kdeg
knowledge and data engineering group

Information Management and Data Engineering

CS4D2a – 4CSLL1 – CS3041

Database Security

Séamus Lawless

seamus.lawless@scss.tcd.ie

Integrity v Security

- Integrity and Security are related but they are not the same thing
  - Integrity is concerned with *accidental* corruption
  - Security is concerned with *deliberate* corruption
- Integrity
  - Integrity Constraints
- Security
  - Security Policies
  - Access Control

Database Security

- Ensuring security for large DBs is an important and difficult task
- Many different issues involved
  - legal, social, ethical etc.
- Most countries have Data Protection Legislation
  - requires holders of personal information to take reasonable precautions to ensure that there is no unauthorised access to the data

Access Control

- Prevent unauthorised persons from accessing the system
  - to obtain information
  - to make malicious changes
- DBMS can restrict access to the DB
  - User Accounts
  - Privileges
  - Security Levels

Access Control

- Database Administrator (DBA) is responsible for:
  - User Account Creation
    - Encrypted table maintained by the DBMS
  - Privilege Granting and Revocation
    - Discretionary Access Control
  - Security Level Assignment
    - Mandatory Access Control
  - Role-Based Access

Privileges

- Access privileges can be specified at two levels
  - Account Level
    - DBA can specify the privileges that each account holds independently of the relations in the database
  - Relation Level
    - DBA can control the privilege to access each individual relation or view in the database

Account Level Privileges

- These privileges apply to the capabilities provided to an account
- Examples of privileges include:
  - CREATE SCHEMA
  - CREATE TABLE
  - CREATE VIEW
  - ALTER
  - DROP

Relation Level Privileges

- Can be specified on entire relations or on specific attributes
  - determine what operations can be performed
- Each relation has an “owner”
  - typically the account which created the table
  - This account then controls the granting and revoking of privileges to other accounts for that table

Relation Level Privileges

- Privilege types are:
  - Read Privilege
    - gives an account the ability to use SELECT to retrieve rows from this relation
  - Modification Privileges
    - gives an account the ability to use INSERT, UPDATE and DELETE to modify rows in this relation
  - Reference Privilege
    - gives an account the ability to refer to this relation when specifying integrity constraints

Views

- Views are an important *discretionary authorisation mechanism*
- Allow the owner of a relation(s) to grant partial access to the information contained in that relation
  - Access to a restricted set of attributes
  - Access to a restricted set of rows
- A view acts as a new relation in the DB

Granting Privileges

- Privileges are allocated to users using the GRANT command in SQL
  - GRANT privilege TO user;
  - GRANT privilege ON relation TO user;
- The owner of a relation
  - automatically has all the relation privileges granted to them
  - can use the GRANT command to specify user privileges for that relation

Revoking Privileges

- It is often desirable to remove a privilege from a particular user
  - temporary access
  - abuse of privilege
- In SQL the REVOKE command is used to cancel privileges
  - REVOKE privilege FROM user;
  - REVOKE privilege ON relation FROM user;

Propagation of Privileges

- Whenever the owner A of a relation R grants privileges on R to another user B, the privilege can be given with or without the GRANT OPTION
  - If the GRANT OPTION is given, then B can also grant that privilege on R to other users
- Command Syntax
  - GRANT privilege ON relation TO user WITH GRANT OPTION;

Dangers of Propagation

- A is the owner of relation R
- A grants B the DELETE privilege on R, with GRANT OPTION
- B grants C the DELETE privilege on R, also with GRANT OPTION
- In this way, privileges can propagate without the knowledge of the relation owner
- If A revokes the privilege granted to B, all the privileges that B propagated should automatically be revoked by the DBMS

Dangers of Propagation

- A is the owner of relation R
- A grants B the DELETE privilege on R, with GRANT OPTION
- A grants C the DELETE privilege on R, also with GRANT OPTION
- B and C both grant D the DELETE privilege on R
- B later revokes the DELETE privilege from D
- However, D continues to have the DELETE privilege, as it was also granted from C

Example

- A DBA creates four user accounts
  - Tywin, Cersei, Jamie and Tyrion
- The DBA only wants Tywin to be able to create relations in the DB
  - GRANT CREATE TABLE TO Tywin;
- Tywin now has the ability to create tables
  - He does not have the ability to grant CREATE TABLE to other users

Example

- Tywin creates two tables:

Land

ID_num	Name	Kingdom_ID	Family_ID	Area	Value

Lords

Tax_num	Name	Address	Age	Role	Salary

- Tywin is the owner of these tables
- He automatically has all relation privileges on each of these tables

Example

- Tywin wants to grant Jamie the ability to insert, retrieve and delete rows in both of these tables
- However, he doesn't want Jamie to be able to pass this ability on to other users
- Tywin issues the following command:
  - GRANT INSERT, SELECT, DELETE ON Land, Lords TO Jamie;

Example

- Tywin wants to grant Cersei the ability to retrieve information from either of the tables
- He also trusts her to pass on this ability to other users of the database
- Tywin issues the following command:
  - GRANT SELECT ON Land, Lords TO Cersei WITH GRANT OPTION;

Example

- Cersei can now propagate this privilege to other user accounts using the GRANT command
- She wants to grant Tyrion the ability to retrieve information from the Lords table
- Cersei issues the following command:
  - GRANT SELECT ON Lords TO Tyrion;

Example

- Tywin decides to revoke the SELECT privilege on Lords from Cersei
- Tywin issues the following command:
  - REVOKE SELECT ON Lords FROM Cersei
- The DBMS must now automatically revoke the SELECT privilege from Tyrion as it was granted to him by Cersei, who no longer has the privilege

Example

- Tywin feels a bit bad, and wants to give Cersei back the ability to see the Lords information
- He also wants Cersei to be able to propagate this privilege again
- However, he only wants her to be able to see:
  - name, age and address
  - Lords who are now members of the Night's Watch
- How does he achieve this?

Example

- Tywin creates a View on the Staff table:

```
CREATE VIEW Lords_Restricted AS
SELECT Name, Age, Address
FROM Lords
WHERE Role = "Night's Watch";
```

- After the view is created, Tywin grants SELECT to Cersei as follows
  - GRANT SELECT ON Lords_Restricted TO Cersei WITH GRANT OPTION

Example

- Finally, Tywin wants to grant Jamie the ability to update the Salary field in the Staff table
- Tywin issues the following command:
  - GRANT UPDATE (Salary) ON Lords TO Jamie
- UPDATE and INSERT are examples of privileges can than be specified on attribute(s)
  - DELETE and SELECT are not attribute specific
    - That functionality is handled using Views

Mandatory Access Control

- Mandatory Access Control classifies data and users based upon *security levels*
  - can be combined with discretionary access control
  - desirable in government, military and intelligence
- Not commonly available in Commercial DBMS
  - Some companies, for instance Oracle, have released special versions of DBMS for government which include MAC

Mandatory Access Control

- Most simple example of security levels are:
  - Top Secret, Secret, Confidential, Unclassified
  - TS ≥ S ≥ C ≥ U
- Each *subject* and *object* are given a security level
  - Subject (User Account, Application Program…)
  - Object (Relation, Tuple, Attribute, View, Operation…)
- The security level of the subject is compared with that of the object
  - for the DBMS to decide if the action is permitted

Access Control Comparison

- Discretionary Access Control
  - Flexible
  - Complex to manage
  - Can be vulnerable to malicious attacks
- Mandatory Access Control
  - Rigid
  - Very secure
- Trade-off between Security and Applicability

Role-Based Access Control

- Privileges and other permissions are associated with organisational roles rather than individual user accounts
- Users are then assigned to appropriate roles
- Roles can be created in SQL using
  - CREATE ROLE
  - DESTROY ROLE

Role-Based Access Control

- GRANT and REVOKE are then used to allocate privileges to the created roles
- Users are allocated to roles
  - GRANT role TO user1
  - Multiple individuals can be assigned to each role
  - Any individual assigned to a role automatically has the privileges associated with that role
- An individual can be assigned to multiple roles

Summary

- Integrity and Security are related but they are not the same thing
  - Integrity is concerned with *accidental* corruption
  - Security is concerned with *deliberate* corruption
- Integrity
  - Integrity Constraints
- Security
  - Privilege Granting and Revocation
  - Security Level Assignment