# **Task 1 :** Understanding Cyber Security Basics & Attack Surface

## 1. Introduction: The CIA Triad

To understand cybersecurity, we must look at the three pillars that protect data. In the context of a Digital Banking Application, these are:

- Confidentiality: This ensures that sensitive information, like account balances and PII (Personally Identifiable Information), is only accessible to authorized users.
  - Example: Using strong encryption so a hacker cannot read my bank statements even if they intercept the data.
- Integrity: This ensures that data remains accurate and has not been altered during transit or storage.
  - Example: Preventing a "Man-in-the-Middle" from changing the recipient's account number during a wire transfer.
- Availability: This ensures that the banking service is reachable whenever the customer needs it.
  - Example: Protecting the bank's servers against DDoS attacks that would crash the app and prevent people from paying their bills.

## 2. Threat Actor Profiles

Understanding who is attacking helps us prioritize our defenses.

- Script Kiddies: Amateurs who use existing "off-the-shelf" hacking tools. They usually target banks for notoriety rather than sophisticated theft.
- Insiders: These are the "threats from within"—employees or contractors. They are dangerous because they already have legitimate access to the system.
- Hacktivists: Groups that attack for political or social reasons (e.g., protesting a bank's investment choices).
- Nation-State Actors: Highly funded, government-backed groups. They target financial infrastructure for espionage or to cause large-scale economic instability.

## 3. Attack Surface

The Attack Surface is the sum of all points where an unauthorized person can attempt to enter or extract data.

The Banking Data Flow:

Below is how data moves when a user interacts with a bank, along with the potential attack points at each stage:

1. User Interface (The Entry Point):
    - Flow: User enters credentials into the mobile app or web browser.
    - Attack Point: Malware/Keyloggers on the user's phone or Phishing sites that look like the real bank.
2. The Network (The Bridge):
    - Flow: Data travels via ISP or Public Wi-Fi to the bank's servers.
    - Attack Point: Eavesdropping or Man-in-the-Middle (MitM) attacks if the connection isn't properly encrypted with TLS.
3. Application Server (The Gatekeeper):
    - Flow: The server processes the login request and runs the "business logic."
    - Attack Point: Broken Access Control (allowing one user to see another's data) or Brute Force attacks on passwords.
4. The Database (The Vault):
    - Flow: The server fetches account details from the database.
    - Attack Point: SQL Injection (SQLi), where an attacker sends malicious code to trick the database into dumping all customer records.

# 4. Where Attacks Can Happen in This Flow

- Different attacks target different parts of that journey:

| Stage | Possible Attacks |
|---|---|
| User | Phishing, keylogging |
| Application | XSS, broken authentication |
| Network | Man-in-the-Middle (MITM), sniffing |
| Server | Remote Code Execution (RCE), privilege escalation |
| Database | SQL injection, data leakage |

# 5. Key Vulnerabilities (OWASP Top 10)

Based on my research of the OWASP Top 10, the following are most critical for a bank:

- A01: Broken Access Control: If a user can change a digit in their URL (e.g., bank.com/account/101 to 102) and see someone else's balance, the bank has failed this control.
- A03: Injection: Specifically SQL Injection. This is dangerous because it targets the database directly, potentially exposing every single customer's data at once.
- A07: Identification and Authentication Failures: This happens when a bank doesn't enforce Multi-Factor Authentication (MFA), making it easy for hackers to use stolen passwords.

# 6. Summary

Cybersecurity is not a "one-and-done" setup; it is a continuous process of reducing the Attack Surface. By understanding the data flow from the user's thumb to the bank's database, we can see that security must be applied at every layer.

Protecting a system requires thinking like an attacker to identify where the "walls" are thinnest. Whether it is sanitizing database inputs to prevent SQLi or educating users about phishing, every step taken to shrink the attack surface makes the overall system significantly more resilient.