CIA (Confidentiality, Integrity, and Availability)

The **CIA Triad** represents the **three fundamental principles of cyber security**:

✓ Confidentiality

Meaning:
Ensures that **only authorized users** can access sensitive data.

Why it matters:
Prevents data leaks, privacy violations, and unauthorized access.

Examples:

- OTP & passwords in **banking apps**
- End-to-end encryption in **WhatsApp**
- Role-based access in companies (employee vs admin)

Threats to Confidentiality:

- Phishing attacks
- Data breaches
- Eavesdropping

Security Controls:

- Authentication (passwords, biometrics)
- Encryption
- Access control

✓ Integrity

Meaning:
Ensures that **data is accurate, complete, and not altered** without permission.

Why it matters:
Incorrect or tampered data can cause serious damage.

Examples:

- Marks in a **college database**
- Transaction amount in **online payments**
- Medical records

Threats to Integrity:

- SQL Injection
- Malware
- Unauthorized modifications

Security Controls:

- Hashing
- Digital signatures
- Input validation
- Access permissions

✓ **Availability**

Meaning:
Ensures that **systems and data are accessible when needed**.

Why it matters:
Downtime means business loss and user frustration.

Examples:

- Banking apps during salary day
- E-commerce sites during sales
- Cloud services

Threats to Availability:

- DDoS attacks
- Server crashes
- Ransomware

Security Controls:

- Backups
- Load balancing
- Redundant servers
- DDoS protection

Different Types of Cyber Attacks

A **cyber attack** is an attempt by an attacker to **steal data, damage systems, or gain unauthorized access**.

- ✓ **Phishing Attack**
  What it is:
  Tricking users into revealing sensitive information using fake emails, messages, or websites.
  Example:
  Fake email pretending to be from a bank asking for OTP or password.

- ✓ **Malware Attack**
  What it is:
  Malicious software designed to harm systems.
  Example:
  Downloading a cracked app that installs spyware.

- ✓ **Ransomware Attack**
  What it is:
  Encrypts user data and demands payment to restore access.
  Example:
  Your laptop files get locked and attacker asks for Bitcoin.

- ✓ **Denial of Service (DoS / DDoS)**
  What it is:
  Overloading a server with too many requests to make it unavailable.
  Example:
  Gaming or shopping website goes down during an attack.

- ✓ **Man-in-the-Middle (MITM) Attack**
  What it is:
  Attacker secretly intercepts communication between two parties.
  Example:
  Using public Wi-Fi where attacker steals login credentials.

What is an Attack Surface?

An **attack surface** is the **total number of points** in a system where an attacker can **enter, interact, or extract data**.

- ✓ **Digital Attack Surface**
  These are **software-based entry points**.
  Example:
  A banking website login page → possible attack surface for brute force or SQL injection.

- ✓ **Physical Attack Surface**
  These involve **physical access to systems or devices**.
  Example:
  Attacker inserts an infected pendrive into an office computer.

- ✓ **Human Attack Surface**
  This involves **people**, which is often the weakest link.
  Example:
  Employee clicks a fake email link and enters login credentials.

OWASP Top 10 Vulnerabilities (Latest Conceptual List)

- ✓ **Broken Access Control**
  Users can access data or functions they should not.
  Example: Normal user accessing admin panel
- ✓ **Cryptographic Failures**
  Sensitive data is not properly encrypted.
  Example: Storing passwords in plain text
- ✓ **Injection**
  Malicious input is sent to the system.
  Example: SQL Injection, Command Injection
- ✓ **Insecure Design**
  Security is not considered during application design.
  Example: No rate limiting on login attempts
- ✓ **Security Misconfiguration**
  Incorrect security settings.
  Example: Default passwords, open cloud storage
- ✓ **Vulnerable & Outdated Components**
  Using old libraries with known vulnerabilities.
  Example: Outdated frameworks
- ✓ **Identification & Authentication Failures**
  Weak login and session management.
  Example: No MFA, weak passwords
- ✓ **Software & Data Integrity Failures**
  Untrusted software updates or data.
  Example: No integrity check on updates

✓ **Security Logging & Monitoring Failures**
Attacks are not detected or logged.
Example: No alert for multiple failed logins

✓ **Server-Side Request Forgery (SSRF)**
Server is tricked into accessing internal resources.
Example: Server fetching attacker-controlled URLs

✓ **Why OWASP Top 10 is Important?**

Industry Standard

- Used globally by developers and security teams

Prevents Major Attacks

- Covers the most common real-world vulnerabilities

Helps Secure Applications

- Acts as a **security checklist**

Improves Developer Awareness

- Educates developers about common mistakes

Required in Interviews & Audits

- Frequently asked in **interviews**
- Referenced in **security audits**

Mapping App to possible attack surfaces

Example 1: Email Application (Gmail / Outlook)

Attack Surfaces:

- **Login page** → brute force, credential stuffing
- **Email attachments** → malware, ransomware
- **Links in emails** → phishing
- **SMTP/IMAP protocols** → misconfiguration
- **Cloud storage** → data breach

Mapping:
User input → Web app → Server → Mail database

Example 2: WhatsApp / Messaging App

Attack Surfaces:

- **Account authentication** → OTP hijacking
- **Media files** → malicious files
- **APIs** → broken authentication
- **End-to-end encryption keys** → key leakage

- **Backup storage** → insecure cloud backups

Mapping:
User → Mobile App → API → Server → Database

Example 3: Banking / Payment App

Attack Surfaces:

- **Login & OTP** → phishing, MITM
- **Transaction APIs** → replay attacks
- **Session management** → hijacking
- **Mobile device** → malware
- **Database** → financial data theft

Mapping:
User → App → Secure API → Banking server → DB

Data Flow in an Application

User → Application → Server → Database → Server → Application → User

✓ **User → Application**

(**Input Stage**)

What happens:

- User enters data (login, message, payment, form)
- Example: username & password in a login form

Possible Attacks:

- **Phishing** (fake login pages)
- **Keylogging malware**
- **Brute force attacks**
- **Social engineering**

Reason: User is the weakest link.

Application (Frontend Layer)

(**Web / Mobile App**)

What happens:

- App collects input
- Sends request to server via HTTP/HTTPS

  Possible Attacks:

- **Cross-Site Scripting (XSS)**
- **Client-side validation bypass**
- **Insecure storage (cookies, local storage)**
- **Session hijacking**

Reason: Frontend code is visible to attackers.

✓ **Application → Server**

(**Network Layer**)

What happens:

- Data travels over the internet
- Usually via APIs

Possible Attacks:

- **Man-in-the-Middle (MITM)**
- **Packet sniffing**
- **Replay attacks**
- **API abuse**

Reason: Insecure network or no encryption.

✓ **Server (Backend Layer)**

What happens:

- Server processes requests
- Applies business logic
- Communicates with database

Possible Attacks:

- **SQL Injection**
- **Command Injection**
- **Broken access control**
- **Authentication bypass**
- **Remote Code Execution (RCE)**

Reason: Poor input validation or misconfiguration.

✓ **Server → Database**

(**Data Storage Layer**)

What happens:

- Server queries database
- Stores or retrieves data

Possible Attacks:

- **Database injection**
- **Unauthorized queries**
- **Data tampering**
- **Privilege escalation**

Reason: Weak database permissions.

✓ **Database**

(**Data at Rest**)

What happens:

- Sensitive data stored
- Example: passwords, personal info

Possible Attacks:

- **Data breach**
- **Insider attacks**
- **Unencrypted data exposure**

Reason: No encryption or poor access control.

✓ **Return Flow (Database → Server → App → User)**

Possible Attacks:

- **Data leakage**
- **Information disclosure**
- **Response manipulation**