BCA Part II
Paper-XI: DBMS Using MS-ACCESS
Topic: Database Security, Integrity and Control

Prepared by: Dr. Kiran Pandey
(School of Computer Science)

Email-id: kiranpandey.nou@gmail.com

DEFINING DATABASE SECURITY, INTEGRITY AND THREATS

SECURITY: Protecting the database from unauthorized access, alteration or deletion.

INTEGRITY: It refers to accuracy or validation of the data. A threat is any situation, event or personnel that will adversely affect the database security and smooth and efficient functioning of the organization.

THREAT: It may be intentional or accidental. Given below are some database security threats….

- Data tampering
- Eavesdropping and data theft
- Falsifying User's identities
- Password related threats
- Unauthorized access to data
- Lack of accountability

SECURITY AND INTEGRITY THREATS

This word has been used several times already. Security threat is any hostile agent which randomly or with use of specialized techniques can obtain or change information in the information system. Random security threats are:

- ***Natural or accidental disasters***- earthquake, water damage or fire. As data as hardware is damaged which leads to the integrity violence and service rejection.
- ***Errors and bugs in hardware and software*** - causes improper application of security policies.
- ***Human errors*** - unintentional violations such as incorrect input or wrong use of applications.

Intended security threats can be categorized according to their originator:

- ***authorized users*** - abuse there privileges
- ***hostile agents*** - various hostile programs - viruses, Trojan horses, back-doors

Some important security features are:

- Views
- Authorization and controls
- User defined procedures or privileges
- Encryption procedures to be limited.

To explain the concept of view, let us consider the example of a bank clerk who needs to know the names of customers of each branch but is not authorized to see specific loan information. The view is defined as follows:

CREATE VIEW CUST_LOAN AS SELECT BRANCHNAME, CUSTOMER_NAME FROM BORROWER, LOAN Where BORROWER.LOAN_NO = LOAN.LOAN_NO;

Since the clerk is authorized to see this view so clerk can execute a query to see the result. **SELECT * from CUST_LOAN**; When the query processor translates the result into a query on actual base table in the database we obtain a query on BORROWER and LOAN tables. This permission must be checked on clerk's query processor begins.

Authorization is a process of allowing the users to perform certain operations on certain data objects in a shared database.

Requirements on DBMS Security

At this moment we have basic image of information system security and we can take a look at concrete aspects that should be covered with DBMS security mechanisms.

1. ***Protection from improper access***- only authorized users should be granted access to objects of DBMS. This control should be applied on smaller objects (record, attribute, value).
2. ***Protection from inference*** - inference of confidential information from available data should be avoided. This regards mainly statistical DBMSs.
3. ***Database integrity*** - partially is ensured with system controls of DBMS (atomic transactions) and various back-up and recovery procedures and partially with security procedures.
4. ***Operational data integrity*** - logical consistence of data during concurrent transactions (concurrency manager), serializability and isolation of transactions (locking techniques).
5. ***Semantic data integrity*** - ensuring that attribute values are in allowed ranges. This is ensured with integrity constraints.
6. ***Accountability and auditing*** - there should be possibility to log all data accesses.
7. ***User authentication*** - there should be unambiguous identification of each DBMS user. This is basis for all authorization mechanisms.
8. ***Management and protection of sensitive data*** - access should be granted only to narrow round of users.
9. ***Multilevel security*** - data may be classified according to their sensitivity. Access granting should then depend on that classification.
10. ***Confinement (subject isolation)*** - there is necessity to isolate subjects to avoid uncontrolled data flow between programs (memory channels, covert channels).

At least five aspects from the previous list must be ensured with special techniques that do not exist in unsecure DBMSs. There are three basic ways to do it:

- ***flow control*** - we control information flows in frame of DBMS

- ***inference control*** - control of dependencies among data
- ***access control*** - access to the information in DBMS is restricted

The main integrity problems in database includes:

• Lost updates

• Uncommitted data

• Inconsistent retrievals

DEFENCE MECHANISM

Generally four levels of defense are recognized for a database security:

• Physical security

• Human factors

• Operating system

• Database system

The basic security standards which technologies can assure are :

CONFIDENTIALITY

Access control - Access to data is controlled by means of privileges, roles and user accounts. Authenticated users – Authentication is a way of implementing decisions of whom to trust. It can be employ passwords, finger prints etc. Secure storage of sensitive data – It is required to prevent data from hackers who could damage the sensitive data. Privacy of communication - The DBMS should be capable of controlling the spread of confidential personal information from unauthorized people such as credit cards etc.

INTEGRITY

Integrity contributes to maintaining a secure database by preventing the data from becoming invalid and giving misleading results. It consists of system and object privileges control access to applications tables and system commands so that only authorized users can change the data. Integrity constraints are applied to maintain the correctness and validity of the data in the database. Database must be

protected from viruses so firewalls and anti-viruses should be used. Ensures that access to the network is controlled and data is not vulnerable to attacks during transmission across network.

AVAILABILITY

Data should always be made available for the authorized user by the secure system without any delays. Availability is often thought of as a continuity of service assuring that database is available. Denial of service attacks are attempts to block authorized user's ability to access and use the system when needed. It has number of aspects such as:

Ease of use – Resources managed by users for working with databases should be effectively managed so that it is available all the time to valid users.

Flexibility – Administrators must have all the relevant tools for managing user population.

Scalability - System performance should not get affected by the increase in number of users or processes which require services from system.

Resistance – User profiles must be defined and the resource used by any user should

Database Recovery plays a very important role in restoring database to a correct state in the case of failure. Few common failures include System crashes, Media failures. Application Software errors, Natural physical disasters and Carelessness.

Some Recovery Concepts

• **Backup mechanism** – it makes periodic backup copies of the database.

• **Logging concept –** that keeps the track of current state of transaction and the changes made in the database.

• **Check pointing mechanism** – that enables update to be made permanent. The choice of the best possible strategy depends upon the extent of damage that had occurred to the database. If there has been a physical damage like disk crash then the last backup copy of the data is restored. However if database has become inconsistent but not physically damaged then changes caused inconsistency must

be undone. It may also be required to redo some transactions so as to ensure that the updates are reflected in the database.

INTEGRITY

Integrity helps in maintaining a secure database by preventing the data from becoming invalid and giving misleading results. It consists of system and object privileges control access to applications tables and system commands so that only authorized users can change the data. Integrity constraints are applied to maintain the correctness and validity of the data in the database. Constraints can be defined in three ways:

• **Business Constraints** A value in one column may be constrained by value of some another or by some calculation or formulae.

• **Entity Constraints** – Individual columns of a table may be constrained e.g. Not null.

•**Referential Constraints** – Sometimes referred to as key constraints. e.g. Table two depends upon table one.

Benefits of Constraints

- It guarantees integrity and consistency
- Is defined as a part of table definition
- Applies across all applications
- Cannot be circumvented
- Application development and productivity
- Requires no special programming
- Easy to specify and maintain
- Defined once only

AUDITING AND CONTROL

Auditing is the monitoring and recording of selected user database actions. It can be based on individual actions, such as the type of SQL statement executed, or on combinations of factors that can include user name, application, time, and so on.

Security policies can trigger auditing when specified elements in an Oracle database are accessed or altered, including the contents within a specified object. Auditing is typically used to:

- Enable future accountability for current actions taken in a particular schema, table, or row, or affecting specific content
- Deter users (or others) from inappropriate actions based on that accountability
- Investigate suspicious activity

  For example, if some user is deleting data from tables, then the security administrator might decide to audit all connections to the database and all successful and unsuccessful deletions of rows from all tables in the database.

- Notify an auditor that an unauthorized user is manipulating or deleting data and that the user has more privileges than expected which can lead to reassessing user authorizations
- Monitor and gather data about specific database activities

  For example, the database administrator can gather statistics about which tables are being updated, how many logical I/Os are performed, or how many concurrent users connect at peak times.

- Detect problems with an authorization or access control implementation

  For example, you can create audit policies that you expect will never generate an audit record because the data is protected in other ways. However, if these policies do generate audit records, then you will know the other security controls are not properly implemented.

Audit records include information about the operation that was audited, the user performing the operation, and the date and time of the operation. Audit records can be stored in either a data dictionary table, called the **database audit trail**, or in operating system files, called an **operating system audit trail**.

When auditing the controls of a database, the auditor would check to see that the following controls have been implemented and maintained to ensure database integrity and availability:

- Definition standards
- Data backup and recovery procedures
- Access controls
- Only authorized personnel can update the database
- Controls to handle concurrent access problems such as multiple users trying to update the same record at the same time
- Controls to ensure the accuracy, completeness and consistency of data elements and relationships.
- Checkpoints to minimize data loss
- Database re-organizations
- Monitoring database performance
- Capacity planning
- Who can access the database without going through the application?

When we speak of who can access the database, we have already identified one of the major audit concerns and that is what access does the DBA have? As everyone knows the DBA basically has the "keys to the kingdom" and can do (read, write, change, delete) anything. What you have to make sure of is that someone is watching. Someone is monitoring (logging) the actions the DBA takes. And the DBA, doesn't have the ability to de-activate the log nor do they have access to the log.

It goes without saying that Access Control is the number one issue with database management systems. That being said let's not forget to audit disaster recovery and restoration, patch management, change management, incident logging and all the other issues an auditor should look for.

There is another issue that auditors need to deal with when auditing DBMS and that is to perform some type of data integrity testing. Data integrity testing is a set of substantive tests (NOTE: Substantive not Compliance testing) that examines accuracy, completeness, consistency and authorization of data presently held in a system. There are two common types of data integrity tests; **relational and referentia**l. Relational integrity tests are performed at the data element and record-based levels. It is enforced through data validation routines built into the application or by defining the input condition constraints and data characteristics at the table definition in the database stage. Sometimes it is a combination of both.