DAY - 1: Understanding Cyber Security Basics & Attack Surface

Cybersecurity is basically the art and science of keeping digital stuff safe from bad actors. Think of it as locking your house, but for computers, phones, apps, networks, and all the data flying around. The big three rules everyone follows are called the CIA triad — Confidentiality, Integrity, and Availability.

CIA Triad:

Confidentiality — Keeping secrets secret. Only the people who are supposed to see the info get to see it. In banking apps: Your account balance, card numbers, and transaction history stay hidden from everyone else. Banks use strong encryption so even if someone snoops on your Wi-Fi, they can't read what's being sent. A famous real-world screw-up was the Capital One breach a few years back — hackers got away with 100 million people's personal and financial details because of a misconfigured cloud firewall. Suddenly strangers could see your credit limits and social security numbers. Nightmare fuel for identity theft. On social media like Instagram or WhatsApp: Private DMs, stories set to "close friends," or profile info shouldn't leak. Remember scandals where data got harvested for ads or worse? That's confidentiality going out the window, leading to stalking, scams, or doxxing.

Integrity — Making sure data doesn't get messed with without permission. It's about trust that what you see or send is exactly what was meant. Banking example: When you transfer ₹5000 to a friend, nobody changes it to ₹50,000 halfway through. Banks use digital signatures and checksums so the amount stays the same from your phone to their servers. If a hacker tampers with it (like in a man-in-the-middle attack), the system should catch it. Social media example: You post a photo or status — it shouldn't get secretly edited or faked by someone else. Deepfakes or manipulated posts during elections spread lies fast because integrity broke down. People lose trust in what they read online.

Availability — Making sure the service is there when you need it. No one should block you from using it. Banking: Imagine it's payday and you can't log in to pay bills or transfer money because of a massive DDoS attack flooding the servers. Or ransomware locks everything up — you're locked out completely. Social media: Platforms go down from attacks (or sometimes just overload), and suddenly you can't message friends, scroll feeds, or post updates. During big events, this can feel like the world stopped talking.

The triad isn't perfect — making things super secure (like forcing crazy-long passwords every time) can annoy users and hurt availability. Good security finds a smart balance.

About Attackers:

Cyber attackers aren't all the same. They have different skills, reasons, and tools. Depending upon that they can be classified into the following types:

- Script Kiddies — The beginners or casual troublemakers. They download ready-made hacking tools from forums, run them without really understanding, and attack easy targets

like websites with old software. Motivation? Bragging on Discord, boredom, or minor chaos. They can still cause real damage, like defacing a small business site.

- Insiders — People already inside the organization — employees, ex-employees, contractors. They have logins and know where the juicy data is. Some do it for revenge (leaking company secrets after being fired), others for money (selling customer data), or just by accident (falling for phishing). Super dangerous because they skip the "break in" part.
- Hacktivists — Groups or solo people driven by a cause. Think Anonymous or similar crews. They DDoS government sites, leak embarrassing docs, or deface pages to protest politics, environment, or corporate stuff. It's activism with code.
- Ethical Hackers — Government-sponsored pros (from countries like the US, China, Russia, etc.). They have huge budgets, custom malware, zero-day exploits nobody else knows about. Goals: Spy on companies/governments, steal tech secrets, disrupt power grids, or influence elections. These are the advanced persistent threats (APTs) that stay hidden for months or years.

There are also straight-up cybercriminals chasing money (ransomware gangs, phishing scammers) and organized crime rings.

Common Attack Surfaces:

An attack surface is the vulnerability through which the attacker can access the information of the user or in simple terms any spot where a hacker can poke, prod, or sneak in.

- Web applications — Anything you access in a browser: login pages, search bars, comment forms. Super common targets.
- Mobile apps — Your phone apps store data locally, ask for permissions, talk to servers.
- APIs — The behind-the-scenes connections apps use to talk to each other (e.g., your banking app pulling transaction data).
- Networks — Wi-Fi, routers, open ports, Bluetooth — places where data travels or devices connect.
- Cloud infrastructure — AWS, Google Cloud, Azure setups. Misconfigured storage buckets (public S3 buckets leaking files) are a classic mistake.

OWASP Top 10 (2025 Edition):

OWASP stands for Open Web Application Security Project. The Top 10 list contains the most wanted list of web app dangers. These 10 aren't just a checklist; they're prioritized based on how often they appear in real tests and how badly they hurt when exploited. The key points with why they're risky are as follows:

- A01:2025 - Broken Access Control — Apps let users see or do things they shouldn't (e.g., changing someone else's profile by tweaking a URL). Super common and dangerous—leads to data theft or full takeovers.
- A02:2025 - Security Misconfiguration — Jumped high in recent times. Wrong settings, like leaving debug mode on or using default passwords in cloud setups. Easy to exploit in complex environments.

- A03:2025 - Software Supply Chain Failures — Big expansion from old "vulnerable components." Covers hacked libraries, tainted builds, or compromised dependencies—SolarWinds-style attacks hit here.
- A04:2025 - Cryptographic Failures — Weak or missing encryption—data gets exposed in transit or at rest.
- A05:2025 - Injection — Old classic like SQL injection: bad input tricks the app into running harmful commands.
- A06:2025 - Insecure Design — Flaws baked in from the start (poor threat modeling, missing business logic checks).
- A07:2025 - Identification and Authentication Failures — Weak logins, session hijacking, credential stuffing.
- A08:2025 - Software and Data Integrity Failures Failing to verify code, updates, or data (no signatures, insecure deserialization, tampered pipelines). Dangerous because attackers insert backdoors or malware post-deployment, often staying hidden for months.
- A09:2025 - Security Logging and Monitoring Failures Weak, missing, or insecure logging with no real-time alerts. Dangerous because breaches go unnoticed—attackers operate freely without detection or fast response.
- A10:2025 - Mishandling of Exceptional Conditions Poor error handling: leaking secrets in stack traces, failing open, or crashing on bad input. Dangerous because small bugs turn into big leaks, bypasses, or DoS—easy to find with fuzzing.

Attackers love them because tools automate finding/exploiting many (scanners for misconfigs/injection, credential stuffing kits, etc.). One weak category can let attackers' chain into full compromise.

Some of the daily and very common attack surfaces are

- Email (Gmail, Outlook) → Web interface, mobile client, network protocols (IMAP/SMTP), APIs, cloud backends.
- WhatsApp → Mobile app, encrypted chats (but metadata/backups vulnerable), linked devices, call APIs.
- Banking apps → Mobile/web front, strong auth layers, transaction APIs, encrypted networks, cloud databases.

Dataflow:

- You (User) → Open the app on your phone/laptop and type/input something.
- Application (Client) → Your device processes it, adds auth tokens, encrypts if needed, sends over the internet (usually HTTPS) to the backend.
- Server → Receives, checks if you're legit, runs logic (e.g., "is this transfer allowed?"), talks to the database.
- Database → Stores or pulls the actual data (messages, balances).
- Response comes back the same way.

Attack are spots everywhere like,

- At your device: Malware, phishing texts, stolen phone.
- App to server: Snooping if encryption weak, injection tricks, fake API calls.
- Server itself: Bugs in code, bad configs, insider sabotage.

- Server to database: Over-privileged queries, unencrypted storage.
- Whole path: DDoS floods, ransomware locks files, logging misses hide the crime.

Cybersecurity is all about protecting three things—keeping stuff private, making sure it stays true, and ensuring it's always there when you need it. Bad guys range from clueless teens with scripts to shadowy government teams. Every app you use has multiple weak points along the chain from your finger tap to the distant server. The OWASP list reminds us the biggest dangers often come from sloppy access, bad setups, or trusting third-party code too much. The fix according to me is Layer defences: encrypt everything possible, check inputs ruthlessly, watch logs closely, patch fast, and never assume "inside" means safe.