AN TOÀN & BẢO MẬT DỮ LIỆU

Nội dung

- An toàn dữ liệu
- Bảo mật dữ liệu
- Ngôn ngữ điều khiển truy cập dữ liệu (DCL)

An toàn dữ liệu

- Giới thiệu
- Các loại sự cố và cách giải quyết

Giới thiệu

- Cơ sở dữ liệu luôn cần phải ở trạng thái nhất quán, nghĩa là phải đảm bảo tất cả ràng buộc toàn vẹn (RBTV)
- Các sự cố dẫn đến RBTV bị vi phạm như:
  - Sự cố do nhập liệu sai – Errornous Data Entry
  - Sự cố trên thiết bị lưu trữ – Media failures
  - Sự cố giao tác – Transaction failures
  - Sự cố hệ thống – System failures

Sự cố do nhập liệu sai

- Bao gồm:
  - *Dữ liệu sai hiển nhiên*: Là sự nhập sai dữ liệu mà máy tính có thể phát hiện được
    - Vd : Nhập thiếu 1 số trong dãy số điện thoại, nhập sai khóa ngoại, nhập chuỗi tràn, sai kiểu dữ liệu…
  - *Dữ liệu sai không hiển nhiên*: Là sự nhập sai dữ liệu liên quan đến ngữ nghĩa mà máy tính khó có thể tự nó phát hiện được
    - Vd : Nhập sai 1 số trong dãy số điện thoại
- Giải quyết: Hệ quản trị CSDL cung cấp các cơ chế cho phép phát hiện lỗi
  - Ràng buộc khóa chính, khóa ngoại
  - Ràng buộc miền giá trị
  - Trigger

Sự cố trên thiết bị lưu trữ

- Là những sự cố:
  - Là những sự cố gây nên việc mất hay không thể truy xuất dữ liệu ở bộ nhớ ngoài (ổ cứng, CD, băng từ…)
    - Vd : Cháy nổ gây phá hủy thiết bị lưu trữ,…
    - Vd : Đầu đọc của đĩa cứng hư, sector trên đĩa cứng hư, …
  - Đây là loại sự cố nguy hiểm nhất, khó khôi phục trọn vẹn
- Giải quyết:
  - Phải backup thường xuyên (toàn bộ hoặc chỉ phần thay đổi), chu kỳ không được quá thưa
  - Chạy nhiều bản CSDL song hành (1 bản chính – primary và nhiều bản phụ – minor) và thực hiện đồng bộ tức thì
    - Tốn thiết bị lưu trữ và đòi hỏi phần cứng rất mạnh
    - Kìm hãm tốc độ hệ thống
    - Bản minor phải đặt ở vị trí địa lý khác bản primary

Sự cố giao tác

- Sự cố làm cho 1 giao tác kết thúc không bình thường (không đến được lệnh commit hay lệnh rollback của chính nó)
- Ví dụ
  - Chia cho không
  - Giao tác bị hủy
  - Dữ liệu nhập sai
  - Tràn số
- Giải quyết : Khi giao tác T bị sự cố, DBMS sẽ
  - Hủy T và các giao tác bị quay lui dây chuyền theo nó
  - Tra lock-table và giải phóng các khóa mà các giao tác này đang giữ
  - Reset lại các giá trị mà các giao tác này đã ghi
  - Thực hiện lại tất cả các giao tác này

Sự cố hệ thống

- Là những sự cố gây nên bởi
  - Lỗi phần cứng
    - Cúp điện
    - Hư bộ nhớ trong
    - Hư CPU
    - ...
  - Lỗi phần mềm
    - Lỗi hệ điều hành
    - Lỗi DBMS
    - ...
- Giải quyết : Hệ quản trị CSDL cần cứu chữa và phục hồi dữ liệu
  - Nhật ký giao tác (transaction log)

Mục tiêu của khôi phục sự cố

- Đưa dữ liệu về trạng thái sau cùng nhất trước khi xảy ra sự cố
- Đảm bảo 2 tính chất của giao tác:
  - Nguyên tố (atomic)
  - Bền vững (durability)

Bảo mật dữ liệu

Bảo mật dữ liệu

- Bài toán phân quyền
  - Quản lý tốt việc truy xuất Dữ liệu của các đối tượng người dùng hợp pháp → Bảo mật Dữ liệu
  - Thông qua 2 cơ chế
    - Cơ chế chứng thực
    - Cơ chế phân quyền
      - » Quan điểm phân quyền cụ thể
      - » Quan điểm phân cấp mức độ MẬT
- Bài toán mã hóa
  - Ngăn chặm hiệu quả sự tấn công, xâm nhập của các đối tượng tin tặc → An ninh Dữ liệu

Cơ chế chứng thực (Authentication)

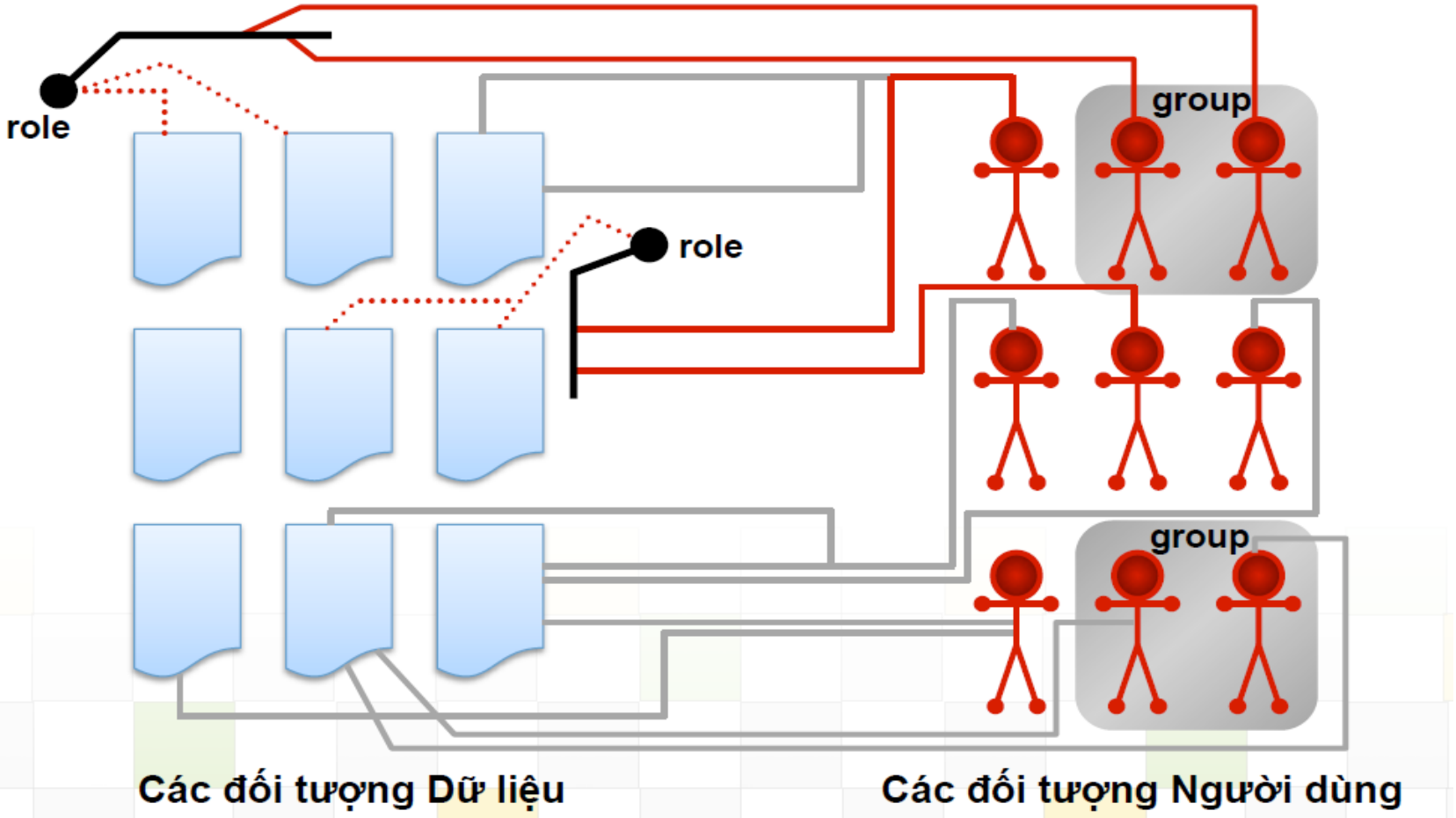
- Mỗi người dùng DBMS được xác định bởi
  - Một tên đăng nhập – user name
  - Một mật mã đăng nhập – password
- Thông tin về user name và password
  - Không được lưu trữ tường minh trong dữ liệu
  - User name và password của DBMS và của OS có thể tách bạch nhau hay dùng chung cho nhau là tùy hệ thống
    - Vd : Mixed-mode của Microsoft SQL Server

Cơ chế phân quyền

- Một tài khoản chứng thực
  - Được phép đăng nhập vào hệ thống DBMS
  - Được nhìn thấy các CSDL
  - Chưa được phép truy xuất các đối tượng trong các CSDL
- Tài khoản chứng thực muốn truy xuất các đối tượng dữ liệu thì cần được phân quyền cụ thể chi tiết trên các đối tượng dữ liệu đó

Cơ chế phân quyền

- Quan điểm phân quyền cụ thể



Cơ chế phân quyền

- Quan điểm phân cấp mức độ MẬT
  - Các đối tượng Dữ liệu được phân ra các cấp độ bảo mật khác nhau
    - Vd :
      - Cấp 3 : Dành cho tài liệu tuyệt mật
      - Cấp 2 : Dành cho tài liệu mật
      - Cấp 1 : Dành cho tài liệu công khai
  - Các đối tượng Người dùng cũng được phân ra các cấp độ bảo mật khác nhau
    - Vd :
      - Cấp 3 : Dành cho ban giám đốc
      - Cấp 2 : Dành cho các trưởng phòng
      - Cấp 1 : Dành cho nhân viên
  - Khó khăn : Làm sao phân cấp cho hợp lý (♣)

Cơ chế phân quyền

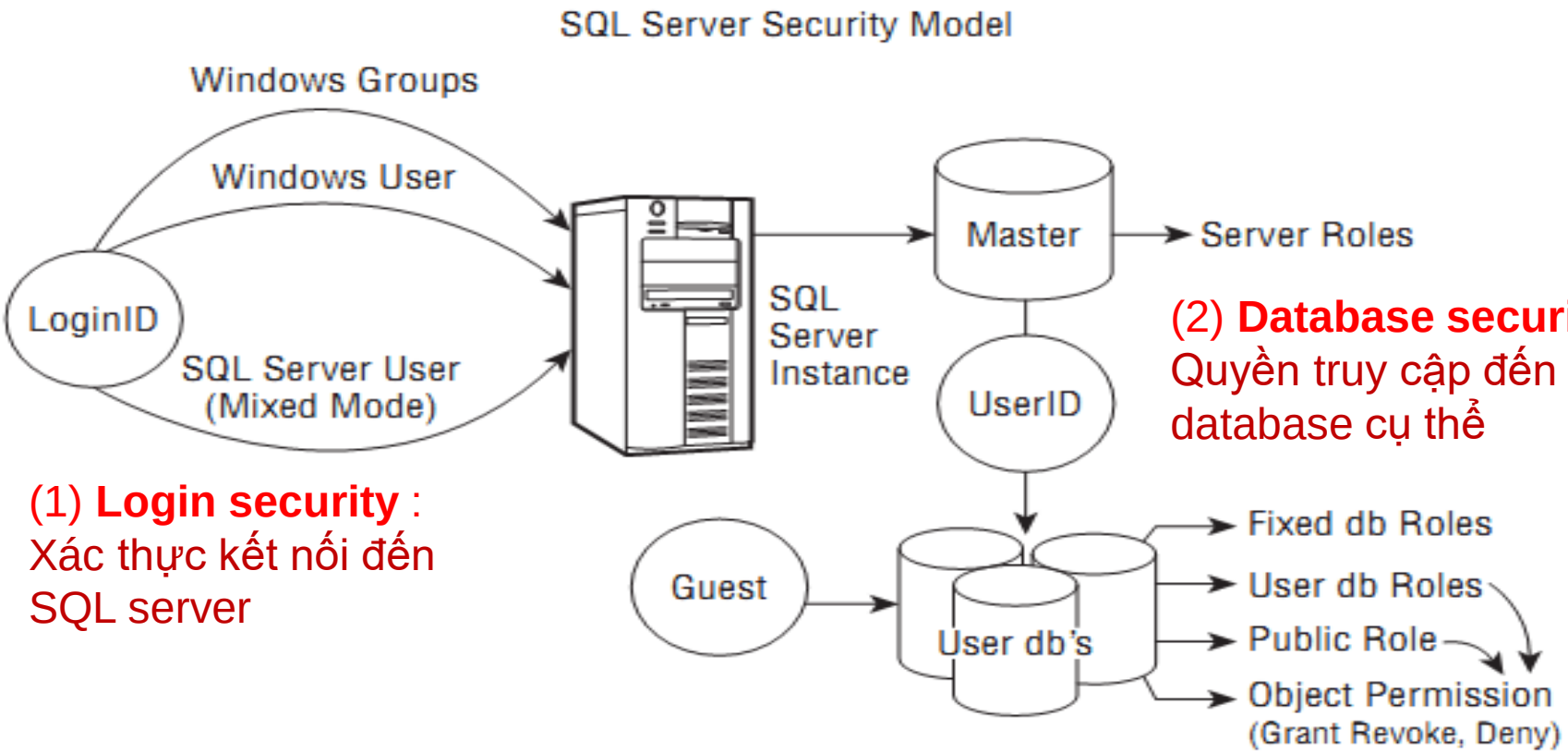
- Quan điểm phân cấp mức độ MẬT
  - Phân quyền
    - Quyền đọc dữ liệu : Người dùng cấp i được đọc các tài liệu cấp i trở xuống
    - Quyền ghi dữ liệu : (♣♣)
      - Ban giám đốc đọc các tài liệu mật nhưng tài liệu ấy không nhất định do họ tạo ra, thông thường lại do nhân viên tạo ra
      - Người dùng cấp i được ghi tài liệu cấp i trở xuống
      - Nếu người dùng X thuộc cấp i tạo ra tài liệu A thuộc cấp j (với j > i) thì chi có X được đọc A trong khi các X' cùng cấp không được đọc A
  - Vì (♣) và (♣♣♣) nên quan điểm này gặp nhiều thách thức và ít được ứng dụng trong các DBMS thương mại

Cơ chế mã hóa

- Bất chấp cơ chế phân quyền, nhiều đối tượng người dùng bất hợp pháp vẫn có thể xâm nhập vào CSDL
  - Ví dụ :
    - Thâm nhập từ mức Hệ điều hành để chép các file dữ liệu của DBMS (như file *.mdf và *.ndf của SQL Server)
    - Chặn trên đường truyền mạng để hứng lấy dữ liệu luân chuyển giữa Client và Server
- Giải pháp : Mã hóa thông tin trước lưu trữ hoặc truyền trên đường truyền
  - Tin tặc lấy được file hay dữ liệu cũng không hiểu được
  - Việc mã hóa không được xung đột với hệ thống index → thách thức
  - Thuật toán mã hóa được chọn sao cho việc giải mã của tin tặc là khó khăn nhất

Ngôn ngữ điều khiển truy cập dữ liệu (DCL)

Mô hình truy cập bảo mật của SQL Server



(1) **Login security** :
Xác thực kết nối đến SQL server

(2) **Database security**:
Quyền truy cập đến database cụ thể

(3) **Database objects security**:
Quyền truy cập đến các đối tượng của database

(1) Login security

- Hai loại chứng thực (Authentication):
  - **Windows Authentication**:
    - User chỉ cần được cấp account trong Windows. SQL Server sẽ dựa vào Windows để chứng thực cho user.
  - **SQL server Authentication**:
    - Người quản trị CSDL tạo ra tài khoản và password đăng nhập của SQL server.

Tạo SQL Server login account

CREATE LOGIN <tên đăng nhập>
WITH PASSWORD = ‘mật khẩu’

- Ví dụ:

```
CREATE LOGIN test
WITH PASSWORD = '123456'
GO
```

- Xem thông tin login: **Sp_helplogins** ['login']

```
Sp_helplogins test
```

- Xóa login:

```
DROP LOGIN test
```

- Login được tạo chỉ mới có quyền login vào SQL server

→Để cấp quyền truy cập vào database thì phải cấp cho login một user (thường tên database user trùng với tên login)

(2) Database access security

- Tạo một user:

CREATE USER user_name [FOR LOGIN login_name]

- Nếu không có FOR LOGIN, thì **user** mới tạo ra sẽ kết hợp với **login** của SQL Server cùng tên.
- Ví dụ:

```
CREATE USER Test FOR LOGIN Test
```

- Xem thông tin login:

```
Sp_helplogins ' Test '
```

Phân biệt login và user

- Login:

  - Là tài khoản mà người sử dụng dùng để kết nối với SQL Server

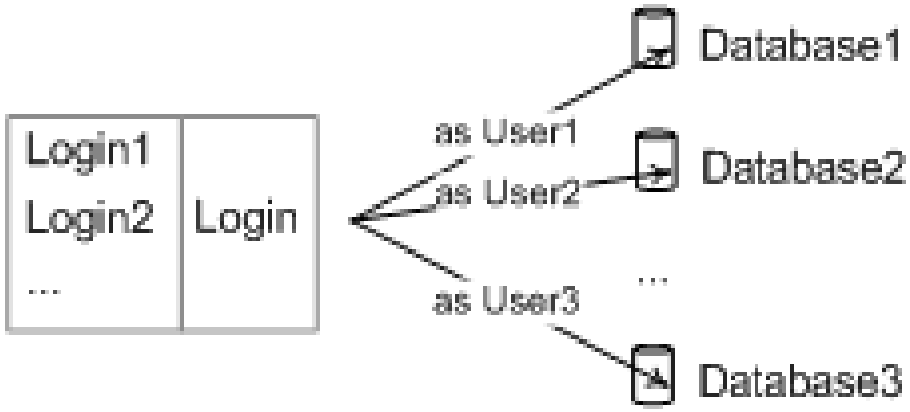
  - Một login có thể có quyền truy cập 0-n database

  - Trong mỗi database, một login sẽ ứng với một user

- User:

  - Một "người dùng" trong một **database cụ thể**

  - Một user ứng với một login



(3) Database objects security

User vừa được tạo chỉ mới có quyền access vào database, để User có quyền (select/insert/update/execute, …) trên đối tượng nào (table, view, SP, UDF, …) thì người quản trị cần cấp quyền trên đối tượng đó (Permission security)

Permission security

- Có hai loại quyền:
  - Statement permission
  - Object permission

Statement permission

- Cho phép một User/role có thể thực hiện các lệnh sau đây:
  - CREATE DATABASE
  - CREATE DEFAULT
  - CREATE PROCEDURE
  - CREATE RULE
  - CREATE TABLE
  - CREATE VIEW
  - BACKUP DATABASE
  - BACKUP LOG

Object permission

Cho phép một User/role có thể thực hiện các lệnh trên một object cụ thể trong database.

lệnh T-SQL	Đối tượng
DELETE	table , view
EXECUTE	stored procedure
INSERT	table , view
SELECT	table, view, và column
UPDATE	table, view, và column

Các thao tác về quyền

- Gán quyền (Grant)
- Từ chối (Deny)
- Hủy (revoke)

Gán quyền (Grant)

- Granting **Statement Permission** :
  - GRANT { ALL | statement [,...n] } TO user_name[,...n]

```
GRANT CREATE DATABASE, CREATE TABLE TO tam
```

- Granting **Object Permission**
  - GRANT { { ALL | permission [,...n] } [(column_name [,...n])] ON { table|view | stored_procedure |extended_procedure | user_defined_function }} TO user_name [,...n]

```
GRANT  SELECT, UPDATE
ON Employee (emp_fname, emp_lname)
TO Tam
```

Từ chối (Deny)

Ngăn User sử dụng quyền và không cho phép User có cơ hội thừa hưởng do là thành viên của một Role.

- Denying **Statement Permission** :

  DENY { ALL | statement [,...n] } TO user_name [,...n]

```
DENY CREATE DATABASE TO Tam
```

- Denying **Object Permission** :

  DENY { { ALL | permission [,...n] } [(column_name[,...n])]
  ON { table|view|stored_procedure |extended_procedure| user_defined_function }}
  TO user_name [,...n]

```
DENY UPDATE, DELETE
ON Employee
TO Tam
```

Hủy(revoke)

Hủy quyền đã cấp **grant** hay từ chối **deny**

- **Revoking Statement Permission :**

  REVOKE { ALL | statement [,...n] } FROM user_name [,...n]

```
REVOKE CREATE DATABASE, CREATE TABLE
FROM tam
```

- **Revoking Object Permission :**

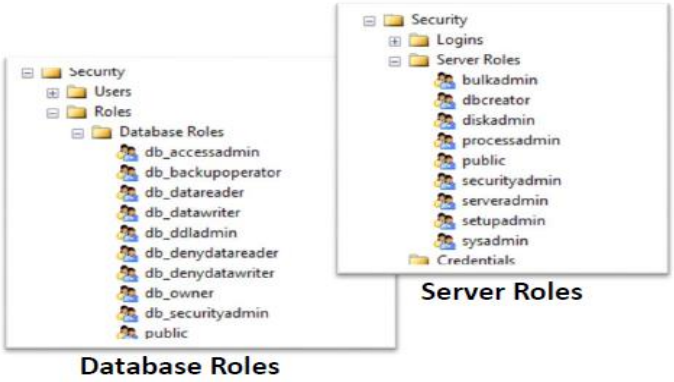
  REVOKE { { ALL | permission [,...n] } [(column_name [,...n])]
  ON{ table | view | stored_procedure |extended_procedure | user_defined_function }}
  {TO | FROM} user_name [,...n]

```
REVOKE UPDATE, DELETE
ON Employee
FROM Tam
```

object and privilege

Object	Privileges	Security commands
SQL server	- SQL server Login - Create/drop database	(1) Create login <loginID> (2) Drop login <loginID> (3) USE master CREATE USER <userID> GRANT CREATE DATABASE TO <userID>
Database	- DB user - BACKUP DATABASE, BACKUP LOG, CREATE TABLE, CREATE FUNCTION, CREATE VIEW, CREATE RULE	Use <dbname> (1) Create user <userID> [for <loginID>] (2) Drop user <userID> (3) Grant/Revoke/Deny
Table, View	SELECT, INSERT, DELETE, UPDATE	Grant /Revoke/Deny
SP, UDF	EXECUTE	

ROLE



Role là một công cụ cho phép cấp quyền cho một nhóm User thay vì thực hiện trên từng user.

Có 2 loại Roles:

- Fixed role: gồm Server role và Database role, do SQL Server tạo sẵn, chúng ta không thể xoá, sửa các role này
- User-defined Database role

Hoặc có thể phân biệt:

- Role mức server: fixed server role do hệ thống tạo sẵn
- Role mức Database: có 2 loại fixed database role, và user defined database role. Mỗi role có phạm vi trong một database

SQL server Roles

- Dùng để gán một nhóm quyền về quản trị server cho một login account

Server Roles	**Thành viên của Server Role có thể …**
sysadmin	Thực thi bất cứ thao tác nào trong một thể hiện SQL Server, thể hiện là dbo của mọi CSDL
serveradmin	Cấu hình SQL Server bằng cách dùng thủ tục hệ thống sp_configure và có thể kết thúc các sevice. Các thành viên của nhóm điều hành viên built-in của Windows là rất tốt để nhận server role này.
setupadmin	Cài đặt và cấu hình linked server, remote server, và replication. Có thể chỉ định một stored procedure được thực thi lúc khởi động (startup), như là sp_serveroption.

SQL server Roles

securityadmin	Thực hiện tất cả các thao tác liên quan đến security trong SQL Server 2008, kể cả quản lý các quyền câu lệnh CREATE DATABASE, điều khiển server logins, và đọc error log.
processadmin	Quản lý các tiến trình chạy các instance của SQL Server. Có thể ngắt (kill) tiến trình của các user, các truy vấn.
dbcreator	Có thể tạo, hiệu chỉnh, và xóa các CSDL.
Diskadmin	Có thể quản trị các tập và các thiết bị dự phòng. Nói chung Role này dùng để tương thích ngược với SQL Server 6.x.
bulkadmin	Có thể thực hiện các câu lệnh BULK INSERT. Cho phép các thành viên của sysadmin server role làm đại diện các tác vụ BULK INSERT mà không cần gán các quyền sysadmin

Gán một login cho server roles

sp_addsrvrolemember [@loginame=]'login',[rolename=] 'role'

- login: là ID đăng nhập vào SQL server
- role : tên server role gán cho login

- Ví dụ:

```
sp_addsrvrolemember 'test','sysadmin'
```

- Lưu ý: Khi mới cài đặt, SQL Server định nghĩa sẵn login sa và các login là administrator của Windows đều là thành viên của SysAdmin.

Xem thông tin về server roles

- sp_helpsrvrolemember [[@srvrolename =] 'role']
  cho xem các thành viên của 1 fixed server role nào đó.
  sp_helpsrvrolemember @srvrolename = 'sysadmin'
- sp_srvrolepermission [[@srvrolename =] 'role']
  cho xem quyền (permission) của 1 fixed server role

sp_srvrolepermission @srvrolename = 'sysadmin'

Database Roles

- Fixed Database Roles
- User Defined Roles

Fixed Database Roles

- Dùng gán một nhóm quyền về quản trị database cho một login account/user.

```
Use <dbname>
sp_addrolemember [@rolename =] 'role', [@membername =] ‘userID‘
```

Database Role	Thành viên của CSDL này có thể …
db_owner	Thực hiện bất kỳ tác vụ trong CSDL của SQL Server 2008.
db_accessadmin	Thêm hay xóa các user và group của Windows 2008 hoặc Win NT và các user trong một CSDL.
db_securityadmin	Kiểm soát tất cả các quyền truy cập, role, thành viên và chủ đối tượng trong CSDL

Database Role	Thành viên của CSDL này có thể …
db_ddladmin	Thêm, hiệu chỉnh, xóa các đối tượng trong CSDL (sử dụng lệnh CREATE, ALTER, và DROP).
db_backupoperator	Chạy các lệnh DBCC, phát hành checkpoint, và dự phòng CSDL (sử dụng các câu lệnh T-SQL: DBCC, CHECKPOINT, và BACKUP).
db_datareader	Đọc dữ liệu từ bất kỳ các bảng hoặc view của người dùng trong CSDL (bạn có quyền SELECT đối với tất cả table và view).
Db_datawriter	có quyền INSERT, UPDATE và DELETE đối với tất cả các table và view.
Db_denydatareader	Không đọc bất cứ dữ liệu nào trong CSDL, Không có quyền SELECT đối với bất kỳ đối tượng nào
Db_denydatawrite	Không hiệu chỉnh bất cứ dữ liệu nào trong CSDL bằng lệnh INSERT, UPDATE, và DELETE

User đặc biệt dbo

- dbo là một user có tất cả các quyền trên DB .
- Mặc định tất cả các thành viên thuộc sysadmin server role đều là dbo của tất cả các DB trên server.
- Một login account là thành viên của db_owner database role thì cũng có quyền như dbo.
- Khi thao tác các object trong DB, các thành viên này có thể dùng tên dbo để chỉ owner thay cho user name.

User Defined Roles- Bước 1

- Định nghĩa một role

  CREATE ROLE role_name [AUTHORIZATION owner_name]

  - **role_name**: tên role mới tạo
  - **owner_name**: là tên user hay role là thành viên của role mới tạo, nếu để trống mục này, role mới tạo sẽ sở hữu bởi user thực thi lệnh CREATE ROLE

  - Ví dụ:

```
CREATE ROLE quanly AUTHORIZATION test
```

User Defined Roles- Bước 2

- Gán quyền cho Role sau khi tạo ra cần phải cấp quyền thao tác trên các đối tượng trong CSDL (đọc/ ghi trên table/view, thực hiện thủ tục,…)

  Grant <privileges> to [role]

- Ví dụ:

```
GRANT SELECT, INSERT, UPDATE
ON DBO.EMPLOYEE
TO QUANLY
GO
```

User Defined Roles- Bước 3

- Gán các user là thành viên của role
- Ví dụ: gán user tam là thành viên của role quanly.

```
sp_addRoleMember 'quanly', 'tam'
```