Cyber Security Fundamentals:

1. Defining Cyber Security & The CIA Triad:

Cyber security is the practice of protecting systems, networks, and programs from digital attacks. At its core lies the CIA Triad, a model designed to guide policies for information security.

• Confidentiality: Ensuring data is accessible only to authorized users.
Example: In Banking, encryption ensures your account balance is only visible to you and bank staff.

• Integrity: Ensuring data is accurate and has not been tampered with.
Example: In Social Media, integrity ensures that a post you write isn't altered by a third party before it reaches your friends' feeds.

• Availability: Ensuring systems and data are accessible when needed.
Example: A Banking App must stay online during a holiday weekend so users can transfer funds.

SmartArt Suggestion: Use a Cycle or Pyramid diagram to represent the CIA Triad (Confidentiality, Integrity, Availability).

2. The Threat Landscape: Types of Attackers:

Attacker Type	Motivation	Skill Level
Script Kiddies	Thrill or bragging rights	Low; use pre-made tools
Insiders	Revenge or financial gain	Varies; have authorized access
Hacktivists	Political or social change	Medium; target specific organizations
Nation-State Actors	Espionage or disruption	High; government-funded/sophisticated

3. Common Attack Surfaces:

• Web Applications: Vulnerable to login bypasses or data theft.

• Mobile Apps: Risk of insecure data storage on the device.

• APIs: The "bridges" between software that can be exploited if not authenticated.

• Networks: Wi-Fi or internal systems susceptible to eavesdropping.

• Cloud Infrastructure: Misconfigured storage buckets (like AWS S3) leading to data leaks.

4. OWASP Top 10 Highlights:

• Broken Access Control: Users can act outside their intended permissions (e.g., accessing someone else's account).

• Cryptographic Failures: Sensitive data (passwords/PII) is not encrypted, leading to exposure.

• Injection: Hostile data is sent to an interpreter (like SQL) to trick it into executing unintended commands.

5. Preventing OWASP Top 10 Vulnerabilities

→ Broken Access Control

- Enforce role-based access control (RBAC).
- Use the principle of least privilege.
- Regularly test authorization rules with penetration testing.

→ Cryptographic Failures

- Encrypt sensitive data at rest and in transit (AES, TLS).
- Never store passwords in plain text; use bcrypt or Argon2.
- Rotate and manage keys securely with a Key Management System (KMS).

→ Injection

- Use parameterized queries (e.g., prepared statements in SQL).
- Validate and sanitize all user inputs.
- Employ ORM frameworks to reduce direct query handling.

→ Insecure Design

- Apply threat modeling during system design.

- Document and enforce secure coding practices.
- Regularly review architecture for potential flaws.

→ Security Misconfiguration

- Disable default accounts and passwords.
- Keep software and frameworks updated.
- Automate configuration checks with tools like Ansible or Chef.

→ Vulnerable and Outdated Components

- Maintain an inventory of dependencies.
- Use tools like OWASP Dependency-Check or Snyk.
- Apply patches promptly.

→ Identification and Authentication Failures

- Implement multi-factor authentication (MFA).
- Lock accounts after repeated failed login attempts.
- Use secure session management (short expiry, regeneration).

→ Software and Data Integrity Failures

- Verify integrity of software updates with digital signatures.
- Use CI/CD pipelines with automated security checks.
- Monitor for unauthorized changes in critical files.

→ Security Logging and Monitoring Failures

- Enable centralized logging (e.g., SIEM systems).
- Monitor logs for suspicious activity.
- Establish an incident response plan.

→ Server-Side Request Forgery (SSRF)

- Validate and sanitize all URLs before requests.
- Restrict outbound traffic from servers.
- Use network segmentation to limit damage.

6. Mapping Daily Apps to Attack Surfaces:

Application	Primary Attack Surface	Potential Risk
Email	Network / Web App	Phishing links; Credential theft

WhatsApp	Mobile Device / API	Malware on phone; Interception of metadata
Banking App	API / Cloud	Unauthorized fund transfers; Data breach

6. Data Flow & Attack Points:

The Flow:

- User: Enters credentials into a UI.
- Application: Processes the request and sends it to the API.
- Server: Validates the logic and identity.
- Database: Retrieves or stores the requested information.

Where Attacks Happen:

- At the User level: Phishing or Keyloggers.
- Between User & Server: "Man-in-the-Middle" (MitM) attacks.
- At the Server: Injection attacks or Denial of Service (DoS).
- At the Database: Unauthorized data exfiltration.

Flowchart Suggestion: Create a diagram showing User → Application → Server → Database with callouts at each stage indicating possible attack points.

9. Summary & Conclusion:

Cyber security is not a single product, but a constant process of balancing the CIA Triad. By understanding that attackers range from bored teenagers (Script Kiddies) to highly funded Nation-States, we can better harden our Attack Surfaces. Using frameworks like the OWASP Top 10 allows us to secure the journey of data from the User to the Database, ensuring that every step of the digital flow is encrypted and authenticated.