Chapter 6: Database Security and Authorization

**Introduction to DB Security Issues**

# **Database Security**

**Database security** is the mechanisms that protect the database against intentional(hone bilo) or accidental threats. Database security encompasses hardware, software, people, and data.

It is Protection of information contained in the database against unauthorized access, modification or destruction.

Database security aims to minimize losses caused by anticipated events in a cost-effective manner without unduly constraining the users.

In recent times, computer based criminal activities have significantly increased and are forecast to continue to rise over the next few years.

A good database security management system has not only the following characteristics:

* data independence,
* shared access,
* minimal redundancy,
* data consistency, and data integrity but,

also

* privacy, integrity, and availability.
* **Privacy** signifies that an unauthorized user cannot disclose (open) data. Ethical and legal rights that individuals have with regard to control over the dissemination and user of their personal information
* **Integrity** ensures that an unauthorized user cannot modify data. Mechanism that is applied to ensure that the data in the database is correct and consistent
* **Availability** ensures that data be made available to the authorized user unfailingly
* **Copyright** ensures the native rights of individuals as a creator of information.
* **Validity** ensures activities to be accountable by law.

When we talk about the levels of security protection, it may start from organization & administrative security, physical & personnel security, communication security and Information systems security. Database security and integrity is about protecting the database from being inconsistent and being disrupted(betebete). We can also call it database misuse.

Database misuse could be ***Intentional*** or ***Accidental***, where accidental misuse is easier to cope(control) with than intentional misuse.

***Accidental inconsistency occurs due to:***

* System crash during transaction processing
* Anomalies due to concurrent access
* Anomalies due to redundancy
* Logical errors

***Intentional misuse could be:***

* Unauthorized reading of data
* Unauthorized modification of data or
* Unauthorized destruction of data

Most systems implement good **Database Integrity** to protect the system from accidental misuse, whereas, there are many computer based measures to protect the system from intentional misuse, which is termed as **Database Security** measures.

## **Threats to Databases**

Threats to databases can result in the loss or degradation of some or all of the following commonly accepted security goals:

* Theft and Fraud
* Loss of Confidentiality
* Loss of Privacy
* Loss of Integrity
* Loss of Availability

These situations broadly represent areas in which the organization should seek to reduce risk, that is, the possibility of incurring loss or damage. In some situations, these areas are closely related such that an activity that leads to loss in one area may also lead to loss in another. In addition, events such as fraud or loss of privacy may arise because of either intentional or unintentional acts, and do not necessarily result in any detectable changes to the database or the computer system.

**Theft and fraud:** Theft and fraud affect not only the database environment but also the entire organization. As it is people who perpetrate such activities, attention should focus on reducing the opportunities for this occurring.

Theft and fraud do not necessarily alter data, as is the case for activities that result in either

* loss of confidentiality or
* loss of privacy.

**Loss of integrity:** Database integrity refers to the requirement that information be protected from improper modification. Modification of data includes creating, inserting, and updating data; changing the status of data; and deleting data.

Integrity is lost if unauthorized changes are made to the data by either intentional or accidental acts. If the loss of system or data integrity isn’t corrected, continued use of the contaminated system or corrupted data could result in **inaccuracy, fraud, or erroneous decisions**.

Loss of data integrity results in

* **invalid or**
* **corrupted data**, which may seriously affect the operation of an organization.

**Database Integrity** constraints contribute to maintaining a secure database system by preventing data from becoming invalid and hence giving misleading or incorrect results.

* ***Domain Integrity*** means that each column in any table will have set of allowed values and cannot assume any value other than the one specified in the domain.
* ***Entity Integrity*** means that in each table the primary key (which may be composite) satisfies both of two conditions:

That the primary key is unique within the table and

That the primary key column(s) contains no null values.

* ***Referential Integrity*** means that in the database as a whole, things are set up in such a way that if a column exists in two or more tables in the database (typically as a primary key in one table and as a foreign key in one or more other tables), then any change to a value in that column in any one table will be reflected in corresponding changes to that value where it occurs in other tables. This means that the RDBMS must be set up so as to take appropriate actions to spread a change—in one table—from that table to the other tables where the change must also occur.

The effect of the existence and maintenance of referential integrity is, in short, that if a column exists in two or more tables in the database, every occurrence of the column will contain only values that are consistent across the database.

* ***Key constraints*** in a relational database, there should be some collection of attributes with a special feature used to maintain the integrity of the database. These attributes will be named as Primary Key, Candidate Key, Foreign Key, and etc. These Key(s) should obey some rules set by the relational data model.
* ***Enterprise Constraint*** means some business rules set by the enterprise on how to use, manage and control the database

**Loss of availability** *Database availability* refers to making objects available to a human user or a program who/which has a legitimate(authorized) right to those data objects.

* *Loss of availability* occurs when the user or program cannot **access** these objects

**Loss of confidentiality** Database confidentiality refers to the protection of data from unauthorized disclosure. Whereas **privacy** refers to the need to protect data about individuals. The impact of unauthorized disclosure of confidential information can range from violation of the Data Privacy Act to the jeopardization of national security. Unauthorized, unanticipated, or unintentional disclosure could result in loss of public confidence, embarrassment, or legal action against the organization. Breaches of security resulting in loss of confidentiality could, for instance, lead to loss of competitiveness, and loss of privacy could lead to legal action being taken against the organization.

**Threat** may be any situation or event, whether intentional or accidental, that may adversely affect a system and consequently the organization. A threat may be caused by a situation or event involving a person, action, or circumstance that is likely to bring harm to an organization. The harm to an organization may be ***tangible*** or ***intangible.***

* ***Tangible –*** loss of hardware, software, or data
* **Intangible** – loss of credibility(hakegna) or client confidence

# **Countermeasures: Computer Based Controls**

An organization deploying a database system needs to identify the types of threat it may be subjected to and initiate appropriate plans and ***countermeasures***, bearing in mind the costs of implementing each.

Database Management Systems supporting multi-user database system must provide a database security and authorization subsystem to enforce limits on individual and group access rights and privileges.

## Security Issues and general considerations

* **Legal**, **ethical** and **social** issues regarding the right to access information
* **Physical control** issues regarding how to keep the database physically secured.
* **Policy** issues regarding privacy of individual level at enterprise and national level
* **Operational** consideration on the techniques used (password, etc) to access and manipulate the database
* **System** level security including operating system and hardware control
* Security levels and security policies in enterprise level

The designer and the administrator of a database should first identify the possible threat that might be faced by the system in order to take counter measures.

## **Levels of Security Measures**

Security measures can be implemented at several levels and for different components of the system. These levels are:

1. **Physical Level:** concerned with securing the site containing the computer system. The backup systems should also be physically protected from access except for authorized users. In other words, the site or sites containing the computer systems must be physically secured against armed or sneaky(ashimur) entry by intruders.(talika geb)
2. **Human Level:** concerned with authorization of database users for access the content at different levels and privileges.
3. **Operating System:** concerned with the weakness and strength of the operating system security on data files. Weakness may serve as a means of unauthorized access to the database. No matter how secure the database system is, weakness in operating system security may serve as a means of unauthorized access to the database. This also includes @***protection of data in primary and secondary memory from unauthorized access.***
4. **Database System:** concerned with data access limit enforced by the database system. Access limit like password, isolated transaction and etc. Some database system users may be authorized to access only a limited portion of the database. Other users may be allowed to issues queries, but may be forbidden to modify the data. It is the responsibility of the database system to ensure that these authorization restrictions are not violated.
5. **Application Level:** Since almost all database systems allow remote access through terminals or networks, software-level security with the network software is as important as physical

**The following are computer-based security controls for a multi-user environment:**

## **Authorization (give permission)**

Granting a right privilege enables a subject to have legitimate access to a system object. Authorization controls can be built into the software, and govern not only what system or object a specified user can access, but also what the user may do with it.

Authorization controls are sometimes referred to as ***access controls***. The process of authorization involves authentication of ***subjects*** (i.e. a user or program) requesting access to ***objects*** (i.e. a database table, view, procedure, trigger, or any other object that can be created within the system)

**Authentication (check the reality)**

All users of the database will have different access levels and permission for different data objects, and

authentication is the process of checking whether the user is the one with the privilege for the access level. Is the process of checking the users are who they say they are. Each user is given a unique identifier, which is used by the operating system to determine who they are. Thus, the system will check whether the user with a specific username and password is trying to use the resource. Associated with each identifier is a password, chosen by the user and known to the operation system, which must be supplied to enable the operating system to authenticate who the user claims to be.

**Authorization/Privilege**

Authorization refers to the process that determines the mode in which a particular (**previously authenticated**) client is allowed to access a specific resource controlled by a server.

Most of the time, authorization is implemented by using Views.

* Views are unnamed relations containing part of one or more base relations creating a customized/personalized view for different users.
* Views are used to hide data that a user needs not to see.

**Forms of user authorization**

There are different forms of user authorization on the resource of the database. These forms are privileges on what operations are allowed on a specific data object.

**User authorization on the data/extension**

1. **Read Authorization**: the user with this privilege is allowed only to read the content of the data object.
2. **Insert Authorization**: the user with this privilege is allowed only to insert new records or items to the data object.
3. **Update Authorization**: users with this privilege are allowed to modify content of attributes but are not authorized to delete the records.
4. **Delete Authorization**: users with this privilege are only allowed to delete a record and not anything else.
   * Different users, depending on the power of the user, can have one or the combination of the above forms of authorization on different data objects.

**User authorization on the database schema**

1. **Index Authorization**: deals with permission to create as well as delete an index table for relation.
2. **Resource Authorization**: deals with permission to add/create a new relation in the database.
3. **Alteration Authorization**: deals with permission to add as well as delete attribute.
4. **Drop Authorization**: deals with permission to delete and existing relation.

## **Access Controls**

The typical way to provide access controls for a database system is based on the granting and revoking of privileges. A **privilege** allows a user to create or access (that is read, write, or modify) some database object (such as a relation, view, or index) or to run certain DBMS utilities.

Privileges are granted to users to accomplish the tasks required for their jobs. As excessive granting of unnecessary privileges can compromise security: a privilege should be granted to a user only if that user cannot accomplish his or her work without that privilege. A user who creates database object such as a relation or a view automatically gets all privileges on that object. The DBMS subsequently keeps track of how these privileges are granted to other users, and possibly revoked, and ensures that at all times only users with necessary privileges can access an object.

Types of access control

* Discretionary and,
* Mandatory

**Dis\_cretionary Access Control (DAC)**

These are used to grant privileges to users, including the capability to access specific data files, records, or fields in a specified mode (such as read, insert, delete, or update). Grant different privileges to different users and user groups on various data objects. The privilege is to access different data objects. The mode of the privilege could be: Read, Insert, Delete, Update files, records or fields. Is more flexible. One user can have A but not B and another user can have B but not A

**Mandatory Access Control (MAC)**

These are used to enforce multilevel security by classifying the data and users into various security classes (levels) and then implementing the appropriate security policy of the organization. For example, a typical security policy is to permit users at a certain classification (or clearance) level to see only the data items classified at the user’s own (or lower) classification level.

* Classifying data and users into various security classes (or levels) and implementing the appropriate security policy of the organization.
* Each **data object** will have certain **classification** level
* Each **user** is given certain **clearance level**
* Only users who can pass the clearance level can access the data object
* Is comparatively not-flexible/rigid
* If one user can have A but not B then B is accessed by users with higher privilege and we cannot have B but not A

The ability to classify user into a hierarchy of groups provide a powerful tool for administering large systems with thousands of users and objects.

A database system can support one or both of the security mechanisms to protect the data. In most systems, it is better to filter those that are allowed rather than identifying the not allowed. **Since if some object is authorized then it means it is not constrained.**

## **Views**

A view is the dynamic result of one or more relational operations on the base relations to produce another relation. A view is a virtual relation that does not actually exist in the database, but is produced upon request by a particular user. The view mechanism provides a powerful and flexible security mechanism by hiding parts of the database from certain users. Using a view is more restrictive than simply having certain privileges granted to a user on the base relation(s)

## **Backup and recovery**

Backup is the process of periodically taking a copy of the database and log file (and possibly programs) on to offline storage media. A DBMS should provide backup facilities to assist with the recovery of a database following failure. Database recovery is the process of restoring the database to a correct state in the event of a failure.

Journaling is the process of keeping and maintaining a log file (or journal) of all changes made to the database to enable recovery to be undertaken effectively in the event of a failure.

The advantage of journaling is that, in the event of a failure, the database can be recovered to its last known consistent state using a backup copy of the database and the information contained in the log file. If no journaling is enabled on a failed system, the only means of recovery is to restore the database using the latest backup version of the database. However, without a log file, any changes made after the last backup to the database will be lost

## **Integrity**

Integrity constraints contribute to maintaining a secure database system by preventing data from becoming

* invalid and hence giving misleading or incorrect results

## **Encryption**

Authorization may not be sufficient to protect data in database systems, especially when there is a situation where data should be moved from one location to the other using network facilities. Encryption is used to protect information stored at a particular site or transmitted between sites from being accessed by unauthorized users.

Encryption is the encoding of the data by a special algorithm that renders the data unreadable by any program without the decryption key.It is not possible for encrypted data to be read unless the reader knows how to decipher/decrypt the encrypted data.

# **Security at different Levels of Data**

Almost all RDBMSs provide security at different levels and formats of data. This includes:

1. **Relation Level:** permission to have access to a specific relation.
2. **View Level:** permission to data included in the view and not in the named relations
3. **Hybrid (Relation/View):** the case where only part of a single relation is made available to users through View.

Any database access request will have the following three major components

1. **Requested Operation:** what kind of operation is requested by a specific query?
2. **Requested Object:** on which resource or data of the database is the operation sought to be applied?
3. **Requesting User:** who is the user requesting the operation on the specified object?

The database should be able to check for all the three components before processing any request. The checking is performed by the security subsystem of the DBMS.

# **Statistical Database Security**

Statistical databases contain information about individuals which may not be permitted to be seen by others as individual records. Such databases may contain information about various populations. Example: Medical Records, Personal Data like address, salary, etc

Such kind of databases should have special security mechanisms so that confidential information about people will not be disclosed for many users. Thus, statistical databases should have additional security techniques which will protect the retrieval of individual records. Only queries with statistical aggregate functions like Average, Sum, Min, Max, Standard Deviation, Mid, Count, etc should be executed. Queries retrieving confidential attributes should be prohibited. Not to let the user make inference on the retrieved data, one can also implement constraint on the minimum number of records or tuple in the resulting relation by setting a threshold.

# **Role of DBA in Database Security**

The database administrator is responsible to make the database to be as secure as possible. For this the DBA should have the most powerful privilege than every other user. The DBA provides capability for database users while accessing the content of the database.

The major responsibilities of DBA in relation to authorization of users are:

1. **Account Creation:** involves creating different accounts for different **users** as well as **user GROUPS.** This action creates a new account and password for a user or a group of users to enable access to the DBMS.
2. **Security Level Assignment**: involves in assigning different users at different categories of access levels. This action consists of assigning user accounts to the appropriate security clearance level.
3. **Privilege Grant:** involves giving different levels of privileges for different users and user groups. This action permits the DBA to grant certain privileges to certain accounts.
4. **Privilege Revocation:** involves denying or canceling previously granted privileges for users due to various reasons. This action permits the DBA to revoke (cancel) certain privileges that were previously given to certain accounts
5. **Account Deletion:** involves in deleting an existing account of users or user groups. Is similar with denying all privileges of users on the database.