Unit 1: Fundamentals of Access Control

Q1: What is Access Control and what is its Purpose?

Definition:

• Access control is a security mechanism that regulates who can view or use resources (data, files, systems) in a computing environment,.

• It ensures that only authorized users or processes can access specific resources, preventing unauthorized interference.

Purpose (The CIA Triad):

1. **Confidentiality:** Ensures sensitive data is disclosed only to authorized users (e.g., stopping a student from seeing another student's grades).

2. **Integrity:** Prevents unauthorized modification or deletion of data, ensuring accuracy,.

3. **Availability:** Ensures legitimate users have timely access to resources when needed (e.g., employees can always access email).

Example: In a company portal, an **Employee** can only view their own payslip, while an **HR Manager** can view payslips for all staff. A random outsider cannot access the portal at all,.

Q2: What are the Basic Components of Access Control?

There are three core components involved in any access decision:

1. **Subject (Who):** The active entity requesting access. This can be a human user, a program, or a process,.

2. **Object (What):** The passive entity or resource being accessed. This includes files, databases, printers, or tables.

3. **Access Rights (How):** The permissions or actions the subject is allowed to perform on the object (e.g., Read, Write, Execute, Delete),.

Example:

• **Subject:** Alice (Student).

• **Object:** ExamResults.pdf.

• **Access Right:** Read-Only.

Q3: Difference between Identification, Authentication, and Authorization?

1. **Identification:** The user claims an identity (e.g., entering a Username).

2. **Authentication:** The system verifies the identity is real (e.g., entering a Password, OTP, or Fingerprint),.

3. **Authorization:** After verification, the system determines what the user is allowed to do (e.g., User is allowed to *view* the file but not *delete* it),.

Unit 2: Policies, Models, and Mechanisms

Q1: Compare DAC, MAC, and RBAC Models.

These are the three primary frameworks for managing permissions:

1. Discretionary Access Control (DAC):

- **Concept:** The **owner** of the resource decides who gets access,.

- **Pros:** Flexible and user-friendly.

- **Cons:** Less secure; users might accidentally share sensitive data (e.g., Trojan horses).

- **Example:** In Windows, you create a folder and right-click to share it with a specific friend.

2. Mandatory Access Control (MAC):

- **Concept:** Controlled by a **central authority** using security labels (e.g., Top Secret, Confidential). Users cannot change permissions,.

- **Rules:**

  - *No Read Up:* A user with "Secret" clearance cannot read "Top Secret" docs.

  - *No Write Down:* A user with "Top Secret" clearance cannot write to "Public" files (to prevent leakage).

- **Example:** Military or Intelligence agency databases.

3. Role-Based Access Control (RBAC):

- **Concept:** Access is assigned to **Roles** (Job functions), and users are assigned to Roles.
- **Pros:** Simplifies management in large organizations (Enterprise usage).
- **Example:** A "Doctor" role can update patient records. If Alice is hired as a doctor, she is assigned the "Doctor" role and automatically gets those permissions.

Q2: Explain Access Control Lists (ACL) vs. Capability Lists.

These are mechanisms to implement the models above.

- **Access Control List (ACL) [Object-Centric]:**
    - A list attached to the **Object** (File) specifying which users can access it.
    - *Structure:* File A -> {Alice: Read, Bob: Write}.
    - *Limitation:* Hard to determine everything a specific user can access across the whole system.
- **Capability List [User-Centric]:**
    - A list (or token) attached to the **User** specifying which objects they can access.
    - *Structure:* Alice -> {File A: Read, File B: Write}.
    - *Limitation:* Hard to revoke access (e.g., if you want to block everyone from File A, you have to search every user's list).

Unit 3: Role-Based Access Control (RBAC)

Q1: Explain Core RBAC vs. Hierarchical RBAC.

• **Core RBAC:** The base model involving Users, Roles, Permissions, and Sessions. Users are assigned roles; roles define permissions.

• **Hierarchical RBAC:** Introduces structure (Parent/Child roles). Higher-level roles **inherit** permissions from lower-level roles,.

  ◦ **Example:** A "Senior Manager" automatically has all the permissions of a "Junior Manager" plus extra rights.

Q2: Explain Static vs. Dynamic Separation of Duties (Constraints).

Constraints prevent conflicts of interest (fraud prevention).

1. **Statically Constrained RBAC (SSoD):**

  ◦ **Rule:** A user cannot be **assigned** two conflicting roles ever,.

  ◦ **Example:** One person cannot be hired as both "Cashier" and "Auditor" because they could steal money and hide the evidence.

2. **Dynamically Constrained RBAC (DSD):**

  ◦ **Rule:** A user can have two conflicting roles, but cannot **activate** them at the same time (in the same session),.

  ◦ **Example:** A bank employee has rights to "Create Loan" and "Approve Loan," but the system stops them from approving a loan they just created in the current login session.

Q3: What are the Limitations of RBAC?

• **Role Explosion:** In large companies, too many specific roles are created (e.g., "Nurse-Day-Shift-Ward-3"), making management chaotic,.

• **Lack of Context:** Traditional RBAC doesn't consider time or location (e.g., allowing login only during office hours),.

Unit 4: Smart Card Security

Q1: Explain Smart Card Architecture and Memory Organization.

A smart card is a plastic card with an embedded chip (microprocessor) used for secure authentication.

• **Operating System (COS):** Manages the hardware and files. It enables the card to talk to card readers.

• **Memory Types:**

1. **ROM:** Stores the Operating System (Permanent, cannot be changed),.
2. **EEPROM:** Stores user data (PINs, Balances, Files). Data stays when power is off (Non-volatile),.
3. **RAM:** Temporary working memory for calculations. Data is lost when power is off,.

Q2: Explain the Smart Card File System.

It follows a tree hierarchy similar to a PC:

1. **Master File (MF):** The Root directory (Main folder),.

2. **Dedicated Files (DF):** Folders acting as groupings for applications (e.g., a "Banking" folder, an "ID" folder),.

3. **Elementary Files (EF):** The actual files containing data (e.g., "Account Balance," "User Name"),.

Q3: What is the Smart Card Life Cycle?

There are 5 phases,:

1. **Fabrication:** Making the physical chip.

2. **Pre-personalization:** Loading the OS and basic files.

3. **Personalization:** Loading specific user data (Name, PIN, Keys).

4. **Utilization:** The user actively uses the card for transactions.

5. **End-of-Life:** The card is destroyed or permanently blocked.

Unit 5: Cloud Security and Trends

Q1: What are the Security Risks in Cloud Data?

- **Data Breach:** Unauthorized access due to weak passwords or hacking.

- **Data Loss:** Accidental deletion or provider failure without backup.

- **Insider Threats:** Malicious employees at the Cloud Provider or the client company misusing access.

- **Shared Technology Issues:** Vulnerabilities in the cloud software affecting multiple clients (tenants).

Q2: What is Cloud Data Auditing and why is it important?

- **Definition:** The systematic checking of cloud operations to ensure security policies and compliance laws (like GDPR) are followed.

- **Importance:**

  - Identifies security gaps or weak points.

  - Ensures accountability (who accessed what data and when).

  - Builds trust between the client and the cloud provider.

Q3: Recent Trends in Database Security.

1. **Zero Trust Architecture:** "Never trust, always verify." Every access request is verified, even if the user is already inside the network,.

2. **AI & Machine Learning:** Using AI to detect strange behavior (anomalies) in real-time to stop attacks,.

3. **Homomorphic Encryption:** A technique allowing data to be processed (calculated) while it is still encrypted, so it is never exposed in plain text,.

4. **Data Security Posture Management (DSPM):** Tools that give a complete view of where sensitive data is stored and its risk level,.