CHAPTER 2

DATABASE SECURITY AND ADMINISTRATION

Database Security

- The mechanisms that protect the database against intentional or accidental threats
- The important issues that database security must concern.
    - Confidentiality
    - Integrity
    - Availability

Threats

- A threat refers to an incident that has the potential to harm a system.
  - Natural threats, such as floods, hurricanes, or tornadoes
  - Unintentional threats, like an employee mistakenly accessing the wrong information
  - Intentional threats, such as spyware, malware, worms, viruses, or the actions of a disgruntled employee

Vulnerability

- A vulnerability refers to a known weakness of an asset (resource) that can be exploited by one or more attackers.
- It is a known issue that allows an attack to succeed.
  - For example, when a team member resigns and you forget to disable their access to external accounts, change logins, or remove their names.
- Vulnerabilities can be exploited by **automated attackers** and not a human typing on the other side of the network.

Risk

- Risk is defined as the potential for loss or damage when a threat exploits a vulnerability.
  - Examples of risk include financial losses, loss of privacy, reputational damage, legal implications, and even loss of life.
- Risk can also be defined as follows:

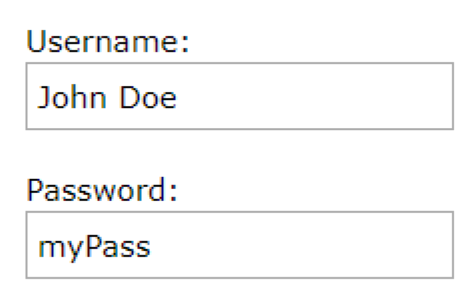
  Risk = Threat X Vulnerability

An example of threat, vulnerability, and risk

- The **threat** of a hurricane is outside of one’s control. However, knowing that a hurricane could strike can help business owners assess weak points and develop an action plan to minimize the impact.
- In this scenario, a **vulnerability** would be not having a data recovery plan in place in the event that your physical assets are damaged as a result of the hurricane.
- The **risk** to your business would be the loss of information or a disruption in business as a result of not addressing your vulnerabilities.

Examples threats to database

- SQL injections

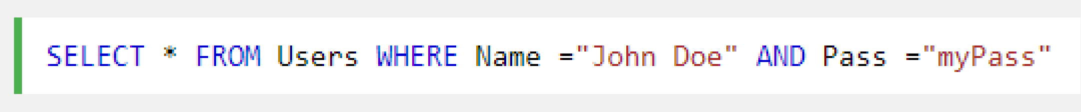


Username:

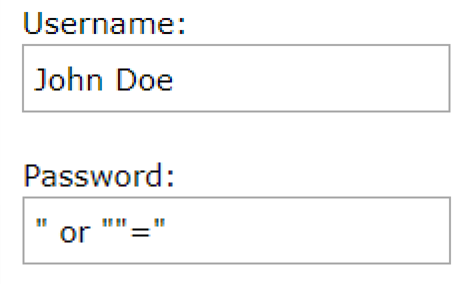
John Doe

Password:

myPass



```sql
SELECT * FROM Users WHERE Name ="John Doe" AND Pass ="myPass"
```

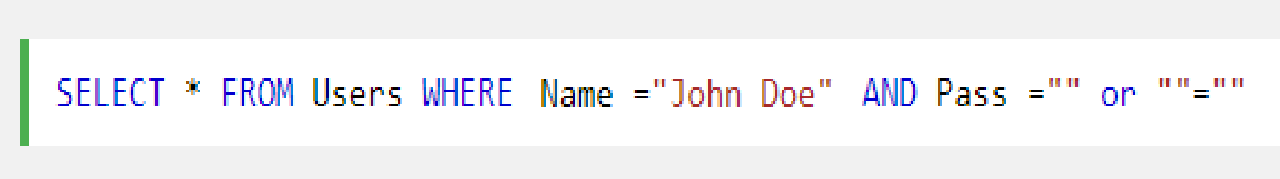


Username:

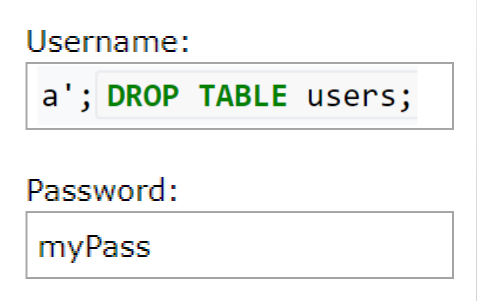
John Doe

Password:

" or ""="



```sql
SELECT * FROM Users WHERE Name ="John Doe" AND Pass ="" or ""=""
```

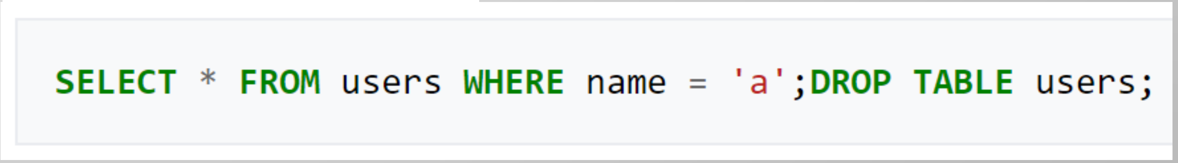


Username:

a'; DROP TABLE users;

Password:

myPass



```sql
SELECT * FROM users WHERE name = 'a';DROP TABLE users;
```

Examples threats to database

- Denial of service attack.
  - DoS attack slows down a database server and can even make it unavailable to all users.
  - DoS attack doesn’t disclose the contents of a database, it may cost the victims a lot of time and money.
- Countermeasures:
  - Decrease the connection establishment period.
  - Use a network Intrusion Detection System (IDS).

Examples vulnerability to database

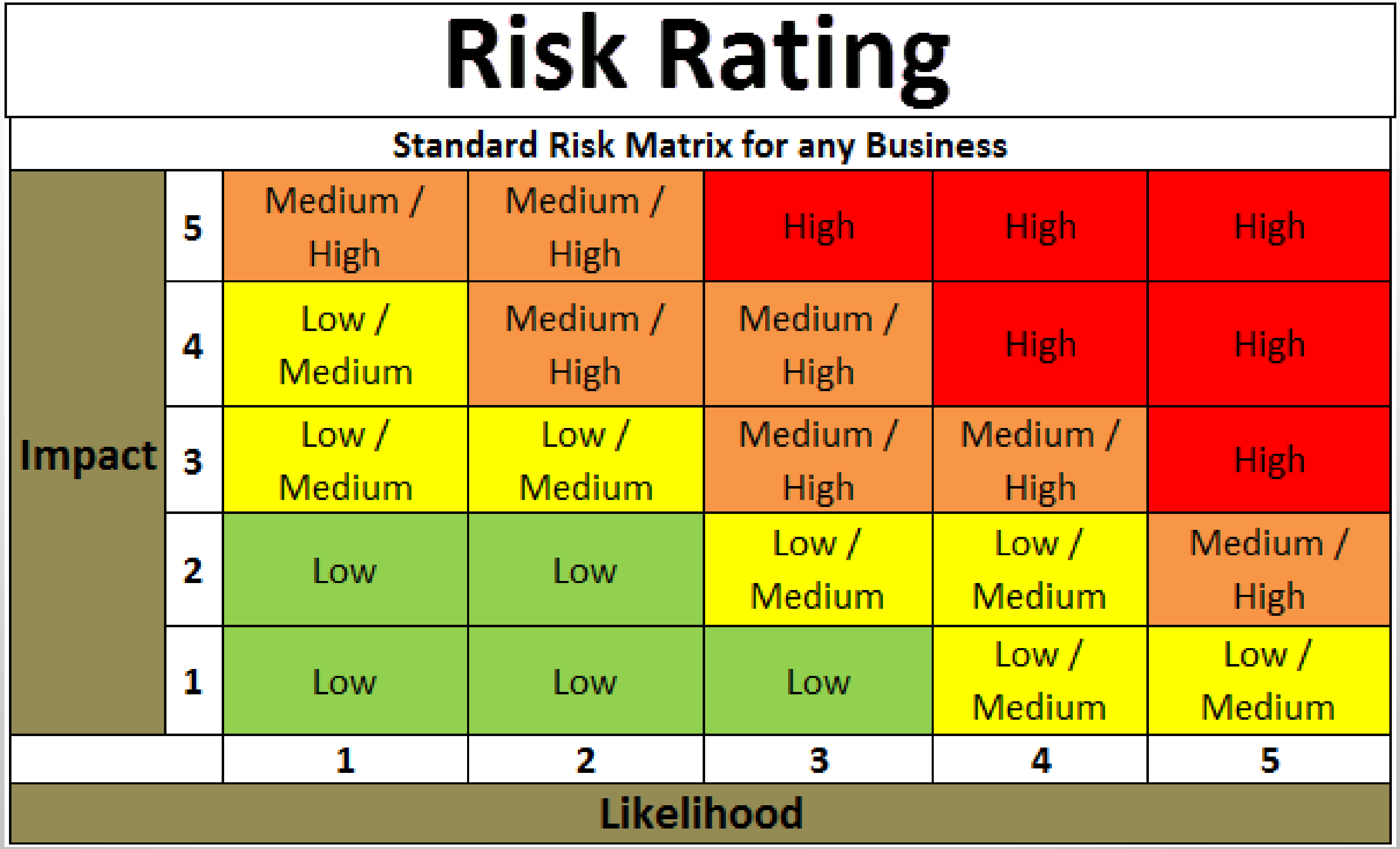
- Excessive Database Privileges.
    - Deploy a strict access and privileges control policy.
    - Don’t grant excessive privileges to company employees
    - Revoke outdated privileges in time.

Effects of threats & vulnerabilities

Threats	Loss of confidentiality	Loss of integrity	Loss of availability
SQL injection			
DoS attack			
Fire, flood, bomb			
Wire tapping			

Vulnerabilities	Loss of confidentiality	Loss of integrity	Loss of availability
Excessive Database Privileges			
Inadequate staff training			
Lack of appropriate policies			
Using outdated software			

Risk assessment



Risk Rating

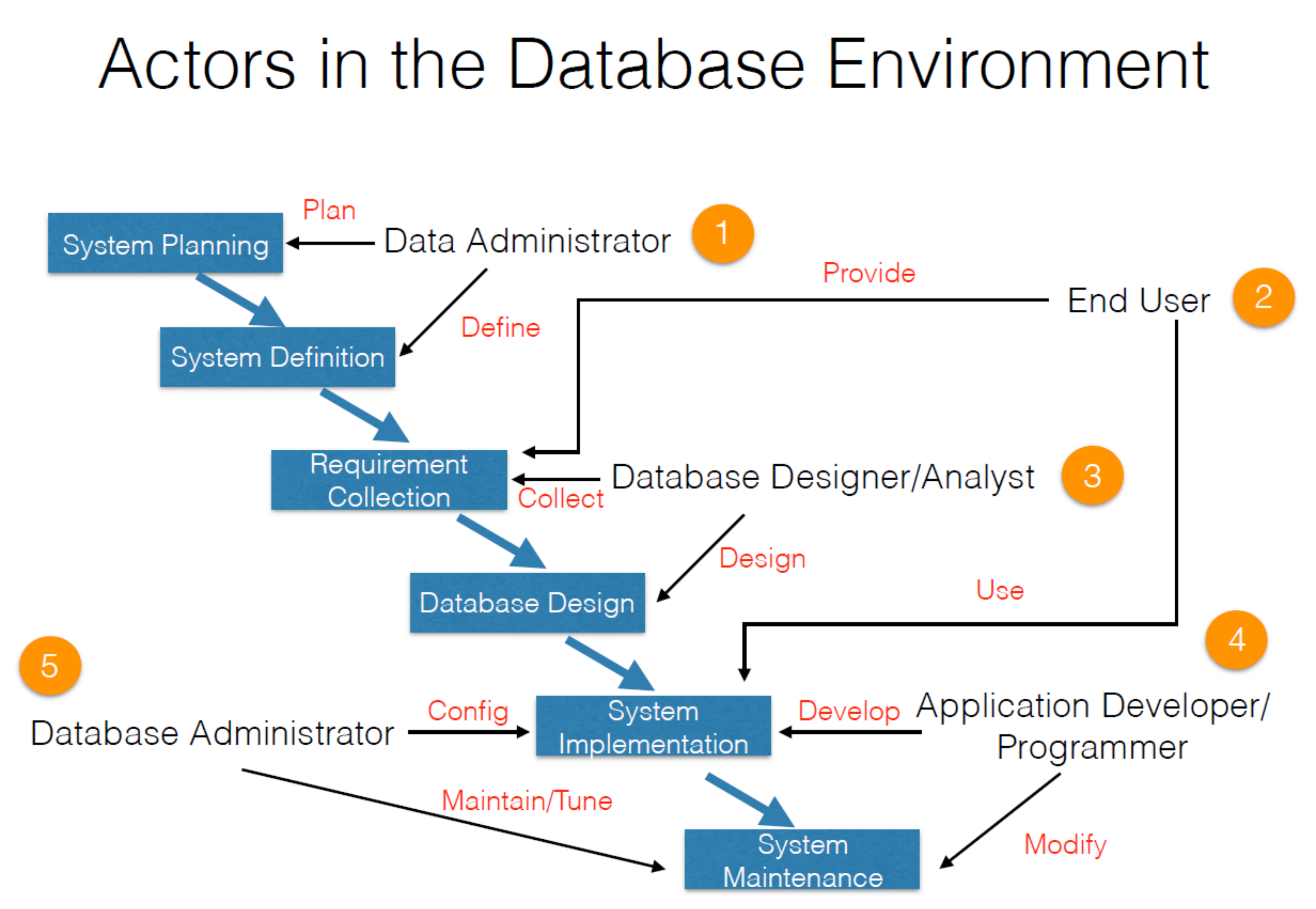
Standard Risk Matrix for any Business

Impact	5	Medium / High	Medium / High	High	High	High
	4	Low / Medium	Medium / High	Medium / High	High	High
	3	Low / Medium	Low / Medium	Medium / High	Medium / High	High
	2	Low	Low	Low / Medium	Low / Medium	Medium / High
	1	Low	Low	Low	Low / Medium	Low / Medium
		1	2	3	4	5
Likelihood						

Database’s components in security

- The components of database involving in the database security are
  - Data
  - DBMS
  - Software
  - Hardware
  - People

INT203 – Database actors



Actors in the Database Environment

System Planning

System Definition

Requirement Collection

Database Design

System Implementation

System Maintenance

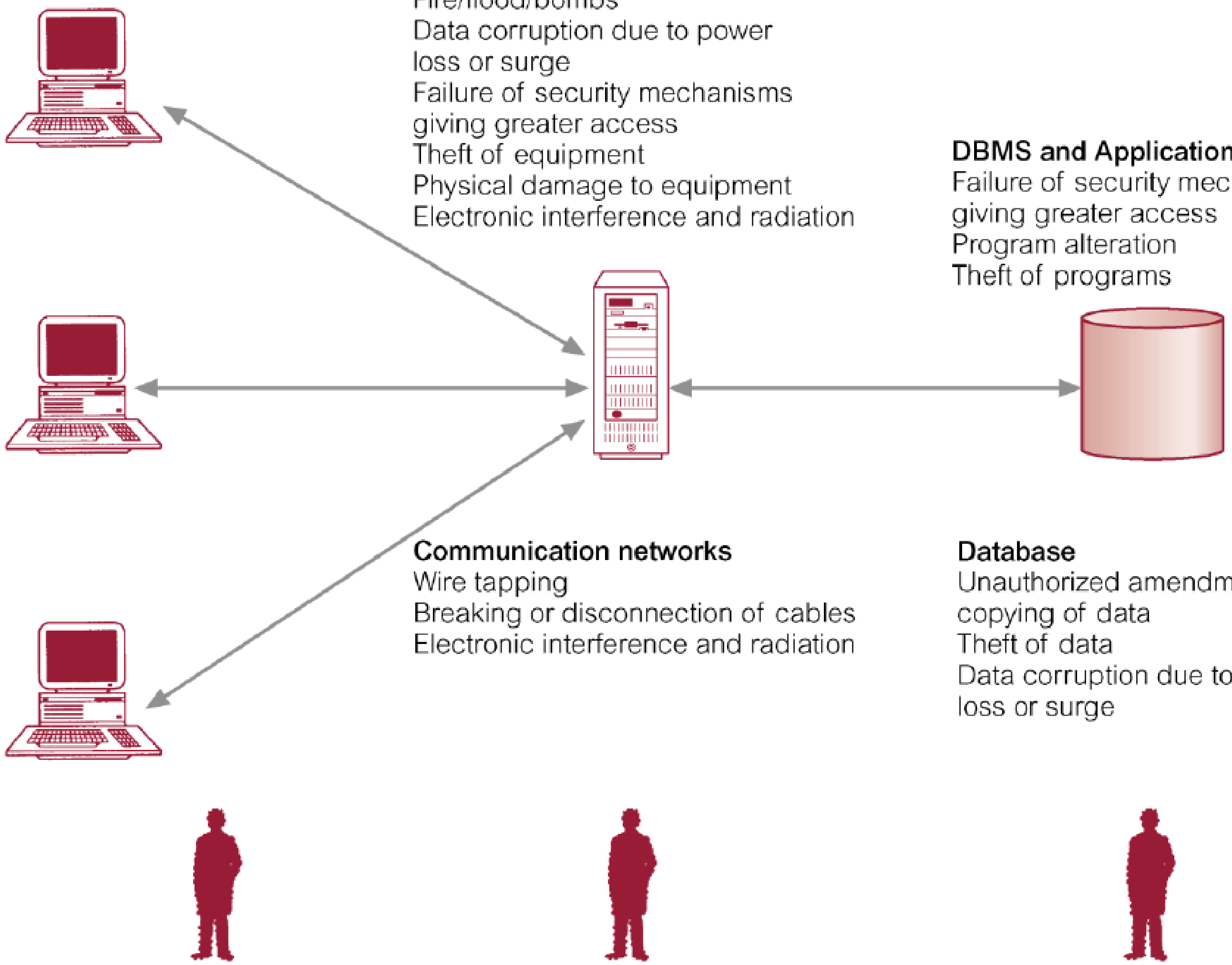
1 Data Administrator: Plan, Define

2 End User: Provide, Use

3 Database Designer/Analyst: Collect, Design

4 Application Developer/ Programmer: Develop, Modify

5 Database Administrator: Config, Maintain/Tune



Hardware
Fire/flood/bombs
Data corruption due to power loss or surge
Failure of security mechanisms giving greater access
Theft of equipment
Physical damage to equipment
Electronic interference and radiation

DBMS and Application Software
Failure of security mechanism giving greater access
Program alteration
Theft of programs

Communication networks
Wire tapping
Breaking or disconnection of cables
Electronic interference and radiation

Database
Unauthorized amendment or copying of data
Theft of data
Data corruption due to power loss or surge

Users
Using another person's means of access
Viewing and disclosing unauthorized data
Inadequate Staff training
Illegal entry by hacker
Blackmail
Introduction of viruses

Programers/Operators
Creating trapdoors
Program alteration (such as creating software that is insecure)
Inadequate staff training
Inadequate security policies and procedures
Staff shortages or strikes

Data/Database Administrator
Inadequate security policies and procedures

Countermeasure

- Authentication
- Authorization
- Creating views
- Encryption
- Backup and recovery
- RAID technology
- Auditing

Authentication

- Authentication is the process of verifying that whether someone/something is who/what it declare to be.
- A common example is entering a username and password when you log in to an OS or website.

Authorization

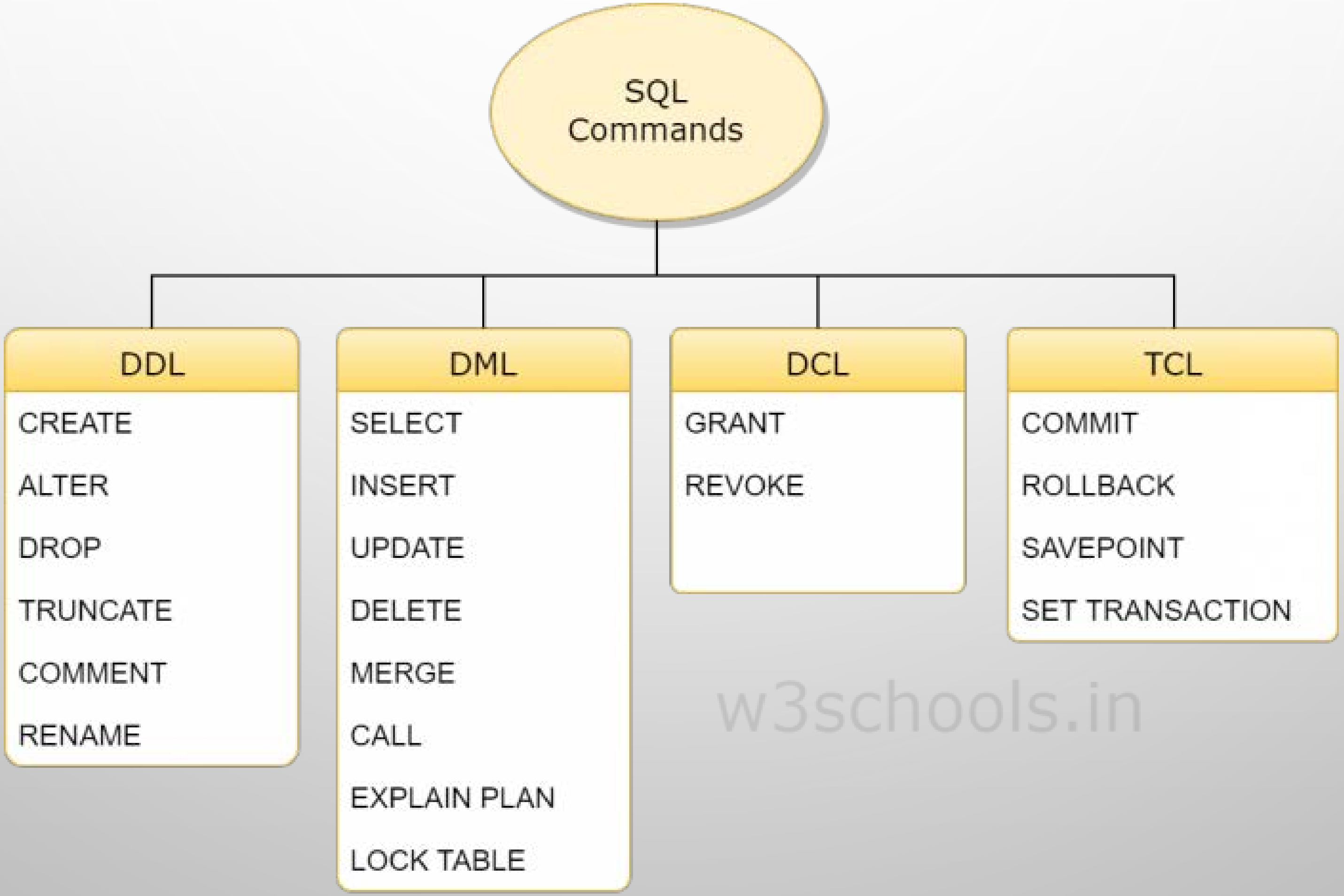
- Authorization is the function of specifying access rights/**privileges** to resources or system.
- Example
  - Most web security systems are based on a two-step process. The first step is authentication, which ensures about the user identity and the second stage is authorization, which allows the user to access the various resources based on the user's identity.

Privileges in database

- Privileges is an authority level used to access the system or database’s objects, to manipulate data, and to perform various administrative functions.
- A user who creates a database object (relation or view) automatically gets all privileges on that object.
- Users can grant they own object to others.

Privileges in database

- SQL commands related to privileges



SQL Commands

DDL	DML	DCL	TCL
CREATE	SELECT	GRANT	COMMIT
ALTER	INSERT	REVOKE	ROLLBACK
DROP	UPDATE		SAVEPOINT
TRUNCATE	DELETE		SET TRANSACTION
COMMENT	MERGE		
RENAME	CALL		
	EXPLAIN PLAN		
	LOCK TABLE		

Privileges in database

- Discretionary access control (DAC) is a means of restricting access to objects **based on the identity of users**. User with a certain access permission is capable of passing that permission (perhaps indirectly) on to any user.
- Mandatory access control (MAC) is based on **security labels**. Users are given a security clearance, and data objects are given a security classification. The clearance and classification data are stored in the security labels, which are bound to the specific users and objects.

Privileges in Oracle

- System privilege has the right to perform a particular action or perform an action on any schema objects of a particular type.
- Object privilege has the right to perform a particular action on a specific table, view, sequence, procedure, function or package.