MAC MODEL IN DBMS

Bachelor of Technology
Computer Science and Engineering

Submitted By

NAME – ABHIRUP BAG
ROLL – 13000122082
DEPARTMENT – CSE(B)
SEMESTER – 6
PAPER – DATABASE
MANAGEMENT SYSTEM (PCC-CS601)

MARCH 2025

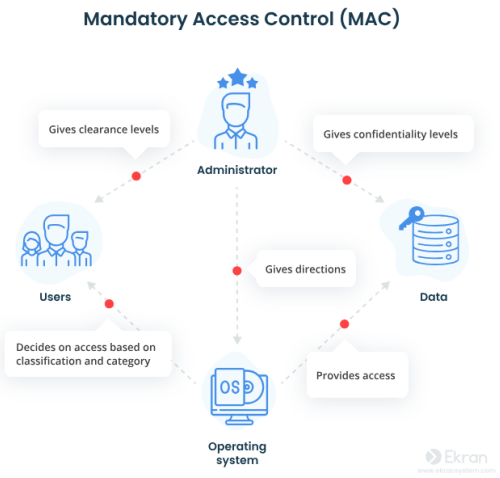
Techno Main
EM-4/1, Sector-V, Salt Lake
Kolkata- 700091
West Bengal
India

Table of Contents

“MAC MODEL IN DBMS”

1. Introduction

Database Management Systems (DBMS) are responsible for storing, retrieving, and managing data in a secure and efficient manner. With the growing concerns of data breaches and unauthorized access, robust security models have become essential. One such model is the Mandatory Access Control (MAC) model, which enforces system-wide security policies and ensures that data access is strictly regulated based on predefined criteria. Unlike discretionary models where users decide who gets access, MAC imposes non-negotiable policies that are centrally managed and enforced by the system.



2. Overview of Access Control Models

In the context of DBMS security, several access control models are widely discussed:

- **Discretionary Access Control (DAC):** Grants permissions based on the identity of the user and allows owners of data to determine access rights.
- **Role-Based Access Control (RBAC):** Assigns permissions to roles rather than individuals, facilitating easier management in large organizations.
- **Mandatory Access Control (MAC):** Enforces access rules based on information classification and user clearance levels, where decisions are not left to user discretion.

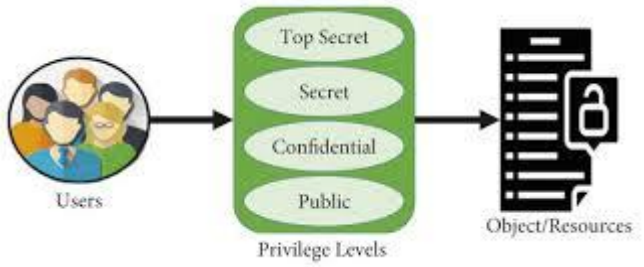
The MAC model stands apart because it provides a rigid, policy-driven framework that minimizes the risks of human error or misuse, making it ideal for high-security environments.

3. Understanding the MAC Model

A. Definition and Core Principles

Mandatory Access Control (MAC) is a security model in which access decisions are made by a central authority based on a system-wide policy rather than by individual data owners. Key principles include:

- **Labelling:** Every data object (e.g., a table, row, or column) is assigned a security label (or classification level).
- **Clearance Levels:** Every user or process is granted a clearance level that determines what data they can access.

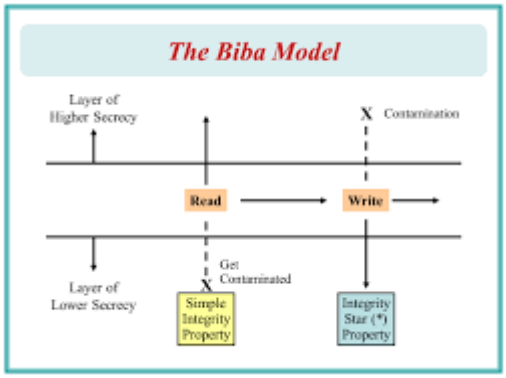


- **Enforced Policy:** Access is granted or denied solely based on the comparison between the data's classification and the user's clearance, often following models such as Bell-LaPadula for confidentiality or Biba for integrity.

B. Theoretical Foundations

The MAC model is rooted in security theories like:

- **Bell-LaPadula Model:** Focuses on maintaining data confidentiality. It enforces "no read up" (users cannot access data at a higher classification than their clearance) and "no write down" (users cannot write data to a lower classification level).
- **Biba Model:** Emphasizes data integrity by implementing rules that are essentially the inverse of Bell-LaPadula.



These models help formalize the security policy that a DBMS must enforce in a MAC environment.

4. How MAC is Mapped to DBMS

A. Data Tagging:

The DBMS assigns security labels to each piece of data, effectively tagging it with its classification. This process is integral for the system to enforce security rules automatically.

B. User Authentication and Clearance Verification:

When a user attempts to access data, the DBMS verifies the user's clearance against the security label attached to the data. The access is permitted only if the user's clearance is equal to or higher than the data's classification.

C. Policy-Driven Access Control:

The system implements strict policies that are not alterable by individual users. This ensures that no discretionary changes can override the established security framework, thereby reducing the risk of unauthorized data access.

5. Implementation of the MAC Model in DBMS

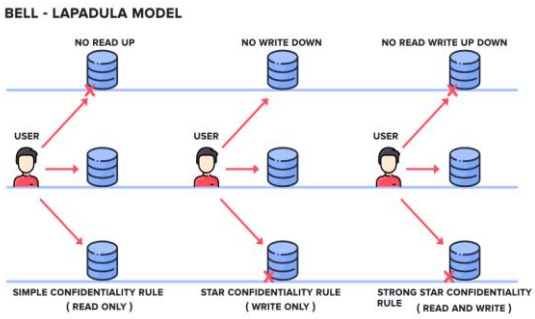
The implementation of the Mandatory Access Control (MAC) model within a Database Management System (DBMS) involves integrating robust security policies into the core architecture of the system. This ensures that all data access operations are regulated strictly by centralized policies rather than discretionary decisions by individual users. The key components and steps in this implementation are as follows:

A. Policy Enforcement in DBMS

i) Automatic Policy Checks:

The MAC model enforces a system-wide security policy that is hard-coded into the DBMS. The system automatically checks every access request against the established security policies. This means that discretionary privileges (like those found in DAC) are overridden by these strict, centralized rules.

ii) "No Read Up" and "No Write Down":



(a) No Read Up: Users cannot read data that is classified at a level higher than their clearance.

(b) No Write Down: Users are prevented from writing data to a level lower than their clearance.

These principles, derived from models such as Bell-LaPadula and Biba, ensure that both confidentiality and integrity are maintained.

B. Integration with Database Architecture

i) <u>Security Labeling:</u>

Every data object—be it a table, row, or even a specific column—is assigned a security label or classification level. These labels (such as "Confidential," "Secret," or "Top Secret") represent the sensitivity of the data and are integral to the MAC model. When a user or process requests access, the DBMS checks the label on the data against the clearance level assigned to the user.

ii) <u>User Clearances and Roles:</u>

In a MAC-based system, each user or process is provided with a specific clearance level that corresponds to the security classifications defined within the system. The DBMS maintains these clearance levels and ensures that each query or data operation adheres to the policies that compare user clearance with data labels.

iii) <u>Policy Enforcement Mechanism:</u>

The DBMS is designed to automatically enforce MAC policies. This means that even if a user has discretionary permissions that might normally allow access, the DBMS will deny any operation that violates the established MAC rules. For example, a user with a "Secret" clearance cannot read data labeled "Top Secret" (“no read up”) or write data to a

lower classification level ("no write down")—a concept derived from models such as Bell-LaPadula and Biba.

iv) <u>Audit and Logging:</u>

To further reinforce security, the DBMS maintains comprehensive logs of access requests, decisions, and any violations. This auditing process is essential for compliance, forensic analysis, and continuous monitoring of the security posture.

C. Steps to Implement MAC in a DBMS

i) Define Security Labels and Hierarchies

(a) Create a label hierarchy (e.g., Top Secret > Secret > Confidential > Public).

(b) Assign labels to database objects:

```
-- Example: Labeling a table in Oracle Label
Security (OLS)
CREATE TABLE Employee (
    ID INT,
    Salary INT LABEL (SECURITY_LABEL =
'RESTRICTED')
);
```

ii) Assign Clearances to Users/Roles

(a) Link user accounts to clearance levels:

```
-- Example: Assigning a clearance level in PostgreSQL (using
extensions like SELinux)
CREATE USER Analyst WITH CLEARANCE 'CONFIDENTIAL';
```

iii) Enforce Access Policies

(a) Use policy engines or DBMS plugins to automate rule enforcement.

(b) <u>Example:</u>

```
-- Oracle Label Security policy enforcement
BEGIN
  SA_POLICY_ADMIN.APPLY_TABLE_POLICY(
    POLICY_NAME => 'Salary_Policy',
    SCHEMA_NAME => 'HR',
    TABLE_NAME => 'Employee',
    TABLE_OPTIONS => 'READ_CONTROL, WRITE_CONTROL'
  );
END;
```

iv) Audit and Monitor Access

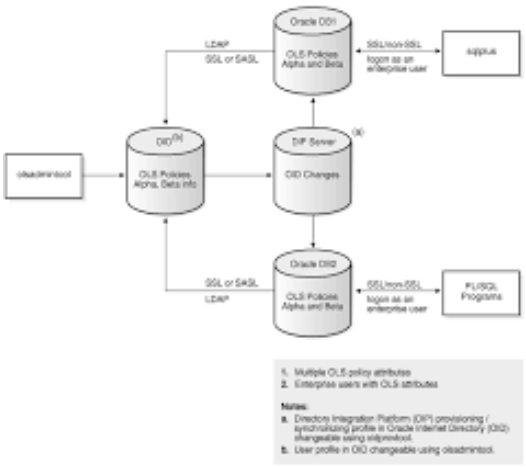
(a) Log access attempts and violations.

(b) Tools: Oracle Audit Vault, Microsoft SQL Server Audit.

D. Example Implementations

Several DBMS products and systems incorporate MAC features, ensuring compliance with high-security standards:

- **Oracle Label Security (OLS):**
  Oracle's DBMS includes an extension that supports MAC by allowing administrators to tag data with security labels and assign clearance levels to users. OLS enforces policies automatically, ensuring that only users with the appropriate clearance can access data at the corresponding classification level.



- **Multi-Level Secure (MLS) Systems:**
  Often used in government and military environments, MLS systems are designed to handle data at various classification levels. In these systems, MAC is a core component, with strict policies implemented to prevent unauthorized access across classification boundaries.

- **Customized Security Solutions:**
  In some enterprise environments, customized MAC solutions are integrated into existing DBMS platforms. These solutions involve additional middleware or security modules that enforce MAC policies, ensuring that the underlying DBMS adheres to stringent security standards without overhauling the core system architecture.

6. Advantages and Limitations of MAC in DBMS

B. Advantages

i) **Enhanced Security:** Since access decisions are made based on a strict, centrally-defined policy, the risk of unauthorized access is minimized.

ii) **Consistency:** All data access adheres to the same policy framework, reducing the possibility of exceptions that could lead to security breaches.

iii) **Compliance:** Many high-security environments (e.g., government or defense) have reg ulatory requirements that necessitate the use of MAC.

C. Limitations

i) **Rigidity:** The inflexible nature of MAC can make it difficult to adapt to dynamic business needs where more granular or discretionary access might be beneficial.

ii) **Complexity:** Setting up and maintaining MAC policies requires a thorough understanding of the classification system and the underlying security needs.

iii) **Performance Overhead:** The continuous checking of security labels and clearances can potentially introduce performance overhead, especially in systems with a high volume of transactions.

7. Practical Use Cases and Scenarios

A. High-Security Environments

i) Government and Military: These sectors often require strict enforcement of security policies where data is classified at multiple levels (e.g., Confidential, Secret, Top Secret).

ii) Critical Infrastructure: Industries like energy, finance, or healthcare may adopt MAC to protect sensitive data and ensure that only authorized personnel have access.

B. Academic and Enterprise Research

i) **Research Databases:** Institutions handling sensitive research data might employ MAC to safeguard proprietary or classified information.

ii) **Enterprise Applications:** Large organizations with sensitive customer data can benefit from the additional layer of security provided by MAC.

8. Best Practices for MAC in DBMS

- **Automate Labeling:** Use scripts or tools to tag objects during schema design.
- **Combine MAC with RBAC:** Assign labels to roles (e.g., Manager_Role: Secret) for scalability.
- **Test Policies Rigorously:** Simulate user access scenarios to avoid over/under-privileging.
- **Leverage Built-In DBMS Features:** Use extensions like Oracle OLS instead of custom code.

9. MAC vs. DAC in DBMS Implementation

Aspect	MAC	DAC
Control	System-enforced via labels.	User-controlled (e.g., GRANT/REVOKE).
Flexibility	Rigid, suited for static environments.	Flexible for ad-hoc permissions.
Use Case	Military, healthcare, finance.	General-purpose databases.
Implementation	Requires labelling and policy engines.	Built-in SQL commands (e.g., PostgreSQL).

10. Conclusion

The MAC model plays a critical role in the landscape of DBMS security by ensuring that data access is tightly controlled through system-enforced policies. By integrating security labels, user clearances, and rigorous policy enforcement, DBMSs that employ MAC can provide a high level of data confidentiality and integrity. While the model's rigidity and complexity may present challenges, its benefits in high-security and regulated environments are invaluable. As data security concerns continue to escalate, the integration of the MAC model within DBMSs remains a cornerstone for protecting sensitive information.

11. References

- **Database System Concepts –** Abraham Silberschatz, Henry F. Korth, S. Sudarshan
- **Database Systems: The Complete Book –** Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom
- **Computer Security: Art and Science –** Matt Bishop
- **Security in Computing –** Charles P. Pfleeger, Shari Lawrence Pfleeger
- **Access Control, Security, and Trust: A Logical Approach –** Ravi Sandhu