

Chương 6

Bảo mật & phân quyền

TẦM QUAN TRỌNG CỦA BẢO MẬT

❑ **Mục đích:** bảo vệ dữ liệu

- Bảo vệ tính toàn vẹn dữ liệu
- Khắc phục các sự cố xảy ra với CSDL
- Chống lại các truy cập trái phép

❑ **Vai trò của người quản trị**

- Lập kế hoạch sao lưu khắc phục sự cố
- Tạo lịch sao lưu tự động
- Tạo tài khoản & phân quyền người dùng

CÁC KHÁI NIỆM CƠ BẢN

- ❑ **Database user:** đối tượng sử dụng cơ sở dữ liệu
 - Mỗi người dùng được xác định bởi UserID.
 - Người dùng có thể được tổ chức thành nhóm gọi là User Group.
 - Chính sách bảo mật được áp dụng cho một người hoặc cho nhóm người dùng

CÁC KHÁI NIỆM CƠ BẢN

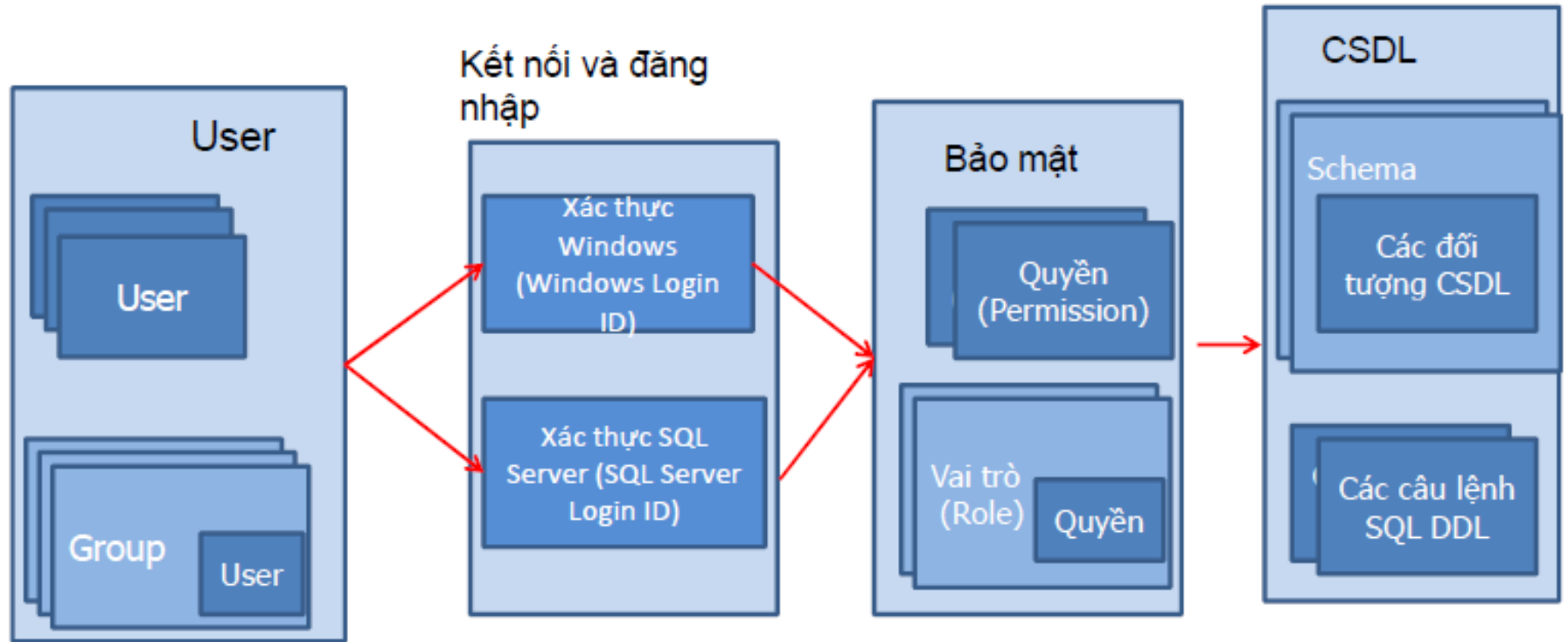
- ❑ **Database objects:** Tập các đối tượng, các cấu trúc lưu trữ được sử dụng trong cơ sở dữ liệu như Table, View, Procedure, Function.
- ❑ **Privileges:** Quyền thực hiện những thao tác được cấp phát cho người dùng trên các đối tượng cơ sở dữ liệu.

BẢO MẬT TRONG SQL SERVER

❑ Bảo mật trong SQL Server gồm 3 lớp:

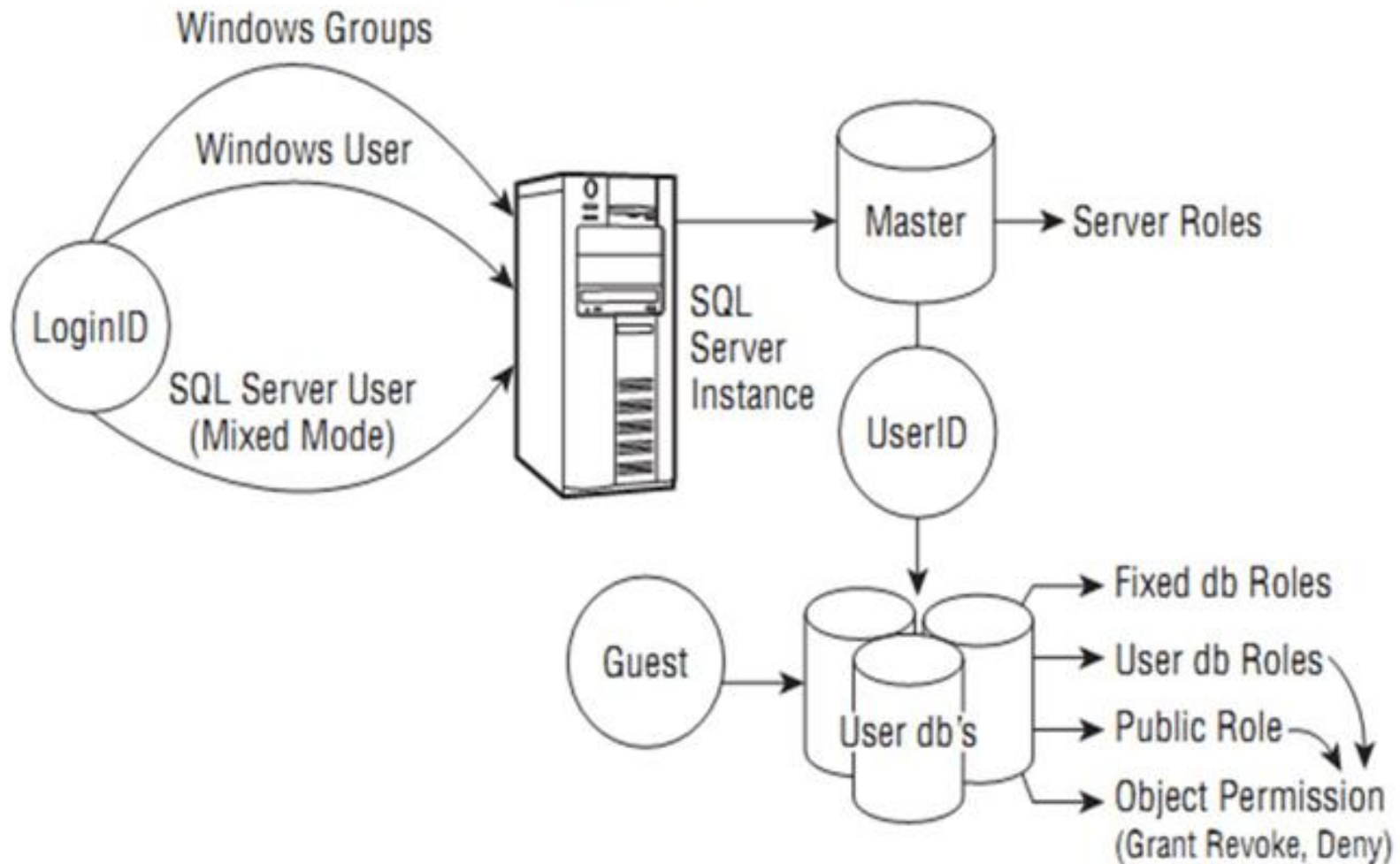
- **Login security:** user có thể log vào SQL Server
- **Database access security:** user có thể truy cập vào một DB cụ thể trên server
- **Permission security:** user có thể thực hiện thao tác gì trên DB

BẢO MẬT TRONG CSDL



BẢO MẬT TRONG CSDL

SQL Server Security Model



BẢO MẬT TRONG CSDL

- ❑ SQL Server sử dụng **Permission** và **Role** để bảo mật CSDL
 - **Permission**: Quy định các actions mà người dùng thực hiện trên các đối tượng CSDL
 - **Role**: tập các quyền được gán cho người dùng.
- ❑ SQL server dựa vào Permission và Role để xác định các đối tượng, hành động mà người dùng được phép thực hiện trên CSDL

MÔ HÌNH BẢO MẬT TRONG SQL SERVER



Network Connection Request / Pre-login Handshake

Connect to the SQL Server Computer



Login Authentication request to SQL Server

Establish Login Credentials



Switch to a database and Authorize access

Establish a Database Context



Attempt to perform some action

Verify permissions for all actions within a database

LOGIN SECURITY

- ❑ Có hai chế độ chứng thực
 - Windows Authentication
 - SQL Server Authentication

LOGIN SECURITY

❑ Windows Authentication

- Users chỉ cần được cấp account trong Windows
- SQL Server dựa vào Windows để chứng thực cho user

Cách kết nối này gọi là kết nối tin tưởng, dựa vào ủy nhiệm bảo mật của Windows

LOGIN SECURITY

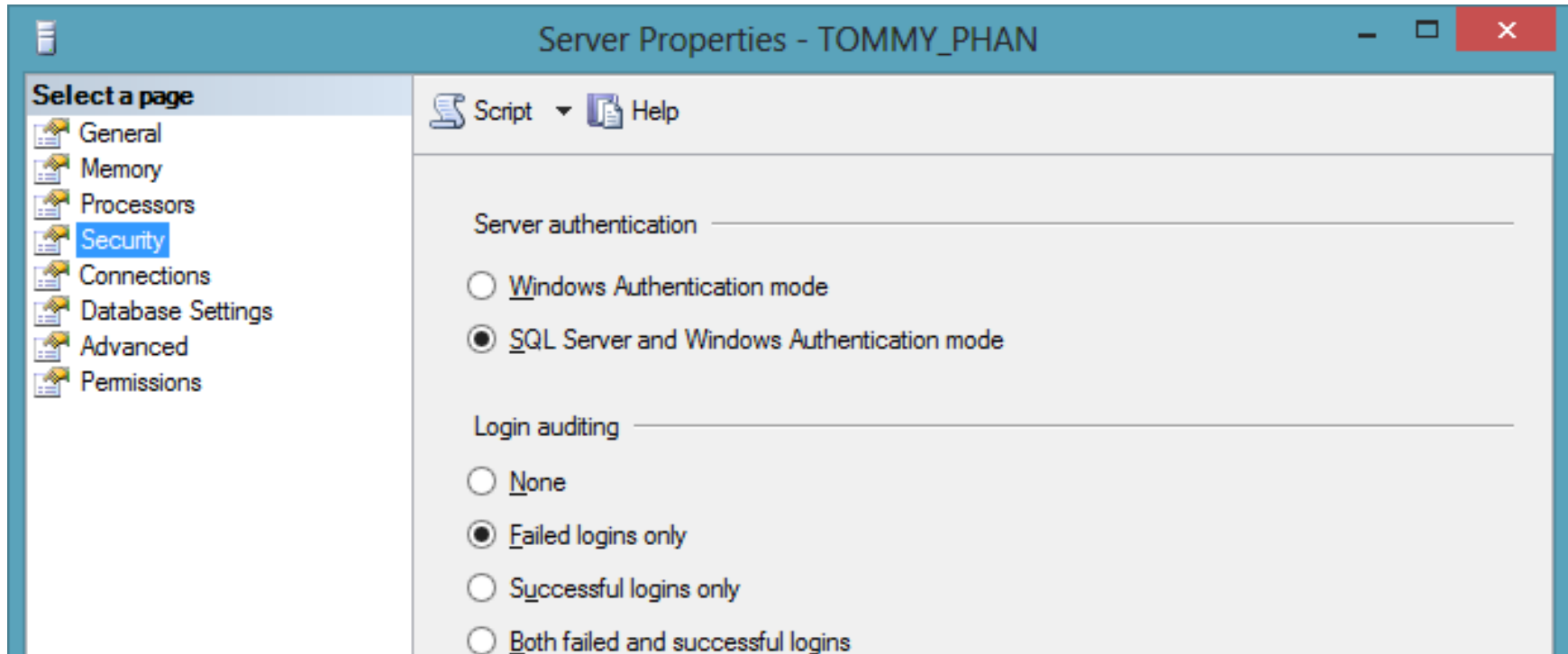
❑ SQL Server Authentication

- Người quản trị CSDL tạo ra tài khoản và password đăng nhập của SQL Server.

*Thông tin đăng nhập được lưu trong bảng **syslogins** của CSDL **master***

Cách thay đổi chế độ chứng thực

- ❑ Click phải trên Server → chọn Properties
- ❑ Trong khung Select a page → chọn Security



LOGINS

❑ Windows logins:

- Tài khoản user hay group lưu trữ trong Active Directory hay local Security Accounts Manager (SAM) database.

❑ SQL logins:

- Dùng cho các đối tượng không có tài khoản Windows
- Dựa vào thông tin lưu trữ và quản lý tài khoản của SQL Server

LOGIN ID VÀ USER ID

❑ Login ID:

- Là tài khoản mà người dùng sử dụng để truy cập vào hệ thống SQL Server
- Các Login chỉ mới có quyền truy cập vào Server, chưa có quyền truy cập vào các Database trên Server.
- Các quyền truy cập vào Database được gắn liền với các người dùng

LOGIN ID VÀ USER ID

❑ User ID

- Nhận dạng người dùng trong một cơ sở dữ liệu.
- Mỗi user luôn được gắn (mapped) với một login ở mức Server

LOGIN ID VÀ USER ID

- ❑ Một **login ID** phải kết hợp với 1 **user ID** trong mỗi DB để truy xuất dữ liệu trong DB.
- ❑ Nếu login ID không được kết hợp tương ứng với 1 user ID thì nó sẽ kết hợp với user **guest**.
 - Nếu DB không có user ID guest thì không thể truy xuất vào DB được
 - **sa** là 1 login account được ánh xạ tự động với user ID **dbo** trong mọi DB.
 - **dbo** là user có tất cả các quyền trên DB

Tạo login trong SSMS

- ❑ Trong Object Explorer, chọn server → Mở thư mục Security → R_Click Logins → chọn “New Login”
 - Nếu tạo Windows login: nhập tên login muốn tạo theo <domain\username>
 - Nếu tạo SQL Login: chọn “SQL Server authentication”
- ❑ Chọn CSDL và ngôn ngữ mặc định

Tạo login trong SSMS

Khi chọn “SQL Server authentication,” ta có thể chọn để trống password

The screenshot shows the 'Login - New' dialog box in SQL Server Enterprise Manager. The 'General' tab is selected in the left-hand pane. The 'Login name' field is empty. Under the 'Authentication' section, 'SQL Server authentication' is selected with a radio button. The 'Password' field is empty, and the 'Specify old password' checkbox is unchecked. The 'Enforce password policy', 'Enforce password expiration', and 'User must change password at next login' checkboxes are all checked.

Tạo login bằng T-SQL

❑ Tạo SQL login

CREATE LOGIN *login_name*

WITH PASSWORD='password' [MUST_CHANGE]

[, DEFAULT_DATABASE = database_name]

[, DEFAULT_LANGUAGE = language]

[, CHECK_EXPIRATION = { ON | OFF}]

[, CHECK_POLICY = { ON | OFF}]

Tạo login bằng T-SQL

❑ Ví dụ:

```
create login loginname with  
password='P@ssword123' MUST_CHANGE,  
CHECK_EXPIRATION =ON,  
default_database=qlbh
```

Tạo login bằng T-SQL

❑ Tạo Windows login

**CREATE LOGIN [Tên Domain\Tên User]
FROM WINDOWS**

Ví dụ:

Create Login [H51m11\user1] From Windows

Tạo login bằng T-SQL

❑ Quy ước đặt Password:

- Không sử dụng các từ “Password”, “Admin”, “sa”, “sysadmin”, “Administrator”
- Không sử dụng tên máy, tên người dùng hiện hành
- Trên 8 ký tự, bao gồm chữ cái, số và ký tự đặc biệt

Xóa, xem login bằng T-SQL

❑ Đổi Password:

```
ALTER LOGIN Login_name  
WITH PASSWORD = 'newpassword'
```

❑ Xóa login:

```
DROP LOGIN login_name
```

❑ Xem các login:

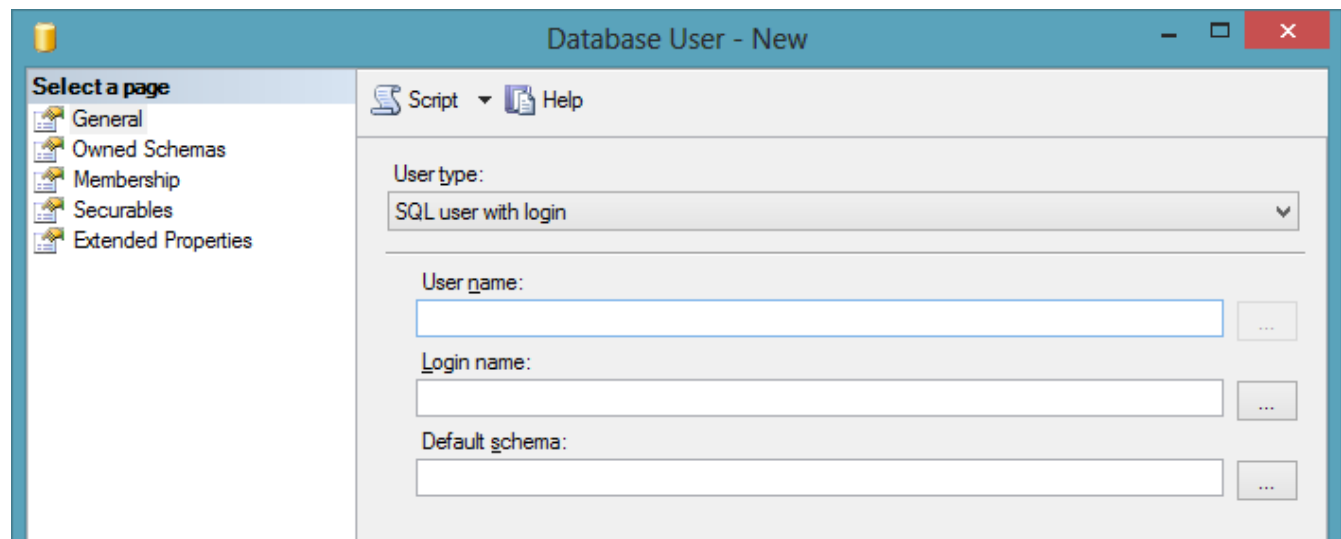
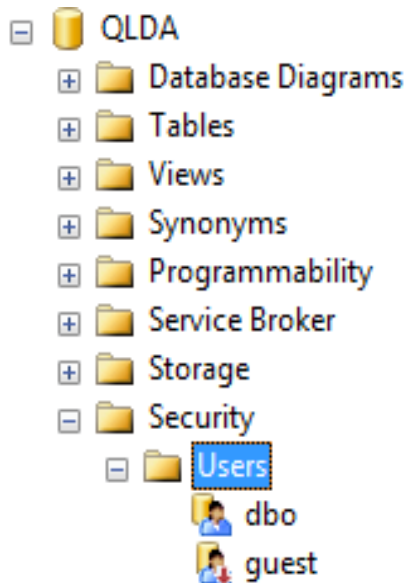
```
EXEC sp_helplogins 'sa'  
EXEC sp_helplogins
```


Database user

- ❑ Mỗi CSDL có một danh sách người dùng được xác thực để truy cập CSDL đó
- ❑ Khi tạo một user:
 - User chỉ có quyền chọn ngữ cảnh CSDL, không có quyền thực thi các thao tác trên CSDL và các đối tượng trong CSDL
 - Để có thể thực hiện những thao tác này người dùng phải được cấp quyền

Tạo user sử dụng SSMS

- ❑ Chọn CSDL → mở rộng thư mục Security
 - Click phải trên Users → New User → Nhập User name
 - Chọn Login và Schema cho user



Tạo user sử dụng T-SQL

❑ **CREATE USER** <Tên user>

[{FOR| FROM} LOGIN <*Tên login đăng nhập*>]

[WITH DEFAULT_SCHEMA=<*Tên schema*>]

Tạo user sử dụng T-SQL

❑ Ví dụ:

```
CREATE LOGIN AbolrousHazem WITH PASSWORD =  
'340$Uuxwp7Mcxo7Khy';  
GO
```

Use AdventureWorks2008;

```
CREATE USER AbolrousHazem FOR LOGIN  
AbolrousHazem;  
  
GO
```

```
CREATE LOGIN User2 with password='@user2'  
CREATE USER User2 FOR LOGIN User2  
  
go
```

Hiệu chỉnh và xóa User

❑ Hiệu chỉnh user

ALTER USER <Tên user> WITH
[NAME= <*Tên user mới*>]
[, DEFAULT_SCHEMA=<*Tên schema*>]

❑ Xóa user

DROP USER < Tên user>

Permission - Roles

Các quyền chuẩn trong SQL

Quyền	Các thao tác được phép thực hiện	Đối tượng áp dụng
SELECT	Truy xuất dữ liệu	Bảng, View, Hàm giá trị bảng
UPDATE	Cập nhật dữ liệu	Bảng, View, Hàm giá trị bảng
INSERT	Thêm dữ liệu mới	Bảng, View, Hàm giá trị bảng
DELETE	Xóa dữ liệu	Bảng, View, Hàm giá trị bảng
EXECUTE	Thực thi một Stored Procedure hay một hàm	Stored procedure, Hàm vô hướng và hàm kết hợp
REFERENCES	Tạo các đối tượng tham chiếu tới đối tượng này	Bảng, View, Hàm
ALL	Có tất cả các quyền đối với đối tượng	Bảng, View, Hàm , Stored Procedure

Roles

- **Roles (Vai trò)**: Tập các quyền dùng để gán cho một người dùng hoặc nhóm người dùng.

❑ Các Roles mặc định của SQL Server

- Server Role (Fixed Server Role)
- Database Role (Fixed Database Role)

Roles

- ❑ Có thể định nghĩa thêm các Role mới
- ❑ Mỗi Role được gán một tập PERMISSION.
- ❑ Ví dụ:
 - Role **dbcreator** có thể thực thi các câu lệnh:
 - CREATE/ALTER/DROP DATABASE
 - RESTORE DATABASE

SERVER ROLES

❑ **Server Roles:** mặc định bao gồm những người dùng quản trị Server

Roles	Mô tả
sysadmin	Có thể thực hiện mọi thao tác trên server. Theo mặc định, tất cả thành viên trong nhóm Windows BUILTIN\Administrators đều là thành viên của role này.
securityadmin	Có thể quản lý ID và mật khẩu đăng nhập cho server, đồng thời có thể cấp, từ chối và thu hồi quyền trên cơ sở dữ liệu
dbcreator	Có thể tạo, thay đổi, xóa và khôi phục cơ sở dữ liệu.
...	...

Database Role

Roles	Mô tả
Db_owner	Có tất cả các quyền đối với CSDL
Db_accessadmin	Có quyền thêm hoặc xóa một LoginID của CSDL
Db_securityadmin	Có thể quản trị quyền đối tượng, quyền CSDL, Vai trò, các thành viên của Vai trò
Db_datawriter	Có thể thêm, xóa, cập nhật dữ liệu trên toàn bộ các bảng trong CSDL
Db_datareader	Có thể truy xuất dữ liệu từ tất cả các bảng trong CSDL
Db_denydatawriter	Không thể thêm, xóa, cập nhật dữ liệu trên toàn bộ các bảng trong CSDL
Db_denydatareader	Không thể truy xuất dữ liệu từ tất cả các bảng trong CSDL
Db_backupoperator	Có thể thực hiện sao lưu CSDL và chạy các kiểm tra tính nhất quán trên CSDL

Gán Server Role cho một login ID

- ❑ **Cách 1:** Sử dụng Server Roles trong **Login** Properties để chọn và gán Server Role cho một login
- ❑ **Cách 2:** Sử dụng Server **Roles** Properties để thêm login ID vào danh sách thành viên của Server Role

Gán Database Role cho một Login ID

Login Properties - xuanhien

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Users mapped to this login:

Map	Database	User	Default Schema
<input type="checkbox"/>	AdventureWorks2012		
<input checked="" type="checkbox"/>	COMPANY	xuanhien	...
<input type="checkbox"/>	master		
<input type="checkbox"/>	model		
<input type="checkbox"/>	msdb		
<input type="checkbox"/>	NorthWind		
<input type="checkbox"/>	qlbh		
<input type="checkbox"/>	QLDA		
<input type="checkbox"/>	quanlisach		
<input type="checkbox"/>	ReportServer		

☐ Guest account enabled for: COMPANY

Database role membership for: COMPANY

- ☐ db_accessadmin
- ☐ db_backupoperator
- ☐ db_datareader
- ☐ db_datawriter
- ☐ db_ddladmin
- ☐ db_denydatareader
- ☐ db_denydatawriter
- ☐ db_owner
- ☐ db_securityadmin
- ☒ public

Connection

Server:
TOMMY_PHAN

Connection:
sa

[View connection properties](#)

Progress

Ready

OK Cancel

Tạo một login với fixed Server Role

❑ Cú pháp:

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

❑ Ví dụ:

– Tạo:

```
CREATE LOGIN Ted WITH PASSWORD = 'P@ssw0rd';  
EXEC sp_addsrvrolemember 'Ted', 'securityadmin';
```

– Xóa:

```
EXEC sp_dropsrvrolemember 'Ted', 'securityadmin';
```

Tạo một login với fixed Server Role

❑ Cú pháp:

- Xem các server role:

EXEC sp_helpsrvrole

- Xem các thành viên của 1 fixed server role:

EXEC sp_helpsrvrolemember ['role']

- Xem các quyền của 1 fixed server role:

EXEC sp_srvrolepermission ['role']

Tạo Database Role

❑ Cú pháp:

```
CREATE ROLE role_name
```

❑ Ví dụ:

– Tạo:

```
CREATE ROLE Employee_Role
```


Thêm user vào Database Role

❑ Cú pháp:

```
sp_addrolemember [ @rolename = ] 'role',  
[ @membername = ] 'security_account'
```

❑ Ví dụ:

– Thêm user vào Role:

```
EXEC sp_addrolemember Employee_Role, User2
```

– Xóa user khỏi Role:

```
EXEC sp_droprolemember Employee_Role, User2
```

Tạo Database Users

- ❑ Chọn folder Databases, Chọn CSDL
- ❑ Mở Security.
- ❑ R_Click **Users** và chọn **New User**.
- ❑ Nhập tên user vào **User Name box**.
 - VD: Carol
 - Nhập tên login (VD: **Carol**) trong “Login name” box, hoặc chọn tên login bằng cách click “...” button.
 - Nhập tên nhánh CSDL (VD: **Sales**) trong “Default schema” box. Click OK.

Tạo mới DB Users bằng T-SQL

❑ Cú pháp

```
CREATE USER name [{{FOR | FROM} source /  
WITHOUT LOGIN]  
[WITH DEFAULT_SCHEMA = schema_name]
```

❑ Hiệu chỉnh

```
ALTER USER <Tên user> WITH  
[NAME = <Tên user mới>]  
[, DEFAULT_SCHEMA = <Tên schema>]
```

❑ Xóa DROP USER <Tên user>

❑ Xem thông tin user

```
EXEC sp_helpuser 'dbo'  
EXEC sp_helpuser
```

Ví dụ

```
USE master;
```

```
CREATE LOGIN [AughtEight\Bob] FROM WINDOWS;
```

```
-- CREATE LOGIN [domain name\login name]
```

```
USE AdventureWorks2008;
```

```
CREATE USER BillyBob FOR LOGIN [AughtEight\Bob]
```

```
WITH DEFAULT_SCHEMA = Sales;
```

Các thủ tục thường thao tác với database user

Stored Procedure	Description
<code>sp_adduser</code>	Creates a new database user.
<code>sp_grantdbaccess</code>	Creates a new database user.
<code>sp_dropuser</code>	Removes a database user.
<code>sp_revokedbaccess</code>	Removes a database user.
<code>sp_addrole</code>	Creates a new user-defined database role.
<code>sp_droprole</code>	Removes a user-defined database role.
<code>sp_addapprole</code>	Creates a new application role.
<code>sp_approlepassword</code>	Changes the password for an application role.
<code>sp_dropapprole</code>	Removes an application role from the database.

Quyền (permission)

❑ Có 3 loại quyền

- Object Permissions
- Statement Permissions
- Implied Permissions

❑ Tất cả các quyền trong SQL server tồn tại 1 trong 3 trạng thái:

- GRANTED (cấp quyền)
- REVOKED (thu hồi)
- DENIED (từ chối)

Quyền trên các đối tượng

- ☐ SELECT
- ☐ INSERT
- ☐ UPDATE
- ☐ DELETE
- ☐ REFERENCES
- ☐ EXECUTE

Quyền trên các câu lệnh

- ☐ BACKUP DATABASE
- ☐ BACKUP LOG
- ☐ CREATE DATABASE
- ☐ CREATE DEFAULT
- ☐ CREATE FUNCTION
- ☐ CREATE PROCEDURE
- ☐ CREATE RULE
- ☐ CREATE TABLE
- ☐ CREATE VIEW

Kế hoạch phân quyền và bảo mật

Cấp phát quyền

❑ Câu lệnh GRANT được sử dụng để cấp phát quyền cho người dùng hay nhóm người dùng trên các đối tượng cơ sở dữ liệu. Câu lệnh này thường được sử dụng trong các trường hợp sau:

- Người sở hữu đối tượng cơ sở dữ liệu muốn cho phép người dùng khác quyền sử dụng những đối tượng mà anh ta đang sở hữu.
- Người sở hữu cơ sở dữ liệu cấp phát quyền thực thi các câu lệnh (như CREATE TABLE, CREATE VIEW,...) cho những người dùng khác.

GRANT

- **GRANT** : gán quyền trên đối tượng (cú pháp đầy đủ)

GRANT

{ *ALL* | *permission* [,...*n*] }

{

[(*column* [,...*n*])] ON { *table* | *view* }

| ON { *table* | *view* } [(*column* [,...*n*])]

| ON { *stored_procedure* | *extended_procedure* }

| ON { *user_defined_function* }

}

TO *security_account* [,...*n*] [WITH GRANT OPTION]

[AS { *group* | *role* }]

Kế hoạch phân quyền và bảo mật

Trong đó:

ALL [PRIVILEGES]: Cấp phát tất cả các quyền cho người dùng trên đối tượng cơ sở dữ liệu được chỉ định. Các quyền có thể cấp phát cho người dùng bao gồm:

- ❑ Đối với bảng, khung nhìn, và hàm trả về dữ liệu kiểu bảng: SELECT, INSERT, DELETE, UPDATE và REFERENCES.
- ❑ Đối với cột trong bảng, khung nhìn: SELECT và UPDATE.
- ❑ Đối với thủ tục lưu trữ và hàm vô hướng: EXECUTE.

Trong các quyền được đề cập đến ở trên, quyền REFERENCES được sử dụng nhằm cho phép tạo khóa ngoài tham chiếu đến bảng cấp phát

Kế hoạch phân quyền và bảo mật

- *Các_quyền_cấp_phát*: Danh sách các quyền cần cấp phát cho người dùng trên đối tượng cơ sở dữ liệu được chỉ định. Các quyền được phân cách nhau bởi dấu phẩy
- *Tên_bảng/tên_khung_nhìn*: Tên của bảng hoặc khung nhìn cần cấp phát quyền.
- *Danh_sách_cột*: Danh sách các cột của bảng hoặc khung nhìn cần cấp phát quyền.
- *Tên_thủ_tục*: Tên của thủ tục được cấp phát cho người dùng.

Kế hoạch phân quyền và bảo mật

- *Tên_hàm*: Tên hàm (do người dùng định nghĩa) được cấp phát quyền.
- *Danh_sách_người_dùng*: Danh sách tên người dùng nhận quyền được cấp phát. Tên của các người dùng được phân cách nhau bởi dấu phẩy.
- WITH GRANT OPTION: Cho phép người dùng chuyển tiếp quyền cho người dùng khác.

Kế hoạch phân quyền và bảo mật

Ví dụ:

Cấp phát cho người dùng có tên *thuchanh* quyền thực thi các câu lệnh SELECT, INSERT và UPDATE trên bảng Products

```
GRANT SELECT, INSERT, UPDATE  
ON Products  
TO thuchanh
```

Kế hoạch phân quyền và bảo mật

Ví dụ:

Cho phép người dùng *thuchanh* quyền xem ProductName, UnitsInstock của bảng Products

```
GRANT SELECT  
(ProductName, UnitsInstock) ON Products  
TO thuchanh
```

hoặc:

```
GRANT SELECT  
ON Products (ProductName, UnitsInstock)  
TO thuchanh
```

Kế hoạch phân quyền và bảo mật

Ví dụ:

Với quyền được cấp phát như trên, người dùng *thuchanh* có thể thực hiện câu lệnh sau trên bảng Products

```
SELECT ProductName, UnitsInstock from  
Products
```

Nhưng câu lệnh dưới đây lại không thể thực hiện được

```
SELECT * FROM Products
```


Kế hoạch phân quyền và bảo mật

- Trong trường hợp cần cấp phát tất cả các quyền có thể thực hiện được trên đối tượng cơ sở dữ liệu cho người dùng, thay vì liệt kê các câu lệnh, ta chỉ cần sử dụng từ khoá ALL PRIVILEGES (từ khoá PRIVILEGES có thể không cần chỉ định).
- Câu lệnh dưới đây cấp phát cho người dùng *thuchanh* các quyền SELECT, INSERT, UPDATE, DELETE VÀ REFERENCES trên bảng [Order Details]

```
GRANT ALL  
ON [Order Details]  
TO thuchanh
```

Kế hoạch phân quyền và bảo mật

Chú ý:

- Người dùng không có quyền cấp phát những quyền mà mình được phép cho những người sử dụng khác.
- Trong một số trường hợp, khi ta cấp phát quyền cho một người dùng nào đó, ta có thể cho phép người đó chuyển tiếp quyền cho người dùng khác bằng cách chỉ định tùy chọn `WITH GRANT OPTION` trong câu lệnh `GRANT`.

Kế hoạch phân quyền và bảo mật

Ví dụ: Cho phép người dùng *thuchanh* quyền xem dữ liệu trên bảng Products đồng thời có thể chuyển tiếp quyền này cho người dùng khác

GRANT SELECT

ON Products

TO thuchanh

WITH GRANT OPTION

Kế hoạch phân quyền và bảo mật

Cấp phát quyền thực thi các câu lệnh

- ❑ Lệnh GRANT còn có thể sử dụng để cấp phát cho người sử dụng một số quyền trên hệ quản trị cơ sở dữ liệu hoặc cơ sở dữ liệu.
- ❑ Những quyền có thể cấp phát trong trường hợp này bao gồm:
 - Tạo cơ sở dữ liệu: CREATE DATABASE.
 - Tạo bảng: CREATE TABLE
 - Tạo khung nhìn: CREATE VIEW
 - Tạo thủ tục lưu trữ: CREATE PROCEDURE
 - Tạo hàm: CREATE FUNCTION
 - Sao lưu cơ sở dữ liệu: BACKUP DATABASE

Kế hoạch phân quyền và bảo mật

Cấp phát quyền thực thi các câu lệnh

Cú pháp:

```
GRANT ALL | danh_sách_câu_lệnh  
TO danh_sách_người_dùng
```

Ví dụ: Để cấp phát quyền tạo bảng và khung nhìn cho người dùng có tên là *thuchanh*, ta sử dụng câu lệnh như sau:

```
GRANT CREATE TABLE, CREATE VIEW  
TO thuchanh
```

Ví dụ về GRANT

- ❑ GRANT INSERT, SELECT ON Sailors TO Horatio
 - Horatio có thể truy vấn hoặc thêm dòng mới vào table Sailors
- ❑ GRANT DELETE ON Sailors TO Yuppy
WITH GRANT OPTION
 - Yuppy có thể xóa dữ liệu của table Sailors và có thể ủy quyền cho người khác
- ❑ GRANT UPDATE (*rating*) ON Sailors TO Dustin
 - Dustin có thể cập nhật cột *rating* trên các dòng của table Sailor
- ❑ GRANT SELECT ON ActiveSailors TO Guppy, Yuppy
 - Guppy, Yuppy không truy cập trực tiếp table Sailors mà thông qua view ActiveSailors

Kế hoạch phân quyền và bảo mật

Thu hồi quyền

- ❑ Câu lệnh REVOKE được sử dụng để thu hồi quyền đã được cấp phát cho người dùng.
- ❑ Tương ứng với câu lệnh GRANT, câu lệnh REVOKE được sử dụng trong hai trường hợp:
 - Thu hồi quyền đã cấp phát cho người dùng trên các đối tượng cơ sở dữ liệu.
 - Thu hồi quyền thực thi các câu lệnh trên cơ sở dữ liệu đã cấp phát cho người dùng

REVOKE

❑ **REVOKE:** thu hồi lại quyền đã được cấp hay từ chối từ 1 user của CSDL hiện hành

❑ **Cú pháp:**

```
REVOKE [GRANT OPTION FOR]  
<permissions> [ON <object>]  
FROM <user/role>
```

Ví dụ:

```
REVOKE select, insert, update ON titles  
FROM faculty
```


Kế hoạch phân quyền và bảo mật

❑ Cú pháp đầy đủ:

```
REVOKE [GRANT OPTION FOR]
  {ALL [PRIVILEGES] | permission[,...n]}
  {
    [(column[,...n])] ON {table | view}
    | ON {table | view}[(column[,...n])]
    | {stored_procedure | extended_procedure}
  }
{TO | FROM}
  security_account[,...n]
[CASCADE]
[AS {group | role}]
```

Kế hoạch phân quyền và bảo mật

- ❑ **Ví dụ:** Thu hồi quyền thực thi lệnh INSERT trên bảng Products đối với người dùng *thuchanh*.

REVOKE INSERT

ON Products

FROM thuchanh

Kế hoạch phân quyền và bảo mật

- ❑ Thu hồi quyền đã cấp phát trên cột UnitsInstock (chỉ cho phép xem dữ liệu trên cột ProductName)

REVOKE SELECT

ON Products(UnitsInstock)

FROM thuchanh

- ❑ **Chú ý:** Khi ta sử dụng câu lệnh REVOKE để thu hồi quyền trên một đối tượng cơ sở dữ liệu từ một người dùng nào đó, chỉ những quyền mà ta đã cấp phát trước đó mới được thu hồi, những quyền mà người dùng này được cho phép bởi những người dùng khác vẫn còn có hiệu lực.

Kế hoạch phân quyền và bảo mật

- ❑ **Ví dụ:** Giả sử trong cơ sở dữ liệu ta có 3 người dùng là *A*, *B* và *C*. *A* và *B* đều có quyền sử dụng và cấp phát quyền trên bảng *R*. *A* thực hiện lệnh sau để cấp phát quyền xem dữ liệu trên bảng *R* cho *C*:

GRANT SELECT

ON R TO C

- ❑ *B* cấp phát quyền xem và bổ sung dữ liệu trên bảng *R* cho *C* bằng câu lệnh:

GRANT SELECT, INSERT

ON R TO C

Kế hoạch phân quyền và bảo mật

- ❑ Như vậy, C có quyền xem và bổ sung dữ liệu trên bảng R. Bây giờ, nếu B thực hiện lệnh:

**REVOKE SELECT, INSERT
ON R FROM C**

- ❑ Vậy C còn quyền gì trên R?
 - Người dùng C sẽ không còn quyền bổ sung dữ liệu trên bảng R nhưng vẫn có thể xem được dữ liệu của bảng này (quyền này do A cấp cho C và vẫn còn hiệu lực).

Kế hoạch phân quyền và bảo mật

- ❑ Nếu ta đã cấp phát quyền cho người dùng nào đó bằng câu lệnh GRANT với tùy chọn WITH GRANT OPTION thì khi thu hồi quyền bằng câu lệnh REVOKE phải chỉ định tùy chọn CASCADE.
- ❑ Trong trường hợp này, các quyền được chuyển tiếp cho những người dùng khác cũng đồng thời được thu hồi.

Kế hoạch phân quyền và bảo mật

- ❑ **Ví dụ:** Ta cấp phát cho người dùng A trên bảng R với câu lệnh GRANT như sau:

GRANT SELECT

ON R TO A

WITH GRANT OPTION

- ❑ Sau đó người dùng A lại cấp phát cho người dùng B quyền xem dữ liệu trên R với câu lệnh:

GRANT SELECT

ON R TO B

Kế hoạch phân quyền và bảo mật

- ❑ Nếu muốn thu hồi quyền đã cấp phát cho người dùng A, ta sử dụng câu lệnh REVOKE như sau:

REVOKE SELECT

ON R

FROM A CASCADE

- ❑ Câu lệnh trên sẽ đồng thời thu hồi quyền mà A đã cấp cho B và như vậy cả A và B đều không thể xem được dữ liệu trên bảng R.

Kế hoạch phân quyền và bảo mật

- ❑ Trong trường hợp cần thu hồi các quyền đã được chuyển tiếp và khả năng chuyển tiếp các quyền đối với những người đã được cấp phát quyền với tùy chọn WITH GRANT OPTION, trong câu lệnh REVOKE ta chỉ định mệnh đề GRANT OPTION FOR.

Kế hoạch phân quyền và bảo mật

❑ **Ví dụ:** Trong ví dụ trên, nếu ta thay câu lệnh:

REVOKE SELECT

ON Employees

FROM A CASCADE

bởi câu lệnh:

REVOKE GRANT OPTION FOR SELECT

ON Employees

FROM A CASCADE

Thì B sẽ không còn quyền xem dữ liệu trên bảng R đồng thời A không thể chuyển tiếp quyền mà ta đã cấp phát cho những người dùng khác (tuy nhiên A vẫn còn quyền xem dữ liệu trên bảng R).

Kế hoạch phân quyền và bảo mật

Thu hồi quyền thực thi các câu lệnh:

- ❑ Việc thu hồi quyền thực thi các câu lệnh trên cơ sở dữ liệu (CREATE DATABASE, CREATE TABLE, CREATE VIEW,...) được thực hiện đơn giản với câu lệnh REVOKE có cú pháp:

REVOKE ALL | *các_câu_lệnh_cần_thu_hồi*
FROM *danh_sách_người_dùng*

Kế hoạch phân quyền và bảo mật

Thu hồi quyền thực thi các câu lệnh:

- ❑ **Ví dụ:** Để không cho phép người dùng *thuchanh* thực hiện lệnh CREATE TABLE trên cơ sở dữ liệu, ta sử dụng câu lệnh:

**REVOKE CREATE TABLE
FROM thuchanh**

Kế hoạch phân quyền và bảo mật

- **DENY: từ chối** 1 permission và ngăn chặn 1 user, group, role thừa kế permission thông qua mối quan hệ thành viên trong group và role.

❑ Statement permissions:

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

❑ Object permissions:

```
DENY
  {ALL [PRIVILEGES] | permission[,...n]}
  {
    [(column[,...n])] ON {table / view}
    | ON {table | view}[(column[,...n])]
    | ON {stored_procedure | extended_procedure}
  } TO security_account[,...n][CASCADE]
```

DENY

❑ Nhắc lại cú pháp:

DENY <permissions>[ON <object>]

TO <user/role>

❑ Ví dụ:

Use pubs

DENY select, insert, update ON titles TO faculty