Chương 4
Phân quyền trong SQL Server

Giáo trình & Tài liệu tham khảo:

1. Microsoft SQL Server 2008 R2 Unleashed, Ray Rankins, Paul Bertucci, Chris Gallelli, Alex T. Silverstein, 2011, Pearson Education, Inc

2. **MS SQL Server 2012 T-SQL fundamentals**, Tizik Ben-Gan

3. https://docs.microsoft.com/

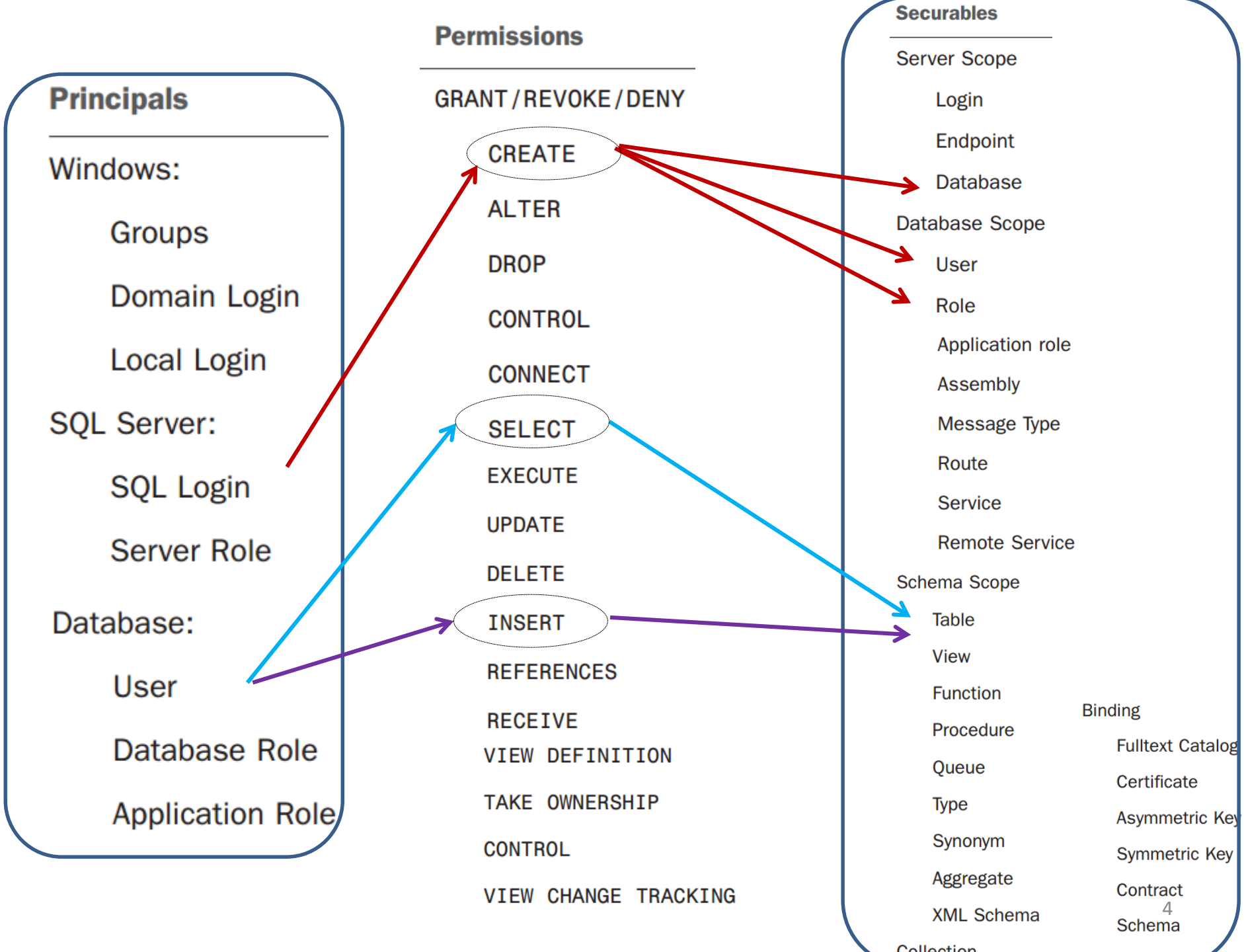
Gv. Nguyễn Như Hoa

Giới thiệu

- Phân quyền là một nội dung security
- Phân quyền trong SQL Server gồm các nội dung
  - Login và user
  - Role và permission

Các khái niệm căn bản

- Mô hình phân quyền của SQL Server được xây dựng dựa trên 3 yếu tố : Principals , Securables , Permissions
- Thiết lập phân quyền là thiết lập gắn kết 3 yếu tố trên, ví dụ
  - a SQL LOGIN (the principal) needs to CREATE (the permission) DATABASEs (the securable)

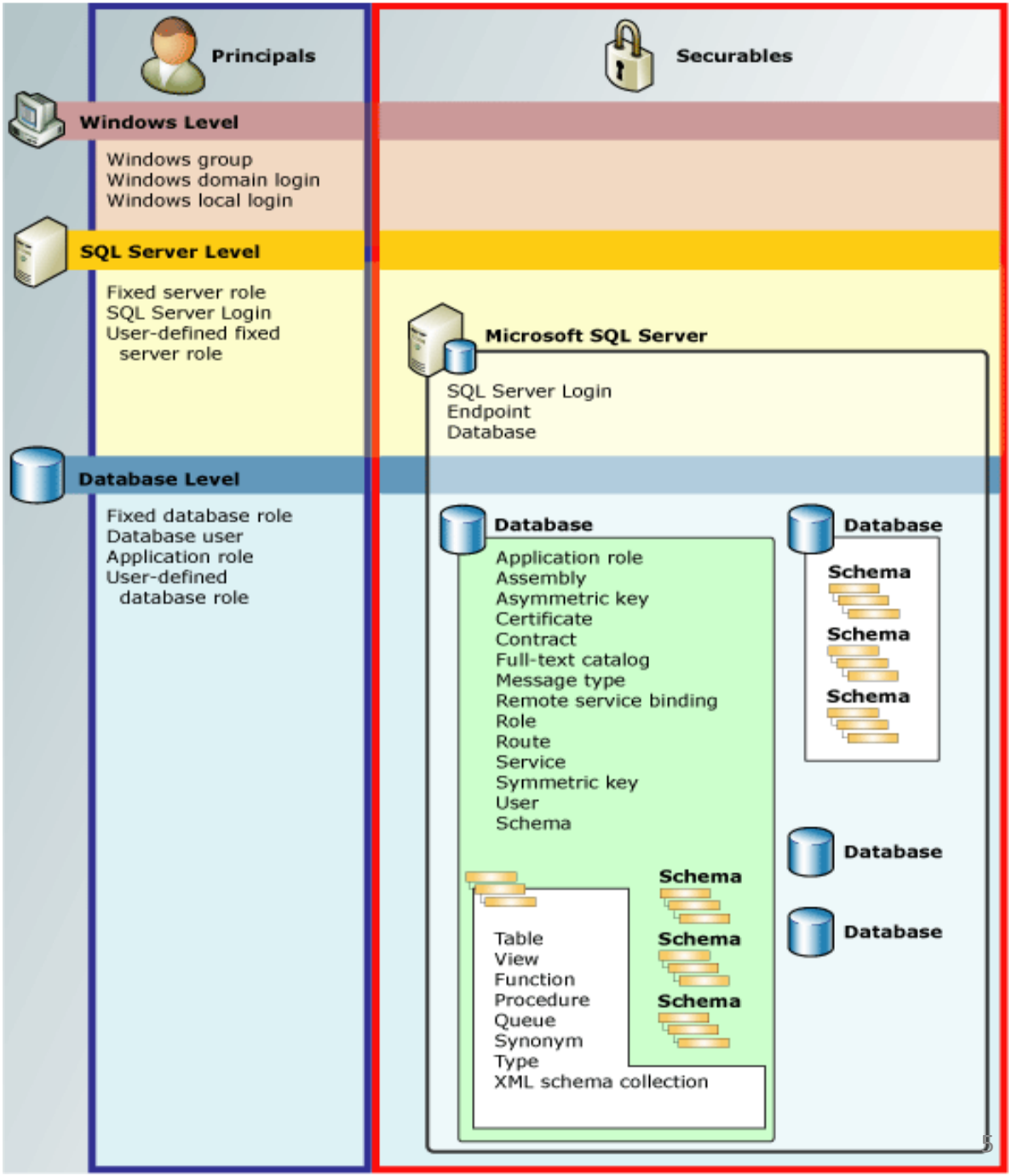

Principals
Windows:
Groups
Domain Login
Local Login
SQL Server:
SQL Login
Server Role
Database:
User
Database Role
Application Role
Permissions
GRANT/REVOKE/DENY
CREATE
ALTER
DROP
CONTROL
CONNECT
SELECT
EXECUTE
UPDATE
DELETE
INSERT
REFERENCES
RECEIVE
VIEW DEFINITION
TAKE OWNERSHIP
CONTROL
VIEW CHANGE TRACKING
Securables
Server Scope
Login
Endpoint
Database
Database Scope
User
Role
Application role
Assembly
Message Type
Route
Service
Remote Service
Schema Scope
Table
View
Function
Procedure
Queue
Type
Synonym
Aggregate
XML Schema
Binding
Fulltext Catalog
Certificate
Asymmetric Key
Symmetric Key
Contract
Schema

Đặc điểm :

1. Các Securable được phân cấp 3 cấp độ

- **Server Level**
- **Database Level**
- **Schema level**

2. Một số Securable có thể chứa các Securable khác tạo nên cấu trúc lồng nhau và phân quyền kéo theo. Vd?



Các khái niệm căn bản

- Logins
  - A login is the identity of the person or process that is connecting to an instance of SQL Server.
  - Có thể connect tới một instance của SQL Server bằng tài khoản của Windows hoặc bằng tài khoản do SQL Server quản lý
  - Quyền có thể cấp cho một login có phạm vi : server-level

Các khái niệm căn bản

- Database user
  - Một login cần phải được map với một database user khi muốn thao tác với một database trong server
  - Một login có thể map với một/hay nhiều database user có tên khác hay trùng với login name khi muốn thao tác trên nhiều database trong server
  - Quyền cấp cho một database user có phạm vi : database-level

Các khái niệm căn bản

- Roles
  - **Các quyền nào đó được gắn cho một role. Sau đó, các member được tham gia vào role. Bất kỳ member nào thuộc role cũng có tất cả quyền mà role có**
  - Một role được xem như một group : gồm nhiều thành viên (là các login hay các database user).

Các khái niệm căn bản

- Roles bao gồm
  - **Server-level roles**
    - Fixed Server roles
    - User-defined Server roles (*có từ SQL Server 2012*)
  - **Database-level roles**
    - Fixed Database roles
    - User-defined Database roles
    - Application roles
- Các **fixed server role** và **fixed database role**
  - Được định nghĩa trước, có sẵn trong mỗi server và mỗi database
  - Không thể thay đổi tập quyền trong các role này

Các khái niệm căn bản

- Permissions
  - Quyền trong Database Engine gồm
    - các quyền ở server-level : được gán cho logins và server roles
    - và các quyền ở database-level : được gán cho database users và database roles

Các khái niệm căn bản

- Permissions
  - Liệt kê tất cả các quyền

```
SELECT * FROM fn_builtin_permissions(default);
```

  - Liệt kê tất cả các quyền thuộc server-level

```
SELECT *
FROM sys.fn_builtin_permissions('SERVER')
ORDER BY permission_name;
```

  - Liệt kê tất cả các quyền thuộc database-level

```
SELECT *
FROM sys.fn_builtin_permissions(‘database')
ORDER BY permission_name;
```

Các khái niệm căn bản

- Permissions - Quản lý permissions
  - Quản lý permissions thông qua 3 lệnh : GRANT, REVOKE, và DENY
  - GRANT and DENY : create a permission rule
  - REVOKE : Removes a previously granted or denied permission rule
  - DENY always overrides a GRANT

Các khái niệm căn bản

- Permissions - Lưu ý

  Ảnh hưởng của thứ tự thực hiện các lệnh

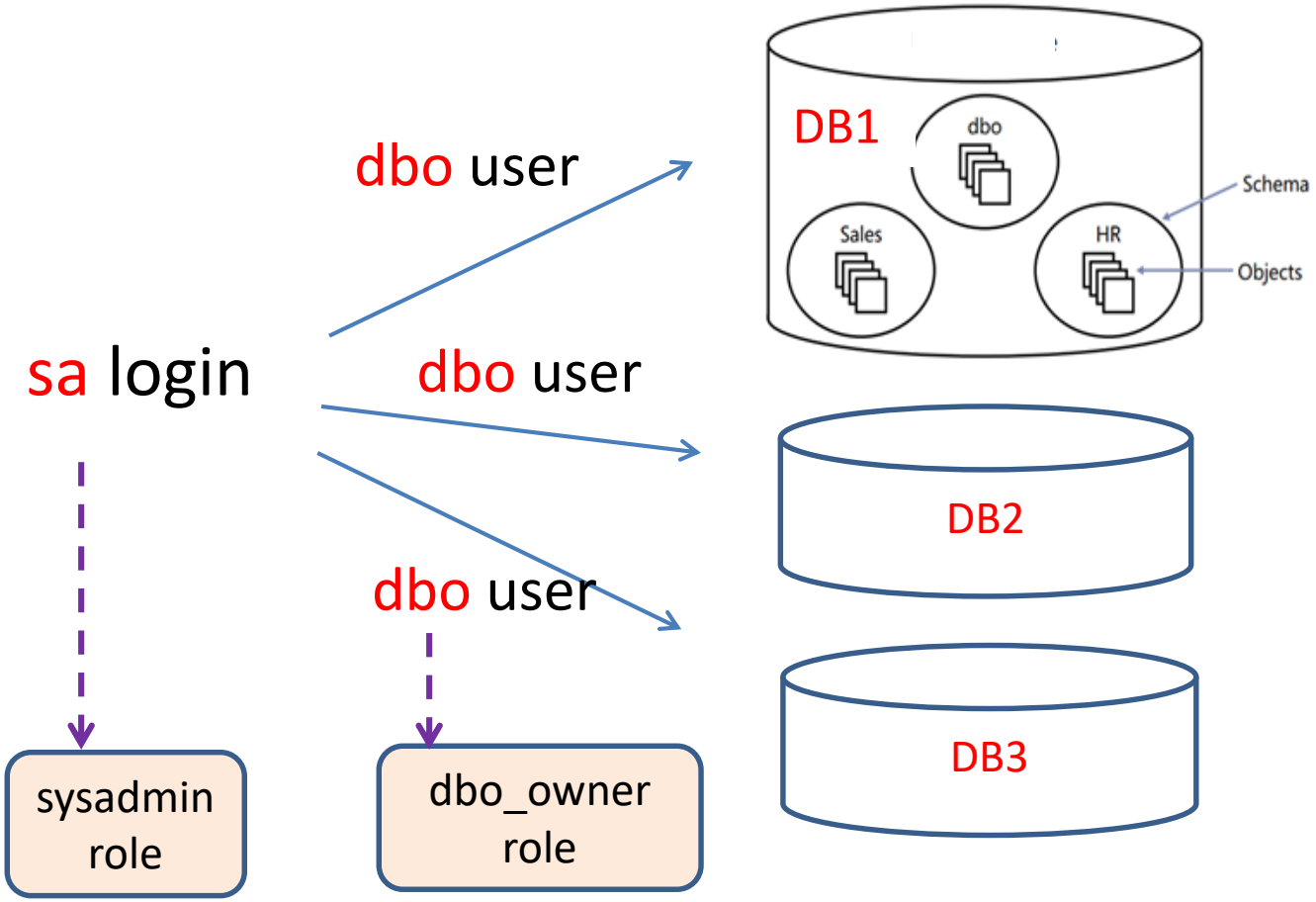
  – A GRANT of a permission removes any REVOKE or DENY on a securable. For example, if a table has SELECT permission denied on it and then the SELECT permission is granted, the DENY permission is then removed on that table.
  – DENY and REVOKE remove any GRANT permission on a securable.
  – REVOKE removes any GRANT or DENY permission on a securable.
  – Permissions denied at a higher scope in the security model override grants on that permission at a lower scope.
  – Permissions granted at a higher scope in the security model are overridden by a DENY permission at a lower level

sa login & dbo user

- sa login
  - Là một SQL Server login, được tự động tạo khi cài đặt SQL Server, và không thể xóa bỏ
  - Là thành viên của ***sysadmin server role***
  - Chỉ cấp cho người quản trị server
- dbo user
  - Là một database user được map tự động cho mọi login là thành viên của *sysadmin server role*, hay *db_owner database role ,* hay là *owner* của database
  - là thành viên của ***db_owner database role***

=> sa login được map với dbo user khi truy suất vào bất kỳ database nào trong server

sa login & dbo user



The dbo, or database owner, is a user account that has implied permissions to perform all activities in the database. Members of the sysadmin fixed server role are automatically mapped to dbo.

Server Roles
Fixed Server Roles

Role	Permission
bulkadmin	Allowed to run the BULK INSERT statement.
dbcreator	Allowed to use CREATE, ALTER, DROP, and RESTORE on any database.
diskadmin	Allowed to manage disk files that are used by SQL Server.
processadmin	Allowed to terminate SQL Server processes.
public	Assigned to all logins. Permissions granted to this role are assigned to every login by default.
securityadmin	Allowed to use GRANT, DENY, and REVOKE permissions for logins at the server and database levels. Members of this role can reset passwords for SQL Server logins.
serveradmin	Allowed to change server-wide configuration properties and shut down the server, if needed.
setupadmin	Allowed to add and remove linked servers and execute some system stored procedures.
sysadmin ★	Allowed to perform any activity in the server.

Server Roles
Xem thông tin

Feature	Type	Description
sp_helpsrvrole	Metadata	Returns a list of server-level roles.
sp_helpsrvrolemember	Metadata	Returns information about the members of a server-level role.
sp_srvrolepermission	Metadata	Displays the permissions of a server-level role.

Server Roles

Các thao tác trên Server Roles

Feature	Type	Description
ALTER SERVER ROLE	Command	Adds a login as a member of a server-level role.
ALTER SERVER ROLE	Command	Removes a SQL Server login or a Windows user or group from a server-level role.
CREATE SERVER ROLE	Command	Creates a user-defined server role.
ALTER SERVER ROLE	Command	Changes the membership of a server role or changes name of a user-defined server role.
DROP SERVER ROLE	Command	Removes a user-defined server role.

https://docs.microsoft.com/

Database Roles
Các fixed database role

Role	Permission
db_accessadmin	Allowed to add or remove database access for logins.
db_backupoperator	Allowed to back up the database.
db_datareader	Allowed to read all user table data.
db_datawriter	Allowed to change the data in all user tables.
db_ddladmin	Allowed to run any Data Definition Language (DDL) command against the database. This includes commands to create, alter, and drop database objects.
db_denydatareader	Denied the right to read all user table data.
db_denydatawriter	Denied the right to change the data in any of the user tables.
db_owner ★	Allowed to perform any action on the database. Members of the sysadmin fixed server role are mapped to this database role.
db_securityadmin	Allowed to manage permissions for database users, including membership in roles.
dbm_monitor	Allowed to view the most recent status in the Database Mirroring Monitor.

Database roles

Xem thông tin và Các thao tác trên role

Feature	Type	Description
sp_helpdbfixedrole	Metadata	Returns a list of the fixed database roles.
sp_dbfixedrolepermission	Metadata	Displays the permissions of a fixed database role.
sp_helprole	Metadata	Returns information about the roles in the current database.
sp_helprolemember	Metadata	Returns information about the members of a role in the current database.

CREATE ROLE	Creates a new database role in the current database.
ALTER ROLE	Changes the name or membership of a database role.
DROP ROLE	Removes a role from the database.
sp_addrole	Creates a new database role in the current database.
sp_droprole	Removes a database role from the current database.
sp_addrolemember	**Adds a database user**, database role, Windows login, or Windows group **to a database role in the current database.** All platforms except Parallel Data Warehouse and Azure Synapse should use ALTER ROLE instead.
sp_droprolemember	Removes a security account from a SQL Server role in the current database. All platforms except Parallel Data Warehouse and Azure Synapse should use ALTER ROLE instead.
GRANT	Adds permission to a role.
DENY	Denies a permission to a role.
REVOKE	Removes a previously granted or denied permissions.

Quản trị phân quyền

- Quản lý SQL Server Login
- Quản lý database user
- Quản lý role
- Quản lý phân quyền
- Chọn chế độ xác thực

Quản lý SQL Server Logins

- **Tạo SQL Server Login**

CREATE LOGIN sinhvien WITH PASSWORD = '123456';

- **Xem thông tin**

Select * from SYS.SQL_LOGINS

⇔ điều kiện để thực thi các lệnh trên: user cần phải có quyền **ALTER ANY LOGIN** hay là thành viên của **securityadmin** fixed server role.

Quản lý SQL Server Logins

- **Đổi password**

ALTER LOGIN sinhvien WITH PASSWORD = ‘zugan%x’

- **Thay đổi tên login**

ALTER LOGIN sinhvien WITH NAME = student

- **Disabling / Enabling a login**

ALTER LOGIN sinhvien DISABLE | ENABLE;

- **Xóa SQL Server Login**

DROP LOGIN sinhvien

Quản lý database user

- Các loại user
- Các lệnh Create user/ Alter user / Drop user

Tạo và sử dụng Server Role

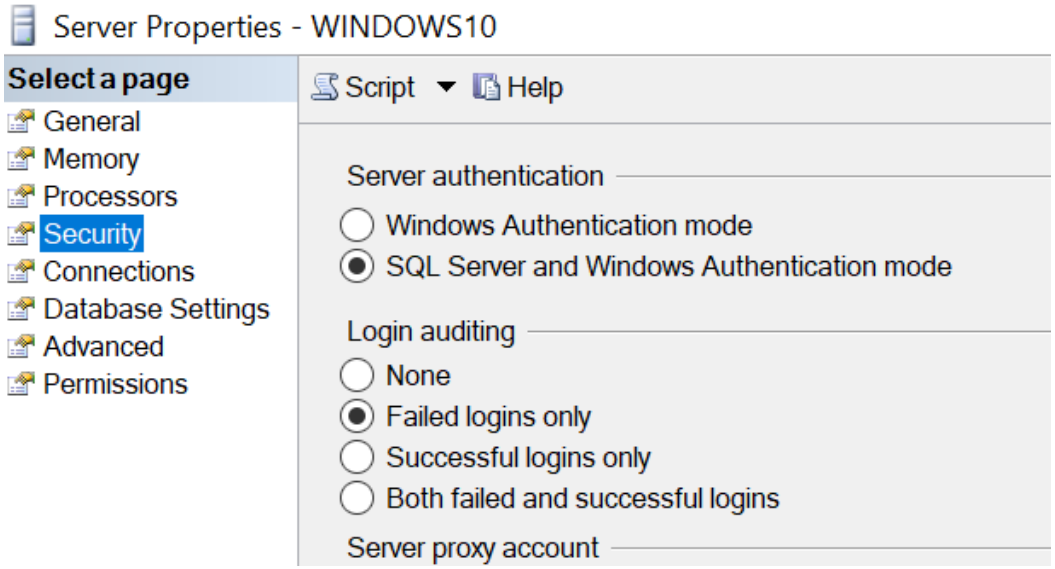
- Add/remove thành viên vào fixed server role
- Tạo một User-defined Server role
  - Add/remove permission vào role

Tạo và sử dụng database role

- Add/Remove thành viên vào fixed database role
- Tạo một User-defined Database role
  - Add/remove permission vào role

Thiết lập cấu hình

- Chọn chế độ xác thực :
  - Chọn server → Properties → chọn Security : trong mục Server Authentication : chọn Window mode hay Mix mode



Ví dụ 1

- **Dùng tài khoản sa**, tạo một SQL Server login '*sinhvien*' , thêm login này vào **dbcreator server role**.

```
CREATE  LOGIN  sinhvien  WITH  PASSWORD = '123‘
go
ALTER SERVER ROLE dbcreator  ADD MEMBER  sinhvien
```

- Login vào bằng tài khoản '*sinhvien*'. Thực hiện tạo database SV_db

```
CREATE DATABASE  SV_DB
```

=> Kiểm tra kết quả ? Cho biết owner của database ?

=> Tài khoản *sinhvien* có thể tạo table trong SV_DB không ?

Ví dụ 2

- Dùng tài khoản sa thực hiện :
  - Tạo một login ‘userNW’

```
CREATE LOGIN userNW WITH PASSWORD = '123‘,
                    DEFAULT_DATABASE = northwind
```

  - Tạo một user cho login trên để connect tới database Northwind

```
USE northwind
CREATE USER userNW FOR LOGIN userNW
```

  - Cấp quyền cho user trên database Northwind bằng cách thêm user vào ‘db_owner’ database role

```
EXEC sp_addrolemember 'db_owner' , 'userNW'
```

Ví dụ 2 (*tiếp theo*)

- Thực hiện connect vào server và truy suất Northwind qua login và user vừa tạo
- **'db_owner' database role** cho phép thành viên của role thực hiện bất kỳ thao tác nào trên database => Hãy kiểm tra với user 'userNW' ?
- Tài khoản userNW có quyền gì với các database khác ?

- Dùng tài khoản sa
  - Cho biết kết quả của lệnh sau ?

    DROP LOGIN userNW
  - Nếu không thực hiện lệnh DROP LOGIN userNW , mà thay bằng DROP USER userNW
    Hãy cho biết kết quả ?

Ví dụ 3.1

- Dùng tài khoản sa, tạo login ‘dbmaker’

CREATE LOGIN dbmaker WITH PASSWORD = '123‘

Go

ALTER SERVER ROLE dbcreator ADD MEMBER dbmaker

- Kết nối vào server bằng tài khoản dbmaker , thực hiện tạo một database

```
CREATE DATABASE xyz
```

- Tài khoản sa có thể thực hiện xóa login này không ? Giải thích lỗi và đưa ra cách để thực hiện lệnh thành công ?

```
DROP LOGIN dbmaker
```

Ví dụ 3.2

- **Thay đổi ownership** của database , sử dụng sa login thực hiện đoạn lệnh sau

```
USE xyz
EXEC sp_changedbowner 'sa'
sp_helpdb xyz
```

- Thay đổi ownership của object trong database ?

(sử dụng sp_changeobjectowner)

Ví dụ 4

- Dùng tài khoản sa, tạo login user1 và user tương ứng, có toàn quyền trên Northwind database

```
CREATE LOGIN user1 WITH PASSWORD = '123'
GO
USE northwind
CREATE USER user1 FOR LOGIN user1
GO
USE northwind
EXEC sp_addrolemember 'db_owner' , 'user1'
```

- Thực hiện tương tự để tạo login user2

Ví dụ 4 (*tiếp theo*)

- User1 và User2 cùng là thành viên của *db_owner* , có nhận xét gì về quyền của User1 và User2 với các object trong database
  - User1 tạo table A và User2 tạo table B trong Northwind . Xác định owner của 2 table này ?
  - User1 có thể xóa table B không ?

Ví dụ 5

- Dùng tài khoản sa , thực hiện các lệnh sau

```
USE  Nothwind
Go
CREATE  ROLE   TestDbRole
CREATE  ROLE    DevDbRole
Go
--- ‘chris’  là một database user của Northwind
EXEC sp_addrolemember  N’DevDbRole’,  N’chris’
Go
EXEC sp_addrolemember  N’DevDbRole’,  N’TestDbRole’

Go
EXEC sp_addrolemember  N’db_datareader’, N’DevDbRole’
```

Ví dụ 5 (*tiếp theo*)

- Cho biết role DevDbRole có thể có quyền gì trên Northwind ?
- Cho biết role TestDbRole có thể có quyền gì trên Northwind ?
- Cho biết user ‘chris’ có thể có quyền gì trên Northwind ?

Ví dụ 6 - lệnh GRANT

- **Dùng tài khoản sa , tạo database và một login và user**

```
Create database testDB
Go
Use testDB
Go
CREATE TABLE Testtable (ID int,  abc varchar(10) );
INSERT INTO Testtable VALUES (1, 'abcd');
INSERT INTO Testtable VALUES (2, 'efgh');

-- Set up a login and user
CREATE LOGIN  TestPer WITH PASSWORD = '123',
GO
USE  testDB
GO
CREATE USER  TestPer   FROM LOGIN TestPer ;
```

Ví dụ 6.1 - lệnh GRANT

- Cấp phát cho người dùng TestPer quyền thực thi các câu lệnh SELECT, INSERT và UPDATE trên bảng testtable

```
GRANT SELECT,INSERT,UPDATE
ON testtable
TO  TestPer
```

- Thu hồi quyền đã cấp

```
REVOKE SELECT,INSERT,UPDATE
ON testtable
FROM  TestPer
```

Ví dụ 6.2 - lệnh GRANT

- Cấp phát cho người dùng **TestPer** quyền thực thi câu lệnh SELECT trên bảng testtable ở **mức cột**

```
GRANT SELECT
ON testtable (ID)
TO TestPer
```

--- TestPer không có quyền thực thi lệnh select * from testtable

- Thu hồi quyền đã cấp

```
REVOKE SELECT
ON testtable(ID)
FROM TestPer
```

Ví dụ 6.3 - lệnh GRANT

- Cấp phát **tất cả** các quyền có thể thực hiện được trên đối tượng cơ sở dữ liệu

```
GRANT  ALL
ON testtable
TO TestPer
```

- Thu hồi quyền đã cấp

```
REVOKE  ALL
ON testtable
FROM TestPer
```

The ALL permission is deprecated and maintained only for compatibility.
It DOES NOT imply ALL permissions defined on the entity.

Ví dụ 6.4 - lệnh GRANT

- Cho phép người dùng TestPer quyền xem dữ liệu trên bảng testtable đồng thời có thể cấp tiếp quyền này cho người dùng khác

```
GRANT SELECT
ON testtable
TO TestPer WITH GRANT OPTION
```

- Thu hồi cả quyền đã cấp tiếp cho những người dùng khác

```
REVOKE SELECT
ON testtable
TO TestPer CASCADE
```

Ví dụ 6.5 - lệnh GRANT

- Sử dụng GRANT để cấp quyền thực thi câu lệnh. Những quyền có thể cấp phát bao gồm:
  - Tạo bảng: CREATE TABLE
  - Tạo khung nhìn: CREATE VIEW
  - Tạo thủ tục lưu trữ: CREATE PROCEDURE
  - Tạo hàm: CREATE FUNCTION

  …
- Cấp phát quyền tạo bảng , view, schema cho người dùng có tên là TestPer

GRANT CREATE TABLE, CREATE VIEW, CREATE SCHEMA

TO TestPer

GRANT

- WITH GRANT OPTION :
  - Người được cấp quyền có khả năng cấp tiếp quyền này cho user khác
- WITH GRANT OPTION AS *principal*
  - Người được cấp quyền có khả năng cấp tiếp quyền này cho user khác , và hệ thống ghi nhận quyền đc cấp bởi *principal*

For example, presume that user Mary is principal_id 12 and user Raul is principal 15. Mary executes
GRANT SELECT ON OBJECT::X TO Steven WITH GRANT OPTION AS Raul
Now the sys.database_permissions table will indicate that the grantor_prinicpal_id was 15 (Raul) even though the statement was actually executed by user 13 (Mary).

- Cấp quyền thao tác mức cột:

  GRANT …

  ON table(column, …)

  TO ….

GRANT UPDATE ON dbo.tbl_EmpMaster(SalesRegion) TO SalesUser

Ví dụ 7 – lệnh DENY

- 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.

Ví dụ 8.1 - GRANT & DENY

- **GRANT that directly overrides a DENY**

  Use testDB

  Go

  DENY SELECT TO TestPer;

  GRANT SELECT TO TestPer;

  ---- *testPer thực hiện lệnh Select trên testDB được ko?*

- **DENY overrides a GRANT**

  GRANT SELECT TO TestPer;

  DENY SELECT TO TestPer;

  ---- *testPer thực hiện lệnh Select trên testDB được ko?*

https://sqlstudies.com/2014/08/04/deny-vs-revoke/

Ví dụ 8.2 - GRANT & DENY

- Tạo một database role , và thêm user vào role

```
Use TestDB
Go
CREATE ROLE  GrantSelectRole;
GRANT SELECT TO  GrantSelectRole;
EXEC sp_addrolemember 'GrantSelectRole', ‘TestPer';
--- login in as TestPer and exec Select command ?-> OK
GO
DENY SELECT TO TestPer;
--- login in as TestPer and exec Select command  ?-> not OK
```

Ví dụ 8.3 - GRANT & DENY

- Thu hồi quyền đã GRANT hay DENY

  REVOKE SELECT TO TestPer;

  REVOKE SELECT TO GrantSelectRole;

- DENY qua role

  DENY SELECT TO GrantSelectRole;

  GRANT SELECT TO TestPer;

--- login in as TestPer and exec Select command ? -> not OK

Tóm tắt

Câu hỏi

- Phân biệt và cho ví dụ ?
  - Permission thuộc server level và database level
  - Principal thuộc server level và database level

Câu hỏi

(1)

```
CREATE LOGIN sv WITH PASSWORD = '123456';
ALTER SERVER ROLE sysadmin ADD MEMBER sv
USE northwind
CREATE USER sv FOR LOGIN sv
```

(2)

```
CREATE LOGIN test WITH PASSWORD = '123456';
USE northwind
CREATE USER test FOR LOGIN test
```

Cho biết quyền của user sv và test trên Northwind sau khi tập lệnh trên được thực thi ?