Database Security and Privacy

Prof. K.R. Chowdhary, Director JIETCOE
Email: kr.chowdhary@jietjodhpur.ac.in

JIET COllege of Engineering

August 1, 2017

Introduction

A complete solution to either the security or the privacy problem requires the following three steps:

- Policy.
- Mechanism.
- Assurance.

Layers & types of information security

- Access control
- Auditing
- Authentication
- Encryption
- Integrity controls
- Backups
- Application security
- Database Security applying Statistical Method

Security related acitivties

- Privileges
  - System privilege
  - Object Privileges
- Abstraction
- Database activity monitoring (DAM)
- Native audit
- Process and procedures

Database Security and Auditing: Laboratory 1: Build a Database

- **create** a database scheme with given database design
- **create** primary and foreign keys for relations
- **instantiate** the database with instances
- **insert a new instance** to the created database
- **delete** or **update** an existing instance
- **manipulate** three options (RESTRICT, CASCADE, or SET NULL) in referential integrity

Database Security and Auditing: Laboratory 1: Build a Database

```
$ mysql -u root -p
mysql>
mysql> CREATE DATABASE books;
mysql> USE books;
mysql> CREATE TABLE authors (id INT, name VARCHAR(20),
    email VARCHAR(20));
mysql> SHOW TABLES;
mysql> INSERT INTO authors (id,name,email) VALUES(1,
   "Vivek","xuz@abc.com");
mysql> INSERT INTO authors (id,name,email)
    VALUES(2,"Priya","p@gmail.com");
mysql> INSERT INTO authors (id,name,email)
    VALUES(3,"Tom","tom@yahoo.com");
mysql> SELECT * FROM authors;
```

Database Security and Auditing: Laboratory 1: Build a Database

```
mysql> exit
mysql> DROP TABLE authors;
mysql> DROP DATABASE books;
```

Database Security and Auditing: Access Control

Security policies can be implemented through access control rules. Access control policies can be grouped into three major classes:

1. Discretionary access control (DAC),
2. Mandatory access control (MAC),
3. Role-based access control (RBAC).

Access Control

- DAC policies of a database system can be implemented by an access matrix model
- An object can be a table, a view, a procedure or any other database object.
- A subject can be a user, a role, a privilege, or a module.
- For instance, an owner can grant to others or revoke from others, a privilege to execute an action on her files.
- An access control list associates each object with a list of subjects and actions
- A capability list associates each user a list of objects and actions that the user is allowed to exercise on the objects.

Access Control

```
GRANT privilege_name
  ON object_name
  TO {user_name |PUBLIC |role_name}
  [WITH GRANT OPTION];

  GRANT SELECT ON employee TO user1;

  REVOKE privilege_name
    ON object_name
    FROM {user_name |PUBLIC |role_name}

 REVOKE SELECT ON employee FROM user1;
```

Privileges and Roles:

- **Privileges:** Privileges defines the access rights provided to a user on a database object. There are two types of privileges.
  1. *System privileges:* This allows the user to CREATE, ALTER, or DROP database objects.
  2. *Object privileges:* This allows the user to EXECUTE, SELECT, INSERT, UPDATE, or DELETE data from database objects to which the privileges apply.
- **Roles:** Roles are a collection of privileges or access rights.

Access Control ...

- SQL Server, MySQL, Oracle Database, DB2 and Sybase support the implementation of access matrix models.
- Role-based access control (RBAC) is an alternative to traditional DAC and MAC
- Role-based policies regulate the access of users to the information based on organizational responsibilities
- One vulnerability of DAC lies in the fact that there is no control on flow of information.

Laboratory 2: Implementing DAC

- This lab can be implemented in either Oracle 10g or Microsoft SQL Server. Both Oracle 10g and SQL server support concept of roles, as a result implementation of DAC can be extended to the implementation of Role-based Access Control (RBAC).
- Objective: To implement database security policies using discretionary access control (DAC). Results: You are able to:
- create users, roles, profiles, privileges.
- interpret given database security policies into an access control matrix.
- assign privileges based on users.
- assign privileges based on roles .
- understand potential vulnerabilities of DAC.

Creating Roles:

```
 CREATE ROLE role_name
[IDENTIFIED BY password];

CREATE ROLE testing
[IDENTIFIED BY pwd];

GRANT CREATE TABLE TO testing;

GRANT testing TO user1;

REVOKE CREATE TABLE FROM testing;

DROP ROLE role_name;

DROP ROLE testing;
```

Laboratory 2: Implementing MAC

- The most common form of MAC is the multilevel security policy using classification of subjects and objects in the system.
- The partial order is defined by a dominance relationship. An access class consists of two components: a security level and a set of categories.
- Most of database vendors can offer functions supporting label security through use of row-level security or fine-grained access control,

Laboratory 3 Countering Trojan Horse Using MAC

- interpret database security policies into MAC rules
- create security levels and labels
- bind security labels with objects
- bind users or roles with security levels
- students are able to justify that their implementation can
- counter Trojan Horse attacks

Application security

- Database applications are new major problem when increasing bugs arising from programming errors in applications.
- SQL injection is typical application attack as a result of insecure code.
- The malicious user can either gain access to information that he/she was not authorized, or delete or alter data in the back-end database.
- For example, a hacker can fill the username with user and password with guess'; delete from table users where username like '%. The database executes two SQL statements:

```
select user from users where username='user' and
password = 'guess';

delete from table users where username like '\%'
```