Unit 5 / Database Security

1. Introduction

Database security refers to protection of data from unauthorized access, unauthorized alteration etc. Data has to be protected against misuse, any intentionally made inconsistency and also any unintentionally introduced data.

Data can be protected by providing security measures at various levels like at operating system level, database level, network level, physical level, human level etc.

Security measures need to be defined at all levels (database, Operating system network, physical and human), So as to ensure secure and consistent data in the database.

In this chapter, we learn about different measures to ensure a secured access to data in the database.

2. Database Security Concepts

Database security refers to protection of data from malicious access. The data in the database needs to be protected against unauthorized access, unauthorized alteration etc. Data has to be protected against misuse, intentionally made inconsistency and also unintentionally made inconsistency of data.

To protect data, we can have security measures taken at different levels

1

Apr. 2018 – 1M
State different levels of security.

i. **Database system:** Users are grouped and allowed to access only parts of the database. The database ensures that these authorization restrictions are not violated.

ii. **Operating system:** The operating system security should be strong, else it can lead to allow unauthorized access to database.

iii. **Network:** Security measures at the network software level is also the most important, since database can be accessed remotely through terminals.

iv. **Physical:** The computer sites must be physically secured against armed entry by intruders.

v. **Human:** Users must be authorized carefully, so that they don't lead to giving access to intruders.

Thus, security at all the above levels must be maintained to ensure database security.

3. Methods of Database Security

1

Oct. 2017 – 4M
What are different methods used for database security?

Security within operating system is implemented at several levels, ranging from passwords for the access to system to the isolation of concurrent processes running within the system.

There are different ways to implement the security at different levels mentioned above.

There are different techniques for database security which are as follows

Different Techniques for Database Security are as follows:
i. Discretionary Security Mechanisms
ii. Mandatory Security Mechanisms
iii. Access Control Mechanism
iv. Statistical Database Security
v. Data Encryption Technique

i. **Discretionary Security Mechanisms:** 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, write or update mode).

ii. **Mandatory Security Mechanisms:** These are used to enforce multilevel security by classifying the data and users into various security classes (or levels) and then implementing the appropriate security policy of the organization.

iii. **Access Control Mechanism:** It includes provisions for restricting access to the database system as a whole. It is handled by creating user accounts and passwords to control the log–in process by the DBMS.

iv. **Statistical Database Security:** It involves controlling the access to a statistical database which is used to provide statistical information or summaries of values based on various criteria.

v. **Data Encryption Technique:** It is used to protect sensitive data that is being transmitted via satellite or some other type of communication network. The data in the database is encoded using some coding algorithm.

An unauthorized user who accesses encoded data will have difficulty deciphering it, but authorized users are given decoding algorithm to decipher the data.

Database Security and the DBA

1

Oct. 2018 – *4M*
Explain how DBA is responsible for managing database security? What are the privileged commands used by DBA?

The DBA (Database Administration) is the central authority for managing a database system. The DBA's responsibility with respect to database security includes granting privileges to users who need to use the system and classifying users and data in accordance with the policy of the organization.

The DBA has a system account, which provides powerful capabilities that are not made available to regular database users. It is also called a super user account.

2

Oct. 2017 – *1M*
State any two uses of grant privilege.

Oct. 2017 – *3M*
What do you mean by granting and revoking privilege?

DBA privileged commands include the following

1. **Account Creation:** This action creates a new account and password for a user or a group of users to enable them access the DBMS.
2. **Privilege Granting:** Allows the DBA to grant certain privileges to certain accounts.
3. **Privilege Revocation:** Allows the DBA to revoke or cancel certain privileges that were previously given to certain accounts.
4. **Security Level Assignment:** Consists of assigning user accounts to the appropriate security classification level.

Thus the DBA is responsible for the overall security of the DBMS.

Action 1 in the above list is used to control access to the DBMS as a whole, Actions 2 and 3 are used to control discretionary database authorizations and action 4 is used to control mandatory authorization.

1
Oct. 2017 - 5M
Write a short note on Discretionary Access Control.

4. Discretionary Access Control Method

The typical method of enforcing discretionary access control in a DBMS is based on granting and revoking privileges. SQL provides commands to grant and revoke access privileges to users.

4.1 Types of Discretionary Privileges

Two levels for assigning privileges to use the DBMS.
i. The Account Level
ii. The Relation Level

The DBMS must provide selective access to each relation in the database based on specific accounts. *There are two levels for assigning privileges to use the DBMS.*

i. **The Account Level:** The DBA specifies the particular privileges that each account holds independent of the relations in the database.

   For example: Create Schema privilege, Create Table privilege, Create View privilege, the Alter privilege, the Drop privilege, Modify privilege and Select privilege to retrieve information from database. The account level privileges apply to the account in general.

ii. **The Relation Level:** It controls the privilege to access each individual relation or view in the database. It specifies for each user the individual relations on which each type of command can be applied. They may also refer to individual columns (attributes) of relations.

The granting and revoking of privileges follow an authorization model for a discretionary privilege known as **Access Matrix** model or the **Authorization matrix.**

Each row of matrix M represents subjects (users, accounts, programs) and the columns represent objects (relations, records, views, columns).

Each position M (i, j) in the matrix represents the types of privileges (read, write, update) that subject i holds on object j.

To control granting and revoking of privileges on relations, each relation R in a database is assigned an owner account, which is the account that was used when R was created in the first place.

The owner of a relation is given all privileges on that relation. The owner of a relation can pass privileges on any of the owned relations to other users by granting privileges to their accounts. SQL provides the GRANT command for granting the Select, Update, Insert, Delete, References on relation R to other users.

In order to revoke privileges, a Revoke command is included for the revoking of privileges granted on a relation R. Revoke command is useful in cases where it's desirable to grant some privileges to a user temporarily.

For propagation of privileges, the grant command can be used with *'with grant option'*.

For example: If A is the owner of a relation R and A grants a privilege on R to another account B, then privilege when given with the grant option means that B can also grant that privilege on R to other accounts.

So, if A grants privileges on R to B with grant option, then B in turn can grant these privileges on R to a third account C, also with grant option. In this way, privileges on R can propagate to other account without the knowledge of the owner of R.

If the owner account A now revoke the privileges granted to B, all privileges that B propagate based on that privilege should automatically be revoked by the system. This is done by using Revoke command with the *'Cascade option'*.

The **grant statement** used to confer authorization has the following format:

```
grant <Privilege list> on <relation name or view name>
to user list
<with grant option>
```

The privilege list allows the granting of several privileges in one command.

The update privilege may be given either all attributes of the relation or on only some. The list of attributes on which update is to be allowed appears in parenthesis immediately after the update keyword. If the list of attribute is omitted, then grant is given to all attributes of the relation R.

The **revoke command** is used to remove or cancel the given privileges from an user.

The format is

```
revoke <Privilege list> on <relation name or view name>
from <user list> [restrict/ cascade]
```

The 'restrict' option when used stops the Revoke action when there are any other cascading revokes.

Consider the following *example* that uses grant and revoke commands for discretionary access mechanism

Consider the relation schema

```
Employee( fname, Minit, lname, SSN, Bdate, Address, Sex, Salary,
          SuperSSN, Dno )
Department( Dname, dnumber, MgrSSN, Mgrstartdate )
Dept-loc( Dnumber, dlocation )
Project( Pname, Pnumber, Plocation, Dnum )
Works-on( eSSN, Pno, Hours )
Dependent( eSSN, dependent-name, Sex, bdate, relationship )
```

Suppose that all the above relations are owned by X, who wants to grant the following privileges to user accounts A, B, C, D and E:

1. Account A can retrieve or modify any relation except Dependent and can grant any of these privileges to other users.

```
Grant Select, Update, Insert, Delete on Employee, Project,
Works-on, Department, Dept-loc to A with grant option.
```

2. Account B retrieves all attributes of Employee and department except for Salary, MgrSSN, Mgrstartdate.

```
Create View V1 As
Select fname, Minit, bdate, SuperSSN, dno, address, sex, Dname
From Employee, department
Where Employee.dno = department.dnumber
Grant Select On V1 TO B
```

3. Account D can retrieve any attribute of Employee or dependent and can modify dependent.

```
Grant Select on Employee, dependent to D
Grant insert, delete, update on dependent to D
```

4. Account E can retrieve any attribute of Employee but only for employee tuples having Dno = 3;

```
Create View V2 as
Select *
From Employee
Where Dno = 3;
```

Grant Select On V2 to E;

4.2 Role Based Authorization

In this case, the users are identified into groups, depending on the role they play in the organization.

For example, consider the shop-floor clerks in production department. Each shop-floor clerk must have the same type of authorizations to the same set of relations. Whenever a new shop floor clerk is appointed, he/she also must have the same authorization on relations as the other clerks. So, these authorizations must be given to the new clerk, on each of the relations he/she is allowed to access.

An approach to this problem is to first identify which database users are shop-floor clerks, and then the authorizations that every shop floor clerk is to be given.

The system can then use these 2 pieces of information to determine authorization of each user who is a shop-floor clerk. This approach gives rise to the concept of Roles: A set of roles is created in the database. Authorizations can be granted to roles, in exactly the same fashion as they are granted to individual users. Each database user is given a set roles that he/she is authorized to perform.

The *disadvantage* of this role based approach is that the system won't be able to identify as to exactly which shop-floor clerk has performed a particular transaction, since here, users are identified only by their Roles.

Any authorization that can be granted to a user can be granted to a role. Roles are granted to a user just like the authorizations. Just like authorization, a user can propagate his role to other users too.

1

Oct. 2018 – *5M*
Explain the mandatory access control mechanism in detail.

5. Mandatory Access Control Method

The discretionary access control method for database security is an all or nothing method; an user either has or does not have a certain privilege. Mandatory access control mechanism classifies data and user, based on security classes.

The typical security classes used are

TS → TopSecret, S → Secret, C → Confidential, U → Unclassified.

Here, TS > S > C > U. The commonly used method for mandatory access control is the *Bell Lapadula model* that classifies each Subject (user, account, program) and Object (relation, tuple, column, view, operation) into one of the security classifications, TS, S, C, or U.

The classification of a subject S is denoted by Class (S) and classification of an object O is denoted by Class (O).

Two restrictions are enforced on data access based on the Subject/Object classifications

i. The simple security property that states a Subject (S) is not allowed read access to an object O, unless Class (S) ≥ Class (O).

ii. The Star (*) property that states a Subject S is not allowed write access to an Object O, unless Class (S) ≤ Class (O).

The first restriction enforces the obvious rule that no subject can read an object whose security classification is higher than the Subject security clearance.

The second restriction prohibited a Subject from writing an object that has lower security clearance than the Subject security clearance. Violation of this rule would allow information to flow from higher to lower classifications.

Multilevel Relations and Polyinstantiation

A security class is assigned to each object in the database. Objects can be of different granularity like tables, records or even individual columns.

Each row can be assigned a security class. A multilevel table is a table with a property that users with different security clearances will see a different collection of tuples, although they access the same table.

The presence of data objects that appears to have different values to users with different clearances is called *polyinstantiation*. Here, the presence of data is revealed to the users according to their clearance level.

6. Uses of Views in Security Enforcement

Suppose that A_1 wants to give A_3 a limited capability to select from the employee relation and wants to allow A_3 to be able to propagate the privilege. The limitation is to retrieve only the Name, B Date and Address, and only for the tuples with D. No. = 5.

A_1 then can create a view as

```
Create View A3 EMP As
Select Name, BDate, Address
From Employee
Where D.No. = 5;
```

After the view is created, A_1 can grant select on the view A_3 EMP to A_3 as,

```
Grant Select On A3 EMP To A3 With Grant Option;
```

Finally, suppose that A_1 wants to allow A_4 to update only the salary attributes of employee; A_1 can then issue the following command.

```
Grant Update On employee (salary) TO A4;
```

The Update or Insert privilege can specify particular attributes that may be updated or inserted in a relation.

7. Overview of Encryption Technique for Database

Encryption techniques are used to provide security for highly sensitive data. In such cases, data will be stored in an encrypted form. This encrypted data can't be read, unless the user knows to decrypt them. Encryption technique also forms the basis of good schemes for authenticating users to a database.

2

Oct. 2018 – *1M*
What is encryption technique used for?

Apr. 2018 – *5M*
Write short note on encryption techniques in database security.

Encryption Techniques

There are many techniques for encrypting data. Simple encryption techniques do not provide adequate security for data, since any unauthorized user can break the code and decrypt it. An *example* of a simple encryption technique is to substitute each character with its next character in the alphabet.

A good encryption technique should possess the following properties

i. Should be relatively simple so that authorized users can encrypt and decrypt data.

ii. The technique should depend on the encryption key, which will be a parameter of the algorithm.

iii. The encryption key should be such that it is difficult for any intruder to determine.

Public key encryption is an alternative scheme, used instead of the sample data encryption standard. This technique has two keys, a public key and a private key. Each user has a public key and a private key. All *public keys* are published and can be seen by anyone. Each *private key* is known to only one user to whom the key belongs. The data is encrypted using the public key, but decryption requires the private key. Since there are two keys in this scheme, data transfer is very secure. This scheme is secure, but is computationally expensive.

1

Oct. 2018 – *3M*
Write a note on statistical database security.

8. Statistical Database Security

A statistical database is one that contains specific information on individuals or events, but is intended to permit only statistical queries, that is, queries that involve aggregate functions. Security in statistical databases is required since it's possible to infer protected information from answer to permitted statistical queries. By repeatedly firing statistical query on a set of data it is possible to infer any individual data from the set.

Statistical databases are used mainly to produce statistics on various populations. The database may contain confidential data on many individuals, which should be protected from user Access. In this case, users are allowed to retrieve statistical information on the population, such as averages, counts, sums and standard deviations. But allowing users to aggregate functions on statistical databases can lead to other unusual problems.

For example, Suppose a user asks for total–bank–balance for all Customers living in 'Pune' and if there is only one Customer living in Pune, then his account balance information will be divulged by the system, which is again a confidential information. So, even if views are created to provide aggregate data, the above problem will be there. So, a simple way to deal with such potential security breaches is to reject the queries that involve fewer than some predetermined number of records.

In some cases, it is possible to deduce the values of individual tuples from a sequence of statistical queries. This also can be avoided by not permitting statistical queries that retrieve records whose total count is below some threshold. Another way to prohibit retrieval of individual record information is to prohibit sequences of queries that refer repeatedly to the same group of tuples.

Exercises

1. Define an Authorization Matrix.
2. State the different levels of security.
3. State the different priviledges that can be granted, on a relation, to the database users.
4. State the advantages of encryption technique used for data security in a database.
5. State the role of DBA with respect to security.
6. Explain Polyinstantiation.
7. Explain the intuition behind the 2 rules in the Bell-LaPadula model for mandatory Access control.