls for HR and Finance departments for Q1 2024."

4️⃣ Develop Audit Plan

Audit plan includes:

- Audit schedule
- Activities and tasks
- Auditor responsibilities
- Checklist preparation
- Tools required

Example activities:

- Interview staff
- Review documents
- Conduct technical tests

5️⃣ Assign Audit Team

Includes:

- Lead auditor
- Technical experts
- Observers (optional)

6️⃣ Collect Pre-Audit Information

Gather background information like:

- Policies
- Network diagrams
- Previous audit reports
- Compliance documents

7️⃣ Develop Audit Checklist

Checklist ensures key topics are covered.

Example checklist items for ISO 27001:

- Are access controls documented?
- Are backups tested regularly?
- Are logs reviewed?

8️⃣ Communicate Audit Plan

Inform stakeholders:

- Purpose
- Schedule
- Scope
- Requirements
- Expectations

9️⃣ Conduct Risk Assessment (Optional but recommended)

Identify:

- High-risk areas
- Critical systems
- Sensitive data

Helps auditors prioritize focus areas.

1️⃣0️⃣ Logistics and Resource Planning

Ensure:

- Access to systems
- Interview availability
- Workspace for auditors
- Tools and permissions

📘 INFORMATION SECURITY MANAGEMENT SYSTEM (ISMS) MANAGEMENT

An **Information Security Management System (ISMS)** is a structured framework consisting of **policies, procedures, processes, and controls** used to protect an organization's information assets.

In simple words:
➡️ *ISMS is the entire security system of an organization that ensures information remains confidential, accurate, and available.*

ISMS Management refers to how an organization **creates, operates, monitors, maintains, and improves** its information security program—usually aligned with **ISO/IEC 27001**.

Objectives of ISMS Management

- Protect information confidentiality, integrity, and availability (CIA)
- Manage security risks systematically
- Ensure compliance with laws and regulations
- Prevent security breaches and cyberattacks
- Improve employee awareness and security culture
- Enable secure business operations
- Support continuous security improvement

Key Components of ISMS

1️⃣ Security Policies

Defines rules & guidelines for security.

2️⃣ Risk Management

Risk identification, analysis, evaluation, and treatment.

3️⃣ Asset Management

Classification and protection of information assets.

4 Access Control

Managing user permissions and authentication.

5 Physical and Environmental Security

Securing buildings, server rooms, and physical devices.

6 Communications and Operations Security

Ensures secure handling of data and IT operations.

7 Incident Management

Process for detecting, responding to, and reporting security incidents.

8 Business Continuity & Disaster Recovery

Ensuring operations continue during disasters.

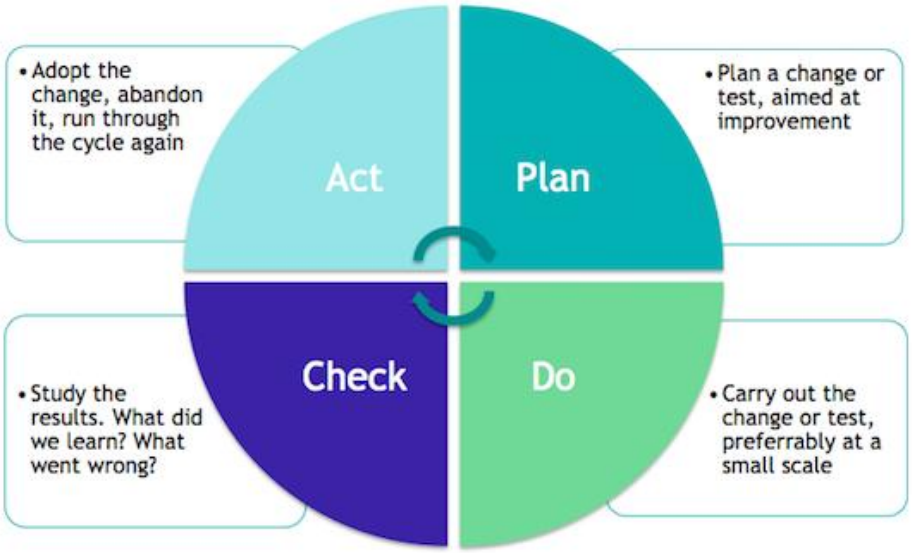
9 Compliance

Meeting legal, contractual, and regulatory requirements.

10 Training and Awareness

Educating employees to reduce human error.

ISMS Management Life Cycle (PDCA Model)



1 PLAN

- Define ISMS scope
- Identify assets
- Conduct risk assessment
- Create security policies
- Select security controls

2️⃣ DO

- Implement policies & controls
- Deploy technical security measures
- Train employees
- Document procedures

3️⃣ CHECK

- Monitor effectiveness
- Conduct internal audits
- Perform security reviews
- Analyze incidents

4️⃣ ACT

- Apply corrective actions
- Improve controls
- Update risk assessments
- Strengthen ISMS continuously

Processes Involved in ISMS Management

1. Risk Assessment and Risk Treatment

- Identify threats, vulnerabilities
- Evaluate risk level
- Select control measures (avoid, reduce, share, accept)

2. Policy Management

- Create policies (password, access control, incident response)
- Review regularly

3. Control Implementation

Deploy security controls such as:

- Firewalls
- Encryption
- Backup systems
- MFA
- Monitoring tools

4. Monitoring & Measurement

- Log analysis
- SIEM tools
- Compliance checks

5. Internal Audits

Ensure ISMS meets ISO 27001 and organizational goals.

6. Training and Awareness Programs

Reduce human error and social engineering risks.

📘 INTRODUCTION TO ISO 27001:2013

ISO/IEC 27001:2013 is an international standard for establishing, implementing, maintaining, and continually improving an **Information Security Management System (ISMS)**.

In simple words:
➡️ *ISO 27001:2013 provides a globally recognized method to keep an organization's information secure.*

It helps organizations manage information security risks using **policies**, **procedures**, and **technical controls**.

Purpose of ISO 27001:2013

- Protect information assets
- Manage cybersecurity risks systematically
- Provide a framework for ISMS
- Ensure confidentiality, integrity, and availability (CIA triad)
- Support legal and regulatory compliance
- Build customer trust
- Enable continuous improvement in security practices

Key Concepts in ISO 27001:2013

1️⃣ Information Security Management System (ISMS)

A structured framework to protect information using risk-based controls.

2️⃣ Risk Management

ISO 27001 requires organizations to identify, assess, and treat security risks.

3️⃣ Continual Improvement

Based on the **PDCA cycle (Plan-Do-Check-Act)**.

4️⃣ Annex A Controls

114 controls grouped into 14 domains (in 2013 version).

5️⃣ Statement of Applicability (SoA)

A mandatory document listing which Annex A controls are applicable and why.

📘 SECURITY POLICY DATABASE (SPD)

Security Policy Database (SPD) is a key component of **IPsec (Internet Protocol Security)** architecture. It determines *how incoming and outgoing IP packets should be treated* from a security standpoint.

In simple words:
➡️ *SPD decides which IP packets should be protected (encrypted/authenticated), which can pass normally, and which should be dropped.*

SPD works together with **SAD (Security Association Database)** to secure network traffic.

What is Security Policy Database (SPD)?

The **SPD** is a table containing a set of **rules** (security policies) that define:

- Which traffic requires IPsec protection
- Whether to encrypt, authenticate, or bypass
- How to apply appropriate security associations

Each policy includes:

- Source & destination IP
- Protocol (TCP/UDP/ICMP)
- Source & destination ports
- Action to apply (PROTECT/BYPASS/DISCARD)

These are called **selectors**.

SPD classifies traffic into one of three actions

1️⃣ PROTECT

Traffic **must** be processed by IPsec (encrypted, authenticated, or both).

Example:
Encrypt all traffic between two branch offices.

2️⃣ BYPASS

Traffic is allowed without IPsec protection.

Example:
Internal LAN traffic that doesn't require encryption.

3️⃣ DISCARD

Traffic is blocked.

Example:
Drop packets coming from blacklisted IP addresses.

IP traffic processing is the handling of data packets as they travel across a network, involving steps like encapsulating data, routing it to the correct destination, and assembling it at the receiver. On an outbound journey, packets are created, addressed, and sent; on an inbound journey, packets are received, checked, and reassembled into the original data. This process ensures data is transmitted reliably and arrives correctly at its intended destination.

A handshake protocol is a process of negotiation between two participants to establish a secure communication channel. The most common example is the TLS/SSL handshake, which involves a series of messages to authenticate both parties, agree on cryptographic algorithms, and generate shared secret keys for encrypted data transfer. Other examples include the TCP three-way handshake, which establishes a reliable connection by synchronizing sequence numbers between a client and server.

Secure Socket Layer (SSL)

Secure Sockets Layer (SSL) is an Internet security protocol that encrypts data to ensure secure communication between devices over a network. Originally developed by Netscape in 1995, SSL provides privacy, authentication and data integrity for online communications. SSL is the predecessor of TLS (Transport Layer Security), which is now the standard protocol for secure communications on the Internet.

Working of SSL

SSL ensures secure communication through three main mechanisms:

1. Encryption: Data transmitted over the network is encrypted, preventing unauthorized parties from reading it. If intercepted, encrypted data appears as an unreadable jumble of characters.
2. Authentication: SSL uses a handshake process to authenticate both the client and server, ensuring each party is legitimate and not an imposter.
3. Data Integrity: SSL digitally signs transmitted data to detect any tampering, ensuring that the data received is exactly what was sent.

Importance of SSL

- Encrypting sensitive information such as login credentials, financial transactions and personal data.
- Authenticating web servers to prevent users from connecting to fraudulent websites.
- Ensuring data integrity so transmitted information cannot be modified during transit.

📘 PGP Cryptographic Functions: Authentication & Confidentiality

PGP (Pretty Good Privacy) is a widely used cryptographic system that provides:

- Confidentiality (Encryption)
- Authentication (Digital Signatures)
- Integrity
- Compression
- Key Management

It combines symmetric key cryptography, public-key cryptography, and hash functions to secure emails, files, and digital communication.

PGP Provides Two Main Cryptographic Functions

1. Authentication (Digital Signatures)
2. Confidentiality (Encryption)

⭐ A. PGP Authentication (via Digital Signatures)

Purpose of Authentication

Ensures:

- Message is really from the claimed sender
- Message was not tampered with
- Sender *cannot deny* sending the message (non-repudiation)

How PGP Authentication Works (Steps)

Step-by-Step Process

Step 1: Sender generates a hash of the message

- Hash function used (SHA-1, SHA-256, etc.)
- Produces message digest

Step 2: Sender encrypts the hash with their *private key*

This produces a digital signature

Step 3: Digital signature is attached to the message

It is sent to the receiver.

Verification Process at Receiver Side

Step 4: Receiver decrypts signature using sender's *public key*

This recovers the original message digest.

Step 5: Receiver computes hash of received message

Step 6: Compare both hashes

- If equal → Message is authenticated & unchanged
- If not → Message tampered or wrong sender

✓ Output of PGP Authentication

- Sender identity is verified
- Message integrity is ensured
- Provides non-repudiation

⭐ B. PGP Confidentiality (via Encryption)

1. Purpose of Confidentiality

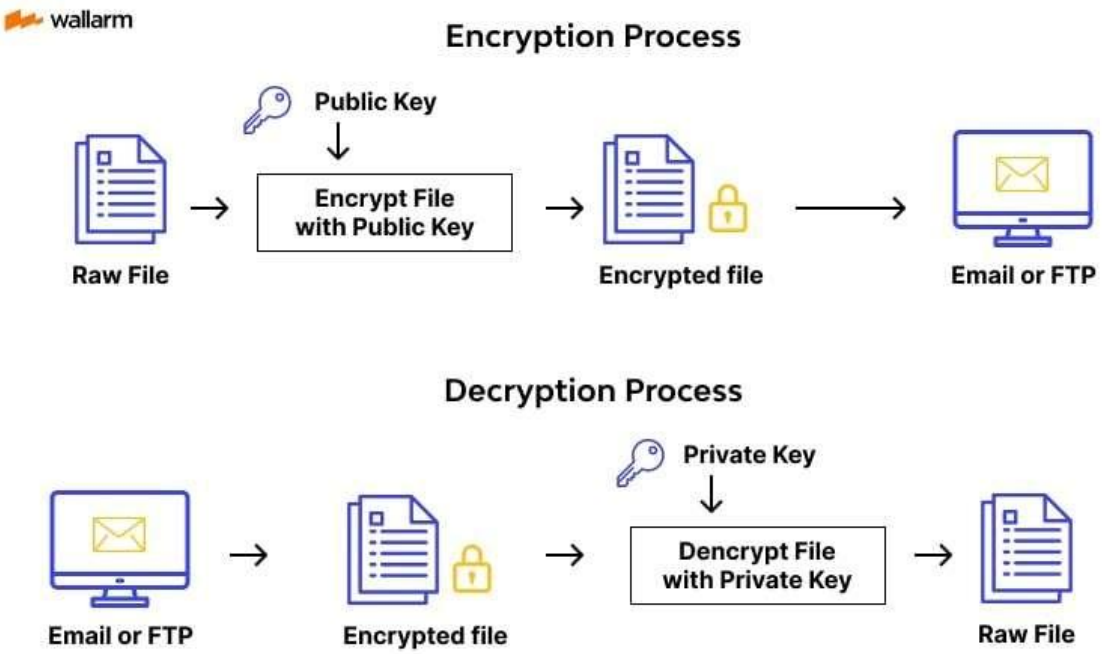
Ensures:

- Only intended receiver can read the message
- Prevents unauthorized access

PGP uses hybrid encryption:

- Symmetric key → Fast message encryption
- Public-key encryption → Secure key exchange

How PGP Confidentiality Works (Steps)



Step-by-Step Process

Step 1: Sender generates a random symmetric session key

Used for fast encryption (AES, IDEA, etc.)

Step 2: Message is encrypted with the session key

Produces ciphertext

Step 3: Session key is encrypted using receiver's *public key*

This is called public-key encapsulation

Step 4: Encrypted session key + encrypted message are sent

Decryption at the Receiver Side

Step 5: Receiver decrypts session key using their *private key*

Now receiver has the symmetric session key.

Step 6: Receiver decrypts the message using this session key

✓ Output of PGP Confidentiality

- Only the receiver can decrypt the key
- Message content remains secret
- Strong hybrid encryption is used

Key Differences Between Hashing & Encryption

Feature	Hashing	Encryption
Purpose	Integrity	Confidentiality
Direction	One-way	Two-way
Reversible?	❌ No	✓ Yes (with key)
Output	Fixed-length hash	Variable-length ciphertext
Key Used?	❌ No	✓ Yes (symmetric/asymmetric)
When data changes?	Completely new hash	Needs re-encryption
Examples	SHA-256, SHA-1, MD5	AES, DES, RSA
Main Use Cases	Passwords, integrity checks	Secure communication, data protection
Collision?	Possible (rare)	Not applicable
Digital Signature role	Creates the message digest	Encrypts digest for signature

Key Difference Between Symmetric & Asymmetric Encryption

Feature	Symmetric Encryption	Asymmetric Encryption
Number of Keys	One key (same key for encrypt/decrypt)	Two keys (public + private)
Speed	Fast	Slow
Security Level	Secure but depends on key secrecy	Very secure due to key pairs
Key Distribution	Difficult (must share key secretly)	Easy (public key can be shared openly)
Use Cases	Bulk data encryption	Key exchange, digital signatures
Algorithms	AES, DES, Blowfish	RSA, ECC, Diffie–Hellman
Confidentiality	Strong	Strong
Authentication	Not built-in	Provides authentication (via signatures)

Feature	Symmetric Encryption	Asymmetric Encryption
Computational Cost	Low	High
Scalability	Poor (needs many keys for many users)	Better scalability

What Is a Buffer Overflow?

A **buffer overflow** occurs when a program writes more data into a buffer (memory region) than it can hold.
This extra data **overwrites adjacent memory**, causing:

- Program crash
- Altered execution flow
- Injection of malicious code

In simple words:
➡️ *Buffer overflow means writing beyond the allocated memory space, which an attacker can exploit to take control of a system.*

Why Buffer Overflow Happens

- Using unsafe functions (e.g., gets(), strcpy(), sprintf())
- Lack of bounds checking
- Poor memory management
- Programmer assumptions about input size
- Low-level languages (C, C++) give direct memory access

How Buffer Overflow Attacks Work

Typical attack flow:

1. **Find vulnerable input field** (e.g., login field, user input).
2. **Craft malicious payload** with:
    - Dummy data
    - Return address overwrite
    - Shellcode / malicious instructions
3. **Send oversized input**
4. Program overwrites stack memory.

5. Control jumps to attacker's code.

What is Mitigation? (General Cybersecurity Definition)

Mitigation refers to the methods, controls, and strategies used to **reduce the impact, likelihood, or severity** of security threats, vulnerabilities, and attacks.

In simple words:
➡️ *Mitigation means applying measures that limit damage and reduce risk from cyber threats.*

Mitigation does **not** eliminate risk completely but minimizes it to an acceptable level.

Goals of Mitigation

- Prevent exploitation of vulnerabilities
- Reduce attack surface
- Limit impact if an attack occurs
- Increase security resilience
- Protect confidentiality, integrity, and availability (CIA)

What Is a Firewall?

A **firewall** is a network security device/software that monitors and controls incoming and outgoing network traffic based on predefined security rules.

In simple words:
➡️ *A firewall acts as a security guard between a trusted internal network and an untrusted external network (like the Internet).*

Purpose of a Firewall

- Block unauthorized access
- Allow legitimate traffic
- Protect internal systems
- Enforce security policies
- Prevent malware & intrusion attempts
- Monitor network activity

Types of Firewalls

1️⃣ Packet Filtering Firewall (Network Layer Firewall)

Evaluates packets based on IP addresses, ports, protocol, and flags.

2️⃣ Stateful Inspection Firewall

Tracks connection states (SYN, ACK, FIN). More secure.

3️⃣ Application Layer Firewall (Proxy Firewall)

Understands protocols like HTTP, FTP, DNS; inspects application content.

4️⃣ Next-Generation Firewall (NGFW)

Includes IDS/IPS, deep packet inspection, malware filtering, SSL inspection.

5️⃣ Hardware vs Software Firewall

- Hardware firewalls → Routers, appliances
- Software firewalls → OS-based (Windows Firewall), endpoint firewalls

Packet Filtering Firewall

Packet filtering is a firewall technique that checks each incoming/outgoing packet individually based on:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Protocol (TCP/UDP/ICMP)
- TCP flags (SYN, ACK, etc.)

Characteristics of Packet Filtering

- Fast and efficient
- Operates at Network Layer (Layer 3) and Transport Layer (Layer 4)
- Does *not* inspect packet contents (payload)
- Stateless (does not track connection state)

Advantages of Packet Filtering

- High performance
- Low overhead
- Easy to implement
- Good for basic security

Disadvantages of Packet Filtering

- No user authentication
- Cannot prevent IP spoofing
- Doesn’t track connections (stateless)
- Cannot inspect application-level data
- Limited security compared to modern firewalls

Working of a Firewall (General Working)

Step 1: Traffic Arrives at Firewall

Incoming or outgoing packets hit the firewall interface.

Step 2: Firewall Examines Packet Header

Checks:

- Source/Destination IP
- Source/Destination port
- Protocol
- Flags
- Interface

Step 3: Compare With Rule Set

Firewall checks rules in order.

Step 4: Apply Action

Possible actions:

- **ALLOW**
- **DENY/DROP**
- **REJECT** (send error message)
- **LOG** (record event)

Step 5: Connection Tracking (Stateful Firewalls)

Stateful firewalls maintain a **state table** for:

- Established connections
- Related connections
- New connections

This allows:

- More intelligent filtering
- Blocking unsolicited traffic

Step 6: Forward or Block Traffic

Once rules are applied:

- Allowed packets move to next hop
- Blocked packets are dropped

What Are HTTP Status Codes?

HTTP status codes are **3-digit responses** sent by a web server to a client (browser or application) to indicate the **result of the request**.

In simple words:
➡️ *They tell you whether the request was successful, redirected, resulted in an error, or failed.*

Why HTTP Status Codes Are Useful

- Helps debugging web applications
- Improves SEO (correct use of redirects)
- Helps browsers understand server responses
- Critical for API communication
- Helps understand server/client failures

Matrices(Matrix mining) are used to represent datasets, where each row corresponds to an observation or sample, and each column represents a feature or attribute of that sample. This structured representation makes it convenient to apply mathematical operations and transformations to the data

Introduction to Cyber Laws

Cyber Laws refer to the legal framework that governs the use of computers, digital devices, the internet, networks, and online communication. These laws are designed to protect individuals, organizations, and governments from cybercrimes, data breaches, misuse of digital resources, and unauthorized access.

They ensure that online activities are carried out **safely, ethically, and legally**.

Why Cyber Laws Are Needed

1. **Increase in Cybercrimes**
   Hacking, phishing, identity theft, ransomware, and online fraud have grown rapidly.
2. **Protection of Confidential Data**
   Personal data, financial information, and corporate secrets need legal protection.
3. **Regulation of Online Transactions**
   E-commerce, online banking, and digital payments require legal safety standards.
4. **Preserving Digital Rights**
   Copyright, trademarks, and intellectual property must be protected in the digital space.
5. **Ensuring National Security**
   Cyber laws help prevent cyberterrorism, espionage, and attacks on critical infrastructure.

Key Areas Covered Under Cyber Laws

1. Cybercrimes

Laws against offenses like:

- Hacking
- Phishing
- Cyberstalking
- Online harassment
- Virus/worm attacks
- Data theft
- Financial fraud
- Child pornography

2. Intellectual Property Rights (IPR)

Covers:

- Copyright protection
- Software piracy
- Digital signature misuse
- Trademark/domain name disputes

3. Data Protection & Privacy

Regulates:

- Storage and handling of personal data
- User consent
- Data breach reporting
- Sensitive personal data (biometrics, health data, financial data)

4. Electronic Commerce & Transactions

Includes:

- Legal recognition for electronic records and signatures
- Regulations for online contracts
- Authenticity and integrity of digital documents

5. Cybersecurity Standards

Ensures:

- Secure network practices
- Protection of critical information infrastructure
- Compliance with international security norms

E-Commerce

E-Commerce (Electronic Commerce) refers to the buying and selling of goods and services over the internet. It also includes online payments, online marketing, digital supply chains, and customer support.

Features of E-Commerce

- **Global reach** – customers can buy from anywhere
- **24×7 availability**
- **Electronic payment systems** (UPI, cards, wallets)
- **Lower transaction cost**
- **Faster delivery of services**

- **Automation of processes**

Advantages of E-Commerce

- Wide market access
- Cost efficiency
- Easy comparison of products
- Quick and secure payment
- Personalized recommendations

Limitations of E-Commerce

- Cybersecurity risks
- Lack of personal touch
- Delivery delays
- Internet dependency

E-Governance

E-Governance (Electronic Governance) refers to the use of information and communication technologies (ICT) by the government to deliver services, share information, improve efficiency, and ensure transparency.

Objectives of E-Governance

- Provide **efficient and fast services** to citizens
- Promote **transparency and accountability**
- Reduce corruption
- Improve government–citizen interaction
- Reduce paperwork
- Enable online delivery of government services

Pillars of E-Governance

- **Connectivity** (internet infrastructure)
- **Content** (accurate information)
- **Capacity** (trained workforce)
- **Capital** (investment in IT)
- **Citizen Interface** (user-friendly portals)

Advantages of E-Governance

- Faster service delivery
- Reduced corruption
- Better transparency and accountability

- Convenience (no need to visit offices)
- Digital record-keeping

Challenges of E-Governance

- Low digital literacy
- Poor internet access in rural areas
- Data privacy & security issues
- High initial cost
- Resistance to technological change

Certifying Authority (CA)

A **Certifying Authority** is an organization licensed by the **Government of India** to issue **Digital Signature Certificates (DSCs)** to individuals, companies, and organizations.

These certificates verify the identity of the person performing digital transactions.

Functions of a Certifying Authority

1. Issue Digital Signature Certificates (DSC)

- Issues Class 2, Class 3, and now **eKYC-based** certificates
- Used for filing income tax, GST, e-tenders, MCA, banking, etc.

2. Verify Applicant Identity

- Uses Aadhaar, PAN, documents, video verification.

3. Maintain a Secure Database

- Securely stores certificate records and revocation lists.

4. Publish Certificate Revocation List (CRL)

- A list of digital certificates that have been cancelled.

5. Follow PKI Standards

They operate within the **Public Key Infrastructure (PKI)** framework.

Examples of Certifying Authorities in India

- **NIC** (National Informatics Centre)
- **nCode Solutions (GNFC)**
- **e-Mudhra**
- **TCS-CA**
- **IDRBT**

Legal Provisions

Under the **IT Act, 2000:**

- **Section 24:** Appointment of CAs
- **Section 25:** Application for license
- **Section 30–33:** Duties, responsibilities & validity

Controller of Certifying Authorities (CCA)

The **Controller of Certifying Authorities (CCA)** is the **apex authority** under the IT Act, 2000, responsible for regulating and supervising all Certifying Authorities in India.

CCA operates under the **Ministry of Electronics & IT (MeitY).**

Functions of the CCA

1. License and Regulate CAs

- Grants licenses for new Certifying Authorities
- Suspends or revokes licenses if rules are violated

2. Establish National Public Key Infrastructure (PKI)

Ensures secure digital transactions across India.

3. Create and Maintain the Root Certificate

The CCA generates India's **Root Digital Certificate**, which all other certificates trust.

4. Monitor Compliance

Ensures that CAs follow:

- IT Act rules
- Security guidelines
- Operational procedures

5. Conduct Audits

Performs regular technical and security audits of all CA operations.

6. Maintain Repository of Digital Certificates

Maintains:

- Licensed CA list
- Certificate revocation lists
- Root certificate repository

7. Set Digital Signature Standards

Defines encryption standards, cryptographic algorithms, and key lengths.

Legal Provisions

Under the **IT Act, 2000:**

- **Section 17:** Appointment of Controller
- **Section 18:** Functions of Controller
- **Section 29:** Access to CA stored information

Difference Between CA and CCA

Basis	Certifying Authority (CA)	Controller of Certifying Authorities (CCA)
Role	Issues digital certificates	Supervises & regulates all CAs
Level	Operational level	Apex regulatory authority
Appointed by	Licensed by CCA	Appointed by Central Government
Function	Identity verification, issue DSC	Maintain Root PKI, audits, licensing
Examples	e-Mudhra, NIC	Only one – CCA (India Root CA)

Patent Law

A **patent** is an exclusive legal right granted by the government to an inventor for a new invention.
It allows the inventor to **make, use, sell, or license** the invention for a limited period.

In simple terms:
A patent protects an invention from being copied, used, or sold by others without permission.

Key Features of Patent Law

- Grants **exclusive rights** to the inventor
- Protects **new inventions**
- Encourages research and innovation
- Patent is territorial (valid only in granting country)
- Valid for a **limited period (20 years)**

What Can Be Patented?

An invention must satisfy 3 conditions:

1. Novelty

It must be new; no previous publication or use.

2. Inventive Step

It must not be obvious to a skilled person.

3. Industrial Applicability

It must be usable in industry or practical life.

Examples:

- Machines, devices
- Pharmaceuticals
- Software-hardware integrated inventions
- Manufacturing processes

What Cannot Be Patented (India)

According to **Indian Patents Act, 1970**, the following are *not patentable*:

- Mathematical or business methods
- Computer programs **as such** (i.e., without hardware invention)
- Algorithms
- Discoveries of scientific principles
- Plants, animals, seeds
- Traditional knowledge
- Surgical or medical procedures
- Natural substances (unless modified)

Patent Law in India

Indian Patents Act, 1970

Primary legislation governing patents in India.

Important Amendments:

- **1999, 2002, 2005**
- 2005 amendment allowed **product patents** (especially in pharma).

Validity:

- **20 years** from filing date (for all fields).

Governing Authority:

- **Controller General of Patents, Designs & Trademarks (CGPDTM)**

Copyright Law

Copyright is a legal right that protects **original creative works** of authors, artists, musicians, programmers, and creators.

It prevents others from **copying, distributing, performing, or modifying** the work **without permission**.

In simple words:
Copyright protects creative expression.

Works Protected by Copyright

Copyright covers original works like:

1. Literary Works

- Books
- Articles
- Computer programs & software
- Databases

2. Artistic Works

- Drawings, paintings
- Photographs
- Architecture designs

3. Musical Works

- Music compositions
- Song lyrics

4. Dramatic Works

- Scripts, plays
- Screenplays

5. Cinematographic Films

- Movies, short films, video content

6. Sound Recordings

- Recorded songs, podcasts

What is NOT Protected?

- Ideas or concepts (only their expression)
- Procedures, methods, algorithms
- Titles, names, short phrases
- Government works (some are exempted)
- Facts, news, data that is publicly available

Copyright Law in India

The Copyright Act, 1957

Primary law governing copyright protection in India.

Administered by:

- **Copyright Office**
- **Copyright Board**
- Ministry of Education, Govt. of India

Duration of Copyright

- **Literary, musical, artistic works:**
  Life of author + 60 years
- **Films, sound recordings, anonymous works:**
  60 years from date of publication
- **Government works:**
  60 years from publication

Intellectual Property Rights (IPR)

Intellectual Property Rights (IPR) are legal rights that protect creations of the mind — such as inventions, artistic works, designs, symbols, brand names, and software.

In simple terms:
IPR protects creativity, innovation, and brand identity.

Objectives of IPR

- Encourage creativity and innovation
- Protect creators from unauthorized use
- Promote fair competition
- Support economic growth
- Enable creators to earn revenue from their work

Importance of IPR

- Prevents unauthorized copying
- Encourages R&D
- Boosts economic growth
- Helps startups and innovators secure investment
- Strengthens brand value
- Facilitates technology transfer

IPR in Digital & Cyber World

Very important for IT exams.

IPR protects:

- Software source code
- Databases
- Mobile apps
- Creative digital content
- Domain names
- Digital media
- Online logos and brands

Cybercrimes like piracy, plagiarism, and software theft are also covered.

IPR in Cyberspace refers to the protection of creative, digital, and technological works that exist *online* or are created using computers and the internet.
It covers software, digital content, databases, websites, domain names, and online creative works.

Why IPR is Important in Cyberspace

- Prevents software piracy
- Protects digital content from copying and misuse
- Secures creators' rights on the internet
- Prevents domain name disputes
- Ensures fair use of digital resources
- Supports e-commerce and online business integrity

Forms of IPR Relevant in Cyberspace

1. Copyright

Protects:

- Software & source code
- Websites & UI design
- Multimedia, images, videos
- Online articles/blogs
- Digital music

2. Patents

Cover:

- Software-hardware inventions
- Cryptographic algorithms (if embedded in hardware)
- Network protocols innovations
  *(Note: Pure software algorithms are **not patentable** in India)*

3. Trademarks

Protect:

- Domain names (example: amazon.in)
- Logos, brand names of online businesses
- App icons

4. Trade Secrets

Includes:

- Source code
- Encryption keys
- Algorithms
- Data analytics models
- Business logic of online platforms

5. Domain Name Protection

Handled through:

- **ICANN**
- **WIPO Arbitration** for disputes
  Example: cybersquatting cases.

Cyber Threats to IPR

- Software piracy
- Online plagiarism
- Unauthorized copying of digital content
- Illegal streaming
- Counterfeit websites
- Source code theft
- Data scraping without permission

Cyber Ethics

Cyber Ethics refers to the moral principles and acceptable behaviour governing the use of computers, digital devices, the internet, and online communication.

In simple terms:
Cyber ethics means using technology responsibly, safely, legally, and respectfully.

Why Cyber Ethics Are Important?

- Prevent cybercrimes and misuse of technology
- Create a safe online environment
- Protect privacy and data
- Promote responsible online behaviour
- Reduce cyberbullying, harassment, and fraud
- Maintain trust in digital systems

Key Principles of Cyber Ethics

****1. Respect Privacy**

- Do not read others' messages or emails
- Do not misuse personal information
- Ask permission before sharing someone’s data

****2. Be Honest**

- Do not spread fake news
- Do not hack or modify online content

- Do not impersonate others

**3. Follow Copyright & IPR Rules

- Avoid piracy
- Do not copy software, music, movies illegally
- Give credit to original creators

**4. Maintain Integrity

- Do not manipulate data
- Do not break into accounts or systems

**5. No Cyberbullying

- Avoid abusive comments
- Do not threaten, harass, or troll people online

Offences & Penalties under the Information Technology (IT) Act, 2000

✅ **1. Section 43 — Unauthorized Access / Damage to Computer**

Offences:

- Accessing system without permission
- Copying / downloading data
- Introducing virus
- Disrupting services

Penalty:
➡️ *Compensation up to ₹1 crore* (civil liability)

✅ **2. Section 66 — Computer-Related Offences**

Covers Section 43 offences done **dishonestly or fraudulently**.

Punishment:
➡️ *Up to 3 years imprisonment*
➡️ *Fine up to ₹5 lakh*

✅ **3. Section 66B — Receiving Stolen Computer Resource**

Knowingly receiving stolen computer, device or data.

Punishment:

➡️ *Up to 3 years jail*

➡️ *Fine up to ₹1 lakh*

✅ **4. Section 66C — Identity Theft**

Using someone's passwords, digital signatures, biometric data.

Punishment:

➡️ *Up to 3 years jail*

➡️ *Fine up to ₹1 lakh*

✅ **5. Section 66D — Cheating by Personation (Online Fraud)**

Phishing, fake calls, online scams, impersonation.

Punishment:

➡️ *Up to 3 years jail*

➡️ *Fine up to ₹1 lakh*

✅ **6. Section 66E — Violation of Privacy**

Capturing, publishing, or sharing private images without consent.

Punishment:

➡️ *Up to 3 years jail*

➡️ *Fine up to ₹2 lakh*

✅ **7. Section 67 — Publishing Obscene Content**

Posting or transmitting obscene material online.

Punishment:

- **1st offence:** 3 years + fine up to ₹5 lakh
- **2nd offence:** 5 years + fine up to ₹10 lakh

✅ **8. Section 67A — Sexually Explicit Content**

Pornographic and sexually explicit material.

Punishment:

➡️ *Up to 5 years*

➡️ *Fine up to ₹10 lakh*

IP Security (IPSec)

IPSec (Internet Protocol Security) is a suite of protocols used to secure communication over IP networks.

It works at the **Network Layer (Layer 3)** of the OSI model and protects data traveling between two devices or networks.

IPSec is mainly used in **VPNs, secured enterprise networks, and remote access**.

Objectives of IPSec (Very Important for Exams)

IPSec provides the following security services:

1. Confidentiality

Encrypts data so unauthorized users cannot read it.

2. Integrity

Ensures data is not altered during transmission.

3. Authentication

Verifies identity of sender/receiver.

4. Anti-Replay Protection

Stops attackers from capturing and re-sending packets.

5. Secure Key Exchange

Provides safe distribution of encryption keys.

Core Components of IPSec

IPSec consists of:

1. Authentication Header (AH)

Provides:

- Integrity
- Authentication
- Anti-replay

Does NOT provide encryption → no confidentiality.

Used when data privacy is not required.

2. Encapsulating Security Payload (ESP)

Provides:

- **Encryption (Confidentiality)**
- Integrity
- Authentication

- Anti-replay

ESP is used more often than AH because it provides encryption.

IPSec Modes (Very Important)

IPSec works in **two modes**:

1. Transport Mode

- Encrypts **only the data/payload** of the IP packet
- Original IP header remains unchanged
- Used in **host-to-host** direct communication
- Suitable for end systems (PC ↔ PC)

2. Tunnel Mode

- Encrypts **entire IP packet including header**
- Adds a new IP header
- Used in **gateway-to-gateway** communication (Router ↔ Router)
- Most common in **VPNs**

IPSec Process (Simple Steps)

1. Packet arrives → **SPD** checks whether IPSec is required
2. Appropriate **SA** selected from SAD
3. AH/ESP applied → integrity/encryption
4. Packet transmitted securely
5. Receiver validates SA, decrypts, verifies integrity
6. Forward data to application layer

Advantages of IPSec

- Strong network-layer security
- Transparent to applications (no app modification needed)
- Essential for VPNs
- Protects all IP traffic
- Supports both IPv4 & IPv6

Limitations

- Complex to configure

- High computational overhead (encryption/decryption)
- Slower performance for large traffic volumes

Major Components of IPSec Architecture

The IPSec architecture consists of **five key elements**:

1. IPSec Protocols (AH & ESP)

a) Authentication Header (AH)

Provides:

- Integrity
- Authentication
- Anti-replay
  No encryption → No confidentiality

b) Encapsulating Security Payload (ESP)

Provides:

- **Encryption (Confidentiality)**
- Integrity
- Authentication
- Anti-replay

ESP is widely used for VPNs.

2. Security Association (SA)

SA = A one-directional *logical connection* between two IPSec devices.

Each SA defines:

- Security protocol (AH/ESP)
- Encryption algorithm (AES, 3DES)
- Hash algorithm (SHA-256, MD5)
- Keys
- Lifetime
- Mode (Transport / Tunnel)

🔹 **Two SAs are required for bidirectional communication**
(one for each direction).

3. Security Association Database (SAD)

- Stores all active Security Associations
- Each entry contains:
    - SPI (Security Parameter Index)
    - Sequence number
    - Keys
    - Algorithms
    - Lifetime

Receiver uses SAD to identify which SA applies to an incoming packet.

4. Security Policy Database (SPD)

Stores **policies** that decide which packets require IPSec.

Each SPD entry specifies:

- Traffic to protect (IP addresses, ports, protocols)
- Whether to **discard**, **bypass**, or **apply IPSec**
- Whether to use AH or ESP
- Tunnel or transport mode

SPD → Checks outgoing/incoming packets → Selects SA (from SAD).

5. Key Management – IKE (Internet Key Exchange)

IKE is used to automatically negotiate:

- Security associations
- Cryptographic keys
- Authentication methods

IKE runs in **two phases**:

Phase 1

- Establishes a secure, authenticated channel
- Uses Diffie-Hellman key exchange

Phase 2

- Negotiates IPSec SAs
- Generates session keys for AH/ESP

Data Leakage (Data Loss / Data Breach)

Data Leakage refers to the **unauthorized transmission, sharing, or exposure of confidential data** to an external or internal party.

It may happen accidentally or intentionally through:

- Employees
- Hackers
- Weak security systems
- Software vulnerabilities

Data leaked = loss of privacy, money, trust, and security.

Types of Data Leakage

1. Accidental Data Leakage

- Sending sensitive data to the wrong email
- Misconfigured cloud storage
- Lost devices (laptop, USB)

2. Malicious Insider Leakage

- Employees stealing data
- Selling confidential documents
- Unauthorized copying of files

3. External Attack Data Leakage

- Hacking
- Phishing
- Malware, spyware
- Ransomware

4. Physical Data Leakage

- Stolen devices
- Unlocked systems
- Dumpster diving (recovering thrown documents)

Common Causes of Data Leakage

- Weak passwords

- Phishing attacks
- Unprotected Wi-Fi
- Misconfigured cloud servers (AWS S3 buckets, etc.)
- Lack of encryption
- Unpatched software vulnerabilities
- Human error
- Use of personal devices (BYOD)

Email Security Protocols (PGP, SMTP, S/MIME)

Email security protocols protect email content from threats such as interception, tampering, spoofing, and unauthorized access.

The three major protocols are **PGP**, **SMTP**, and **S/MIME**.

1. PGP (Pretty Good Privacy)

Meaning

PGP is an **email encryption program** that provides:

- **Confidentiality**
- **Authentication**
- **Integrity**
- **Non-repudiation**

It uses **hybrid cryptography**:

- **Symmetric key** → encrypts message
- **Asymmetric key** → encrypts symmetric key

Key Features of PGP

- Combines **RSA** (public key) & **AES/IDEA** (symmetric key)
- Uses **Digital Signatures**
- Uses **Web of Trust** (no CA required)
- Compresses email before encryption

PGP Process (Simple)

1. Encryption

1. Message → compressed
2. Generate a random symmetric key
3. Encrypt email with symmetric key
4. Encrypt symmetric key with receiver's public key
5. Send both to receiver

2. Decryption

1. Receiver decrypts symmetric key using private key
2. Decrypts the message

2. SMTP (Simple Mail Transfer Protocol)

Meaning

SMTP is the basic email **sending protocol** used on the Internet.
By default, SMTP is **not secure** — it sends data in **plain text**.

Security Issues:

- No encryption
- Susceptible to spoofing
- Man-in-the-middle attacks
- Replay attacks

Secure Extensions to SMTP

1. SMTPS (SMTP over SSL/TLS)

- Adds encryption to SMTP
- Uses port **465** (implicit TLS)

2. STARTTLS

- Upgrades a normal SMTP connection to **encrypted TLS**
- Uses port **587**

3. SPF, DKIM, DMARC

Prevent email spoofing:

- **SPF** → verifies mail server identity
- **DKIM** → digital signature for emails
- **DMARC** → email authenticity policy

3. S/MIME (Secure / Multipurpose Internet Mail Extensions)

Meaning

S/MIME is a protocol that provides **secure email** using **X.509 digital certificates** issued by **Certifying Authorities (CAs)**.

Key Features of S/MIME

Provides:

- **Encryption**
- **Digital signatures**
- **Authentication**
- **Integrity**
- Works with MIME email formats

Uses:

- **Public Key Infrastructure (PKI)**
- **RSA**, **AES**, **SHA-256**

S/MIME Process

1. Sending Encrypted Mail

1. Sender fetches receiver's public certificate
2. Encrypts the email
3. Signs the email using sender's private key

2. Receiving

1. Receiver decrypts using private key
2. Verifies sender's digital signature