Cybersecurity Fundamentals

Cybersecurity and the CIA Triad

1 Introduction

Cybersecurity is the practice of defending digital systems, networks, and data from malicious attacks and unauthorized access. A foundational model in this field is the CIA triad, which guides security policies based on three core principles: confidentiality, integrity, and availability.

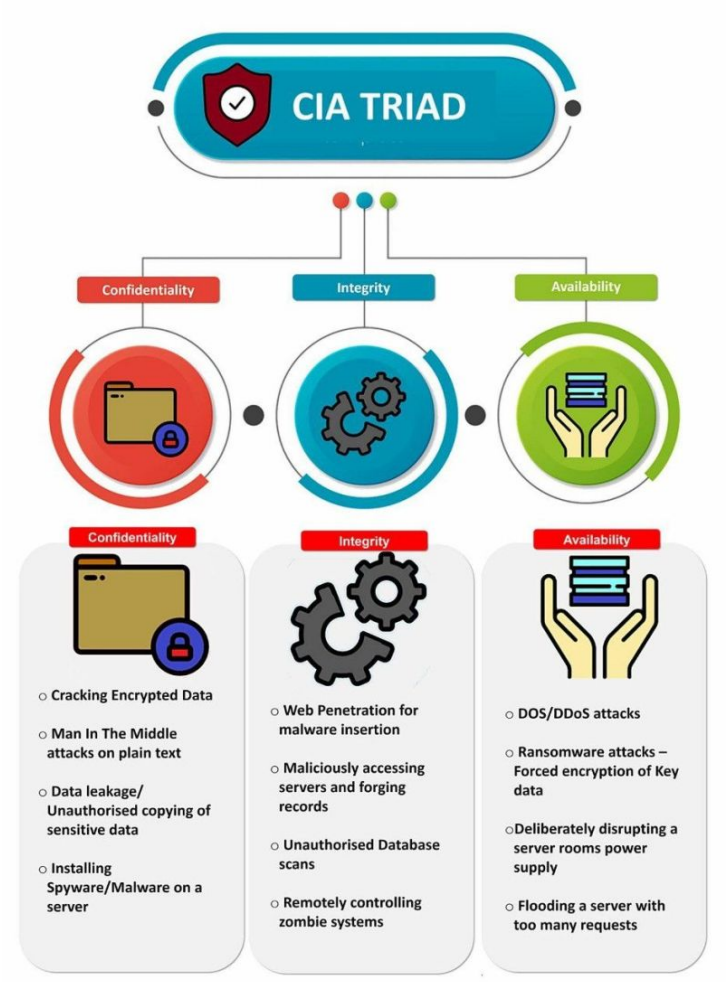

Figure 1: Screenshot illustrating the concept and application of the CIA Triad

2 The CIA Triad Explained

2.1 Confidentiality

Confidentiality ensures that sensitive information is accessible only to authorized individuals. It focuses on preventing unauthorized disclosure of data through mechanisms such as encryption, authentication, and access controls.

2.2 Integrity

Integrity guarantees that data is accurate, authentic, and has not been modified or tampered with by unauthorized parties. It ensures trustworthiness of information using validation checks, hashing, and audit logs.

2.3 Availability

Availability ensures that authorized users can reliably access information and systems whenever needed. This principle protects against service disruptions, system failures, and data loss.

3 Real-World Examples

3.1 Banking Systems

The banking sector heavily relies on the CIA triad to protect financial data and operations.

3.1.1 Confidentiality

When users log into online banking systems, strong authentication methods such as two-factor authentication ensure that only authorized individuals can access sensitive financial information. Data is encrypted during both transmission and storage.

3.1.2 Integrity

Banking systems implement robust controls to ensure that transactions and account balances remain accurate and cannot be altered by unauthorized users. Transaction logs and audit trails help detect and correct discrepancies.

3.1.3 Availability

Banks employ redundant servers, backup power supplies, and disaster recovery plans to ensure continuous availability of online banking services and ATMs, even during hardware failures or cyberattacks such as Distributed Denial of Service (DDoS) attacks.

3.2 Social Media Platforms

Social media platforms apply the CIA triad to protect user data and maintain service reliability.

3.2.1 Confidentiality

Encryption is used to protect personal messages, photos, and login credentials, ensuring access only for authorized users. Strict access controls limit employee access to sensitive user data.

3.2.2 Integrity

Platforms use anti-tampering mechanisms to prevent unauthorized modifications of user profiles or posts. These measures help prevent hackers from spreading false information or altering content under another user's identity.

3.2.3 Availability

Social media companies rely on load balancers and redundant data centers to manage high traffic volumes and maintain uninterrupted service availability, even during system failures or cyberattacks.

Types of Cyber Attackers

4 Introduction

Cyber attackers differ in skill level, motivation, and objectives. By analyzing insights from credible cybersecurity blogs and industry reports, attackers can be broadly classified into script kiddies, insiders, hacktivists, and nation-state actors.

5 Script Kiddies

Script kiddies are individuals with limited technical knowledge who rely on pre-written scripts and automated tools created by others.

Motivation: Curiosity, entertainment, or desire for recognition.

Characteristics:

- Low technical expertise
- Dependence on ready-made tools
- Target poorly secured systems

Common Attacks: Website defacement, simple DDoS attacks, and password brute-force attempts.

6 Insider Threats

Insiders are employees, contractors, or partners who misuse legitimate access to organizational systems.

Motivation: Financial gain, revenge, or negligence.

Characteristics:

- Authorized access to systems
- Difficult to detect
- Can be intentional or accidental

Common Attacks: Data theft, system sabotage, and unauthorized data leakage.

7 Hacktivists

Hacktivists conduct cyberattacks driven by political, social, or ideological beliefs.

Motivation: Protest, activism, or spreading awareness.

Characteristics:

- Public-facing attacks
- Target governments and corporations
- Seek media attention

Common Attacks: Website defacement, data leaks, and Distributed Denial of Service (DDoS) attacks.

8 Nation-State Actors

Nation-state actors are government-sponsored hackers with advanced technical skills and significant resources.

Motivation: Cyber espionage, military advantage, and national security.

Characteristics:

- Highly sophisticated tools
- Long-term stealthy operations (APT)
- Significant funding and infrastructure

Common Attacks: Cyber espionage, critical infrastructure attacks, and supply chain compromises.

9 Comparison of Attacker Types

Attacker Type	**Skill Level**	**Motivation**	**Target**
Script Kiddies	Low	Curiosity, Fun	Weak systems
Insiders	Medium–High	Revenge, Money	Internal systems
Hacktivists	Medium	Ideology	Governments, Firms
Nation-State Actors	Very High	Espionage, Warfare	Critical infrastructure

Attack Surfaces and Their Vulnerabilities

10 Introduction

An attack surface encompasses all potential entry points and vulnerabilities that attackers can exploit to gain unauthorized access to a system or network. Common attack surfaces include web applications, mobile apps, APIs, networks, and cloud infrastructure, each with specific vulnerabilities.

11 Web Applications

Web applications are often internet-facing, making them a primary target. Attackers typically exploit weaknesses in user input fields, authentication mechanisms, or application logic.

11.1 Common Vulnerabilities

- **SQL Injection (SQLi):** Injecting malicious SQL code into input fields to view, change, or delete data in the backend database.
- **Cross-Site Scripting (XSS):** Injecting malicious client-side scripts into web pages viewed by other users to steal session cookies or credentials.
- **Broken Authentication/Access Control:** Weak password policies, lack of multi-factor authentication (MFA), or flaws in user session management can allow attackers to gain unauthorized access.
- **Security Misconfigurations:** Using default credentials, exposing unnecessary debug information, or having overly permissive access controls can provide easy entry points.

12 Mobile Apps

Mobile applications face unique risks due to data storage on devices and communication with backend servers.

12.1 Common Vulnerabilities

- **Insecure Data Storage:** Storing sensitive data like login credentials or personal information on the device without proper encryption.

- **Insecure Communication:** Transmitting sensitive data over unencrypted channels, making it susceptible to interception (Man-in-the-Middle attacks).
- **Weak Authentication:** Easily guessable passwords or the absence of strong authentication protocols like MFA.
- **Improper Platform Usage:** Misusing platform-specific security controls or frameworks.

13 APIs (Application Programming Interfaces)

APIs are the "connective tissue" between different software components and cloud services, and their rapid deployment can lead to overlooked security flaws.

13.1 Common Vulnerabilities

- **Broken Object Level Authorization (BOLA):** An attacker accesses unauthorized resources by manipulating object references in API requests.
- **Lack of Rate Limiting:** Failure to restrict the number of requests a user can make in a given time, allowing for brute-force attacks or API abuse.
- **Excessive Data Exposure:** APIs returning more data than the client needs, potentially exposing sensitive information.
- **Broken User Authentication:** Flaws in authentication mechanisms that allow attackers to bypass access controls.

14 Networks

The network infrastructure itself presents an attack surface through its various connected devices and communication channels.

14.1 Common Vulnerabilities

- **Open Ports and Services:** Unnecessary ports left open can be scanned and exploited by attackers.
- **Outdated Software and Unpatched Systems:** Attackers actively search for and exploit known vulnerabilities in software and operating systems that have not been updated.

- **Weak Passwords and Authentication:** Using weak credentials or poor authentication protocols across network devices like routers and switches.
- **Misconfigured Firewalls:** Improperly configured firewalls can inadvertently allow malicious traffic into the network.

15 Cloud Infrastructure

The move to cloud platforms introduces new complexities and shared responsibilities for security, often leading to misconfigurations that expand the attack surface.

15.1 Common Vulnerabilities

- **Misconfigured Cloud Storage:** Publicly accessible storage buckets (like AWS S3) due to incorrect permission settings, exposing sensitive data.
- **Poor Access Management (IAM):** Overly permissive Identity and Access Management (IAM) roles or a lack of MFA can lead to credential theft and privilege escalation.
- **Lack of Visibility and Monitoring:** Difficulty in monitoring all cloud assets and activities, allowing breaches to go undetected for long periods.
- **Insecure APIs and Endpoints:** Unauthenticated or poorly managed cloud APIs can be a major entry point for attackers.

OWASP Top 10 (2025) Web Application Vulnerabilities

The **OWASP Top 10** lists the most critical web application security risks, which are dangerous because they are frequently exploited and can lead to severe consequences such as data breaches, unauthorized system access, and financial losses. Here's why each is dangerous:

- **A01:2025 - Broken Access Control:** Users access data or functions beyond their authorization (e.g., viewing other users' accounts), leading to unauthorized data disclosure or modification.
- **A02:2025 - Security Misconfiguration:** Default settings, exposed services, or incomplete setups (like open cloud storage) create easy entry points for attackers.

- **A03:2025 - Software Supply Chain Failures:** Compromised or outdated third-party components (libraries, frameworks) introduce vulnerabilities into your application.
- **A04:2025 - Cryptographic Failures:** Weak encryption or poor key management leaves sensitive data exposed, both in transit and at rest.
- **A05:2025 - Injection:** Untrusted data used as commands (SQL, OS, LDAP) can be manipulated to execute malicious code.
- **A06:2025 - Insecure Design:** Flaws in the fundamental architecture or design, often from lack of threat modeling, allow vulnerabilities to exist before coding.
- **A07:2025 - Authentication Failures:** Weak login, session management, or MFA bypasses allow attackers to impersonate legitimate users.
- **A08:2025 - Software or Data Integrity Failures:** Attackers can modify code or data (e.g., insecure updates, unsigned plugins) to achieve malicious goals.
- **A09:2025 - Security Logging & Alerting Failures:** Insufficient logging and alerting means breaches go undetected and attackers can operate longer.
- **A10:2025 - Mishandling of Exceptional Conditions:** Poor error handling reveals sensitive system details, while not managing unexpected states can cause crashes or security gaps.

Attack Surfaces and Threats in Digital Applications

16 Email Applications

The primary attack surface in email systems is the user and the email content itself.

Attack Surfaces:

- User interface
- Email content (links and attachments)
- User credentials
- Third-party integrations
- Mail servers

Possible Attacks and Threats:

- **Phishing and Spear Phishing:** Fraudulent emails used to steal sensitive information or distribute malware.
- **Malware Delivery:** Viruses, ransomware, and Trojans sent through attachments or links.
- **Business Email Compromise (BEC):** Impersonation of executives to trick employees into transferring money or sensitive data.
- **Account Takeover:** Unauthorized access to an email account to spy on data or conduct further attacks.

17 WhatsApp and Similar Messaging Applications

These applications have multiple attack surfaces despite using end-to-end encryption.

Attack Surfaces:

- Network communication (metadata)
- User input fields
- Local data storage
- Third-party libraries
- Device operating system

Possible Attacks and Threats:

- **Man-in-the-Middle (MitM) Attacks:** Interception of network traffic on unsecured Wi-Fi networks.
- **Malware via Links or Files:** Malicious links or files causing device compromise.
- **Social Engineering:** Tricking users into installing fake or modified applications.
- **Unauthorized Data Access:** Exploiting vulnerabilities to access cached or temporary data.

18 Mobile Banking Applications

Mobile banking apps handle highly sensitive data and have complex attack surfaces.

Attack Surfaces:

- APIs and backend services
- Client-side application code
- Local data storage and communication channels
- Authentication systems
- Third-party SDKs

Possible Attacks and Threats:

- **Banking Trojans and Malware**
- **Insecure APIs**
- **Weak Authentication and Session Management**
- **Reverse Engineering and App Tampering**
- **Phishing and Smishing**

Data Flow in a Typical Web Application

Data flow in a typical web application follows a logical sequence from the user's browser to the server and database, allowing for interactions and information retrieval.

18.1 Data Flow Sequence

Data Flow: User → Application → Server → Database

18.2 Stages of the Data Flow Process

User Interaction: A user initiates an action through the client-side application (e.g., filling out a form or clicking a link) on their device such as a web browser or mobile app. The application gathers the input and formats it into an HTTP request.

Application (Client-side): The user's application processes the input and sends the HTTP request across the internet to the server.

Server (Web/Application Server): The request arrives at the server, which validates and interprets it. The application logic running on the server determines what data is needed and how to process the request.

Database: The server communicates with the database using commands such as SQL queries to store, retrieve, update, or delete data as required by the user's request.

Reverse Flow (Response): The database sends the results back to the server. The server formats the response and sends it back to the client application, which displays the final result to the user.

Attack Surfaces in the Data Flow

Attacks can occur at virtually any stage of this process.

18.3 User / Client Side

Cross-Site Scripting (XSS): An attacker can inject malicious scripts into a website, which then execute in the victim's browser [**?**]. This allows attackers to steal sensitive information such as session cookies or login credentials.

Man-in-the-Browser (MitB): Malware running on the user's device can intercept or modify data before it is sent to the server.

18.4 During Transmission (User to Server)

Man-in-the-Middle (MitM) Attacks: If the connection is not encrypted (i.e., not using HTTPS), an attacker can intercept, read, or modify data while it travels across the network [**?**].

18.5 Server Side

SQL Injection (SQLi): An attacker manipulates user input to interfere with database queries [**?**]. This can allow unauthorized viewing, modification, or deletion of data, and even full database control.

Authentication/Authorization Bypass: Attackers exploit flaws in authentication systems to gain unauthorized access or elevate privileges [**?**].

Denial of Service (DoS/DDoS): Attackers overload the server with excessive requests, making it slow or unavailable to legitimate users [**?**].

Insecure Server Configuration: Misconfigured servers may expose sensitive files, services, or introduce additional vulnerabilities.

18.6 Database

Data Exfiltration: If attackers gain database access (often via SQL injection), they can steal complete datasets.

Privilege Escalation: Attackers with limited database access may exploit vulnerabilities to gain higher administrative privileges.

19 Security Mitigation Measures

Robust security measures such as input validation, parameterized queries, HTTPS encryption, and regular security audits are crucial for mitigating these risks. The OWASP Top Ten project provides extensive documentation on the most critical web application security risks and recommended countermeasures.

Summary of Web Application Data Flow and Security Risks

A typical web application operates by transferring data in a structured flow between the user, the application, the server, and the database. When a user interacts with a website, such as submitting a form or clicking a link, the input is collected by the client-side application and sent as an HTTP request to the server. The server processes this request using application logic and communicates with the database to retrieve or modify data. After processing, the server sends the response back to the user's browser, where the result is displayed.

Each stage of this data flow presents potential security risks. On the client side, attackers may exploit vulnerabilities such as Cross-Site Scripting (XSS), where malicious scripts execute in the user's browser, or Man-in-the-Browser attacks, where malware intercepts or alters data before it is transmitted. During data transmission, unsecured connections can lead to Man-in-the-Middle (MitM) attacks that allow attackers to intercept or manipulate sensitive information.

Server-side threats include SQL Injection attacks, authentication and authorization bypasses, denial-of-service (DoS/DDoS) attacks, and insecure server configurations. These vulnerabilities can compromise application functionality, expose sensitive data, or disrupt

services. If attackers gain access to the database, they may perform data exfiltration or privilege escalation, leading to severe data breaches.

To mitigate these risks, robust security measures must be implemented throughout the application lifecycle. These include strict input validation, use of parameterized queries, secure communication through HTTPS, strong authentication mechanisms, and regular security assessments. Resources such as the OWASP Top Ten provide valuable guidance for identifying and addressing the most critical web application security vulnerabilities.