Chương 3: Quản trị SQL Server 2005

- Triển khai CSDL - Database Deployment
- Vấn đề tiềm ẩn trong việc triển khai CSDL
- Xác thực người sử dụng trên SQL Server 2005
- Quản lý nhóm, người sử dụng:
  - Thêm/xóa nhóm, người sử dụng.
  - Cấp phát quyền cho nhóm, người sử dụng.

Triển khai CSDL

- Triển khai CSDL bằng module viết trên .Net – dùng cho các developers
- Triển khai CSDL bằng công cụ của SQL Server 2005 – dùng cho các SQL Server specialists:
  - Detach and reattach the database in Transact-SQL
  - Attach and detach the database in Management Studio

Detach and Reattach the Database in Transact-SQL

```
use master
--Detach
EXEC sp_detach_db 'Asset5'
Go
--Attach
EXEC sp_attach_db @dbname = 'Asset5',
        @filename1 = 'c:\Program Files\Microsoft SQL
    Server\MSSQL.1\mssql\data\Asset5.mdf',
        @filename2 = 'c:\Program Files\Microsoft SQL
    Server\MSSQL.1\mssql\data\Asset5_log.ldf‘
```

Nếu CSDL còn có nhiều file khác thì chúng ta thêm vào các @filename3, @filename4,…

Attach and Detach in Management Studio

- Detach:
  - Open the context-sensitive menu of the database and select Tasks | Detach.
  - The program will open the Detach Database window that shows if the database is ready for the operation.
  - Việc Detach CSDL có thể chưa sẵn sàng nếu như có ít nhất một user khác đang kết nối đến CSDL này.
- Attach:
  - Copy data and log files to a data folder on the target server.
  - Open the context-sensitive menu of the Databases node in the Object Browser and choose Attach.
  - The program will open the Attach window. Click the Add button and browse for the data file (.mdf) of your database. This will automatically load all the remaining files that are part of the database

Attach and Detach in Management Studio

- Xuất hiện lỗi sau khi attach hoặc restore một CSDL được lấy từ một server khác.
  - Cannot add diagram to SQL Server 2005 DB: Database diagram support objects cannot be installed because this database does not have a valid owner .
- Sửa lỗi này:
  - **EXEC sp_dbcmptlevel 'dbname', '90';**
  - **ALTER AUTHORIZATION ON DATABASE::dbname TO valid_login**
- **Ví dụ**
  - EXEC sp_dbcmptlevel 'Northwind', '90';
  - ALTER AUTHORIZATION ON DATABASE::Northwind TO "HUNG-CNPM\ManhHung“
- **60** = SQL Server 6.0; **65** = SQL Server 6.5
- **70** = SQL Server 7.0; **80** = SQL Server 2000
- **90** = SQL Server 2005

Vấn đề tiềm ẩn trong việc triển khai CSDL

- Khi triển khai CSDL như trên sẽ không đảm bảo được liên kết giữa server logins and database users. Vì:
  - Server logins được lưu trữ trong CSDL Master, Database users được lưu trữ trong từng CSDL.
  - Các Database users sẽ được attach theo CSDL, nhưng các Database users này lại liên kết đến server logins trên server cũ. => Lỗi.
- Giải pháp:
  - Sử dụng sp_change_users_login

Thủ tục sp_change_users_login

Cú pháp: sp_change_users_login [@Action =] 'action'
[, [@UserNamePattern =] 'user']
[, [@LoginName =] 'login'] [, [@Password =] 'password']

Action Value	Description
Auto_Fix	Links a user entry in the **sysusers** table in the current database to a SQL Server login of the same name. If a login with the same name does not exist, one will be created. Nếu login không tồn tại thì bạn phải chỉ định cả *user* and *password*. Nếu login đã có thì phải chỉ định user và không được chỉ định password. login phải là NULL, user phải đúng là có trong CSDL, login phải chưa được mapped đến một user nào khác.
Report	Liệt kê danh sách các user trong CSDL hiện thời không có liên kết đến server logins. Khi đó các tham số: *user*, *login*, and *password* must be NULL or not specified.
Update_One	Liên kết user đến một server login. Khi đó: *user* and *login* must be specified. *password* must be NULL or not specified.

Thủ tục sp_change_users_login

- Hiển thị tất cả các user không có liên kết đến server logins của CSDL hiện thời:

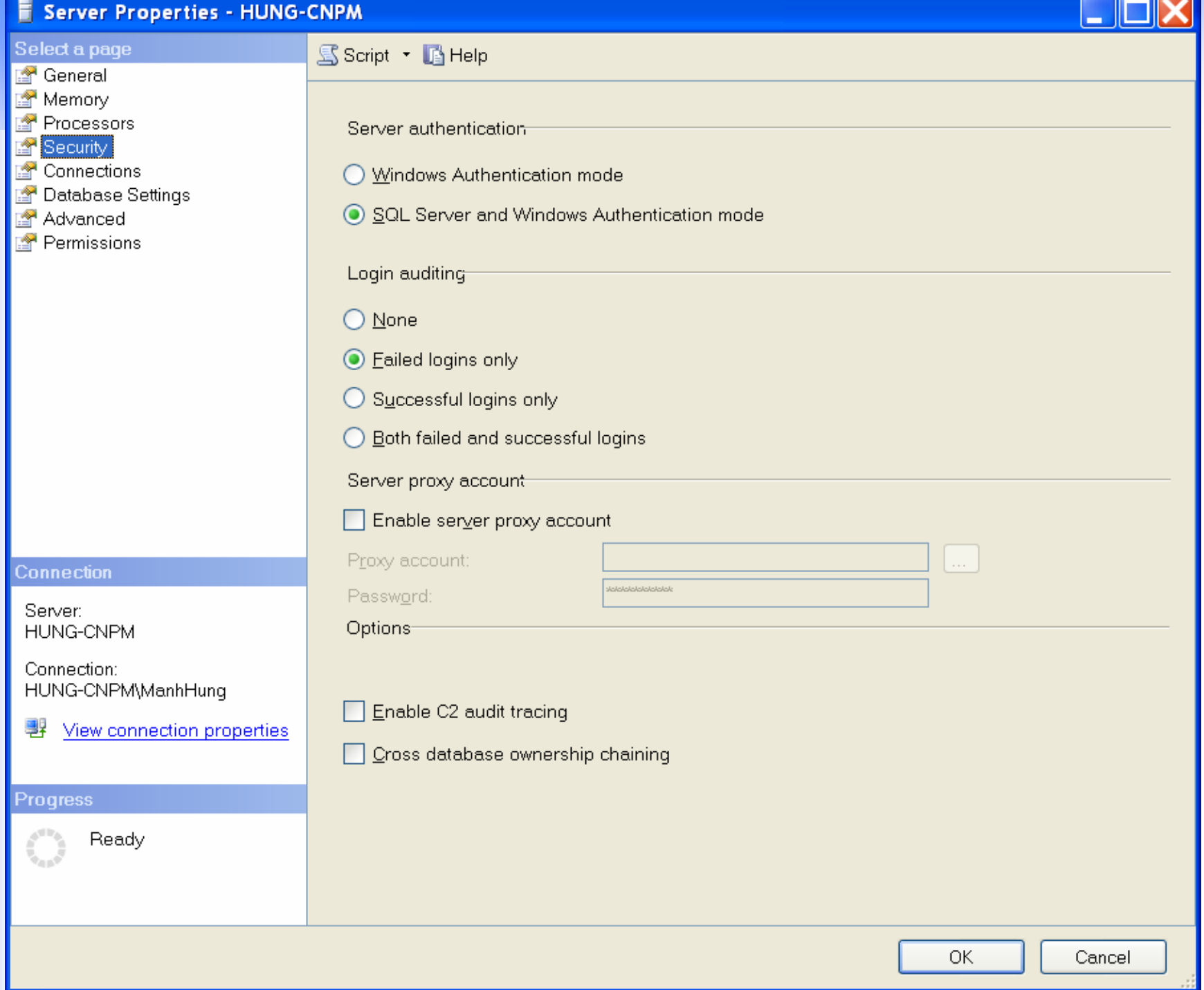
```
exec sp_change_users_login @Action = 'Report'
```

- Liên kết user chỉ định trong @UserNamePattern đến một server login:

```
exec sp_change_users_login @Action = 'Update_one',
                           @UserNamePattern = 'test',
                           @LoginName = 'hung‘
 --login chưa có phải chỉ định cả user and password.
exec sp_change_users_login @Action = 'Auto_Fix',
                           @UserNamePattern = 'nsunderic' ,
                           @password = 'myl.password'
```

Xác thực NSD

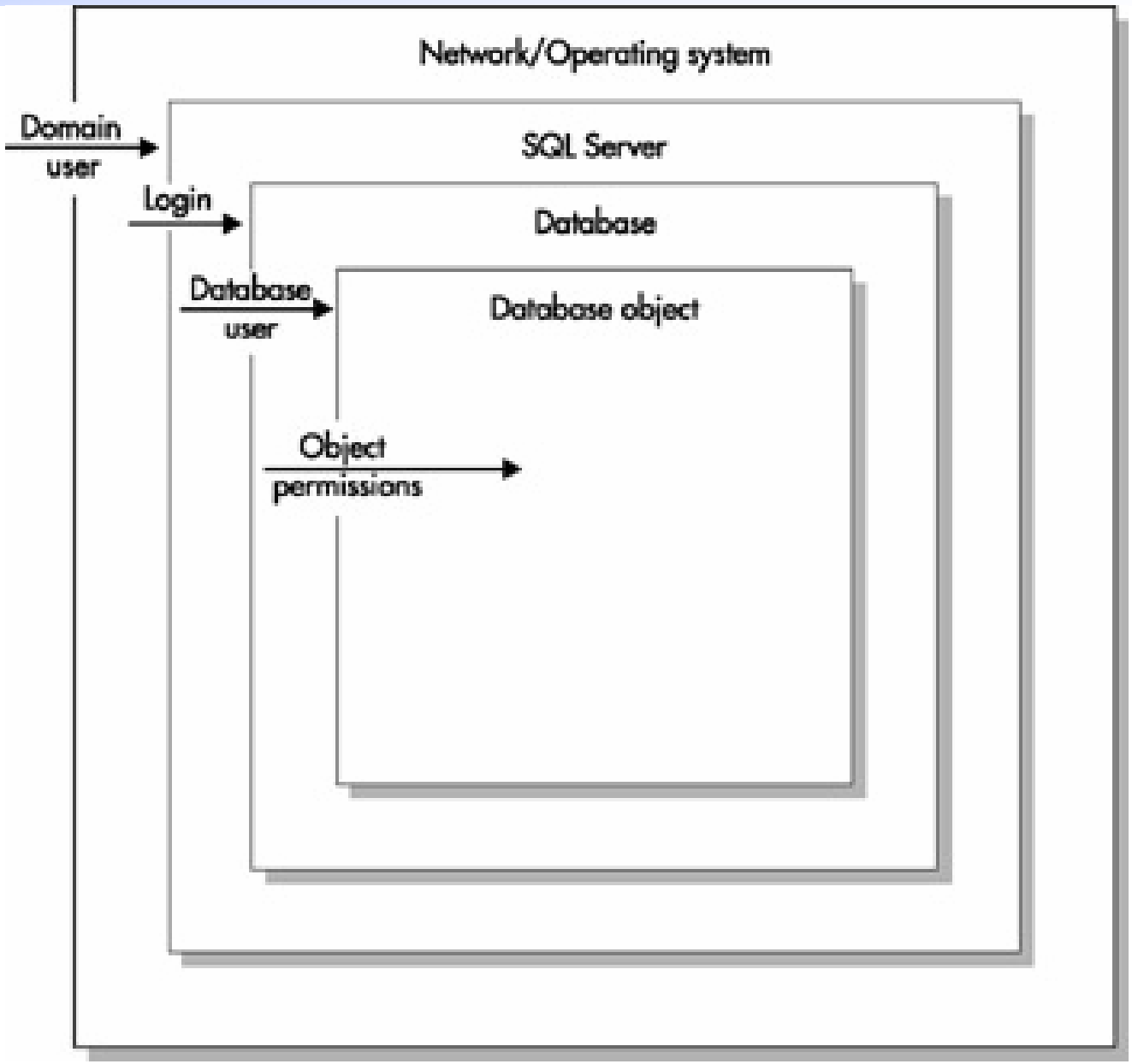
- Các kiểu xác thực:
  - SQL Server and Windows Authentication: hỗ trợ 2 kiểu đăng nhập trên SQL Server và trên Windows
  - Windows Authentication
- Khi cài đặt chúng ta đã chọn một kiểu xác thực cho SQL Server. Tuy nhiên chúng ta có thể thay đổi:
  - Mở MSt.
  - Trong cửa sổ Object Explorer, ấn phải chuột lên server, chọn properties.
  - Chọn nút Security => chọn kiểu xác thực
  - Chọn OK


Server Properties - HUNG-CNPM
Select a page
General
Memory
Processors
Security
Connections
Database Settings
Advanced
Permissions
Script
Help
Server authentication
Windows Authentication mode
SQL Server and Windows Authentication mode
Login auditing
None
Failed logins only
Successful logins only
Both failed and successful logins
Server proxy account
Enable server proxy account
Proxy account:
Password:
Options
Enable C2 audit tracing
Cross database ownership chaining
Connection
Server:
HUNG-CNPM
Connection:
HUNG-CNPM\ManhHung
View connection properties
Progress
Ready
OK
Cancel

So sánh 2 kiểu xác thực

- Kiểu Windows Authentication
  - Chỉ yêu cầu NSD đăng nhập một lần
  - Quản lý tập trung
  - Tận dụng được các tính năng bảo mật của Windows
- Kiểu SQL and Windows
  - Hỗ trợ NSD trên các HĐH khác
  - Quản lý NSD riêng

Access Levels



Tạo thêm NSD mới trên SQL Server

- Chú ý: Nếu sử dụng password policy thì mật khẩu phải tối thiểu là 6 kí tự và phải chứa cả ba loại: chữ thường a-z, chữ hoa A-Z, chữ số 0-9.
- Thêm NSD theo xác thực Windows
  - Mở Security mức Server
  - Ấn phải chuột chọn New Login
  - Lựa chọn: Windows Authentication
  - Nhập tên NSD, hoặc chọn Search
  - Chọn Default DB
  - Chọn Server Roles
  - Chọn User Mapping để chỉ ra các CSDL mà NSD có quyền tương tác.
- Thêm NSD theo xác thực SQL Server (làm tương tự)

Thêm NSD mới bằng T-SQL

```
CREATE LOGIN login_name { WITH <option_list1> | FROM <sources> }
<sources> ::=
    WINDOWS [ WITH <windows_options> [ ,... ] ]
    | CERTIFICATE certname
    | ASYMMETRIC KEY asym_key_name

<option_list1> ::=
    PASSWORD = 'password' [ HASHED ] [ MUST_CHANGE ]
    [ , <option_list2> [ ,... ] ]
<option_list2> ::=
    SID = sid
    | DEFAULT_DATABASE = database
    | DEFAULT_LANGUAGE = language
    | CHECK_EXPIRATION = { ON | OFF}
    | CHECK_POLICY = { ON | OFF}
    [ CREDENTIAL = credential_name ]
<windows_options> ::=
    DEFAULT_DATABASE = database
    | DEFAULT_LANGUAGE = language
```

Thêm NSD mới bằng T-SQL (2)

```
Create database test
go
use test
Go
-- Them login la UserLogin1 theo xac thuc SQL Server
CREATE LOGIN UserLogin1 WITH PASSWORD = '123'
--Them login la HUNG-CNPM\ManhHung lay tu user cua Windows
CREATE LOGIN [HUNG-CNPM\ManhHung] FROM
    WINDOWS;
```

Giấy ủy nhiệm - **CREDENTIAL**

- **Tạo giấy ủy nhiệm**
  - **CREATE CREDENTIAL *credential_name* WITH IDENTITY = '*identity_name*' [, SECRET = '*secret*']**

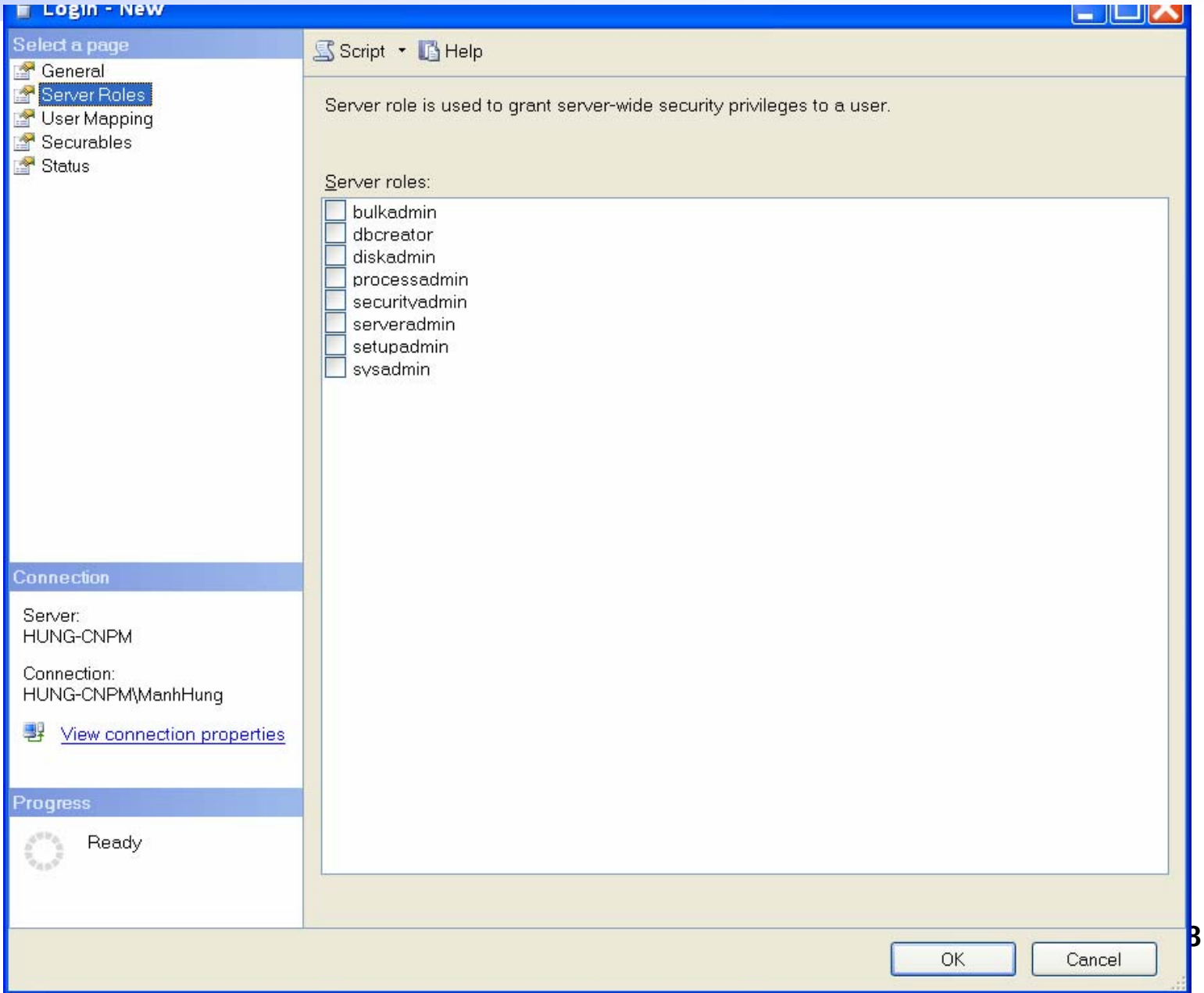
    - IDENTITY is a Windows user. IDENTITY được sử dụng khi từ SQL truy cập các tài nguyên bên ngoài.
    - SECRET là password cần xác thực khi từ SQL truy cập các tài nguyên bên ngoài.

Tạo nhóm mới

- Các Roles trên SQL Server giống như Groups trên Windows
- Trên SQL Server có 4 nhóm:
  - Server Roles: đã được xây dựng sẵn người dùng không thể thay đổi.
  - Database Roles: Định nghĩa các quyền trên CSDL của các nhóm.
  - Database Roles do người dùng định nghĩa
  - Nhóm Application Roles

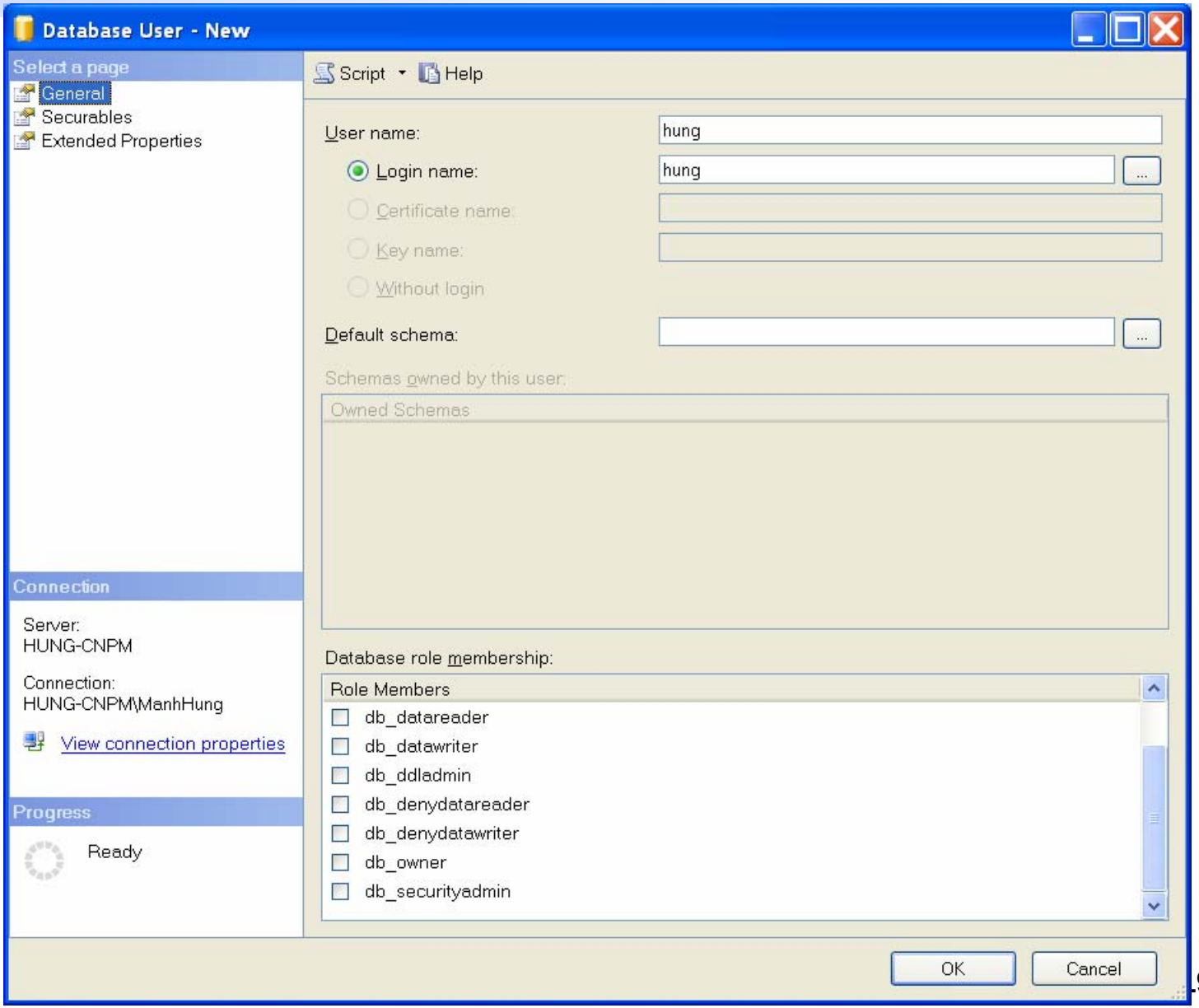
Thêm người sử dụng vào nhóm Server Roles

Chọn Security mức Server



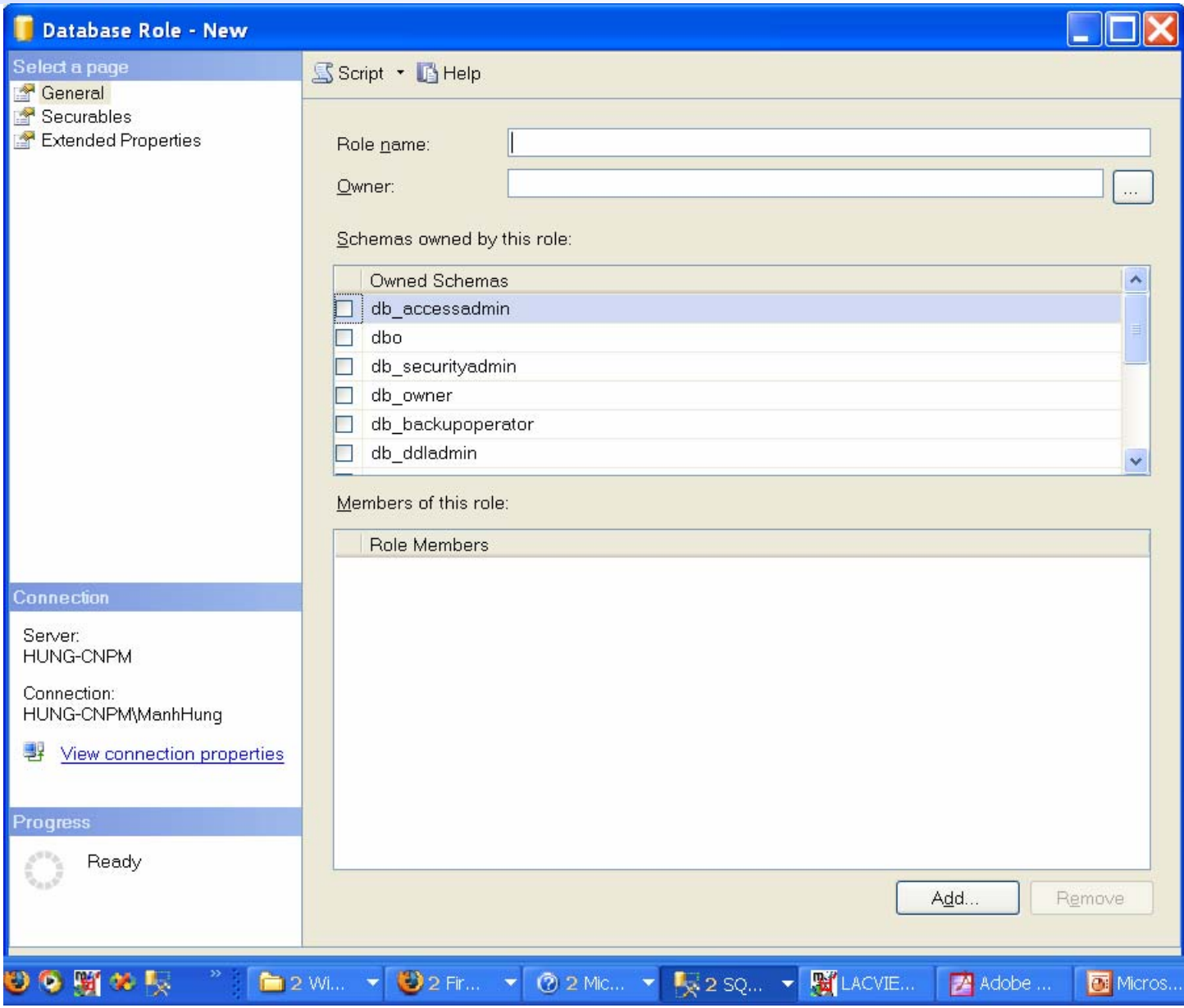
Thêm người sử dụng vào nhóm DB Roles

Chọn
- Security
- mức DB



Tạo nhóm Database Roles, Application Roles

- Mở Security mức DB, mở tiếp Roles.
- Kích phải chuột lên Database Roles -> chọn New



Fixed Server Roles

Fixed S-Role	Server-level Permission
bulkadmin	Granted: ADMINISTER BULK OPERATIONS
dbcreator	Granted: CREATE DATABASE
diskadmin	Granted: ALTER RESOURCES
processadmin	Granted: ALTER ANY CONNECTION, ALTER SERVER STATE
securityadmin	Granted: ALTER ANY LOGIN
serveradmin	Granted: ALTER ANY ENDPOINT, ALTER RESOURCES, ALTER SERVER STATE, ALTER SETTINGS, SHUTDOWN, VIEW SERVER STATE
setupadmin	Granted: ALTER ANY LINKED SERVER
sysadmin	CONTROL SERVER

Fixed Server Roles (2)

Vai trò (role)	Ý nghĩa
Sysadmin	Quản trị hệ thống (sa), có thể thực hiện mọi thao tác trên SQL Server
Securityadmin	-Thêm login vào nhóm này - Cấp phát/Cấm/Hủy bỏ quyền tạo CSDL (grant/deny/revoke CREATE DATABASE) -Quản lý người sử dụng của SQL Server (local), của các server liên kết, Remote server
Serveradmin	- Thêm login vào nhóm này - Có khả năng thay đổi các tham số của SQL Servver (RECONFIGURE, sp_configure) - SHUTDOWN
Setupadmin	- Thêm login vào nhóm này - Có thể thêm, xoá, cấu hình lại các server liên kết - Có thể thực hiện **sp_procoption để đánh dấu một thủ tục Startup** **(**sp_configure 'show advanced options', 1**)**

Fixed Server Roles (3)

Vai trò (role)	Ý nghĩa
Processadmin	- Thêm user vào nhóm này - Có thể kết thúc các tiến trình của người sử dụng, **KILL *spid***
Diskadmin	- Thêm user vào nhóm này - Thêm, xóa thiết bị lưu trữ: sp_addumpdevice, sp_dropdevice.
Dbcreator	- Thêm user vào nhóm này - Có thể tạo, sửa, xoá CSDL - Đổi tên CSDL (sp_renamedb)
Bulkadmin	- Thêm user vào nhóm này - Có thể thực hiện BULK INSERT

Fixed Database Roles

- **public Database Role**:tất cả các DB user đều kế thừa các quyền của public role.

Fixed Database Roles (2)

Vai trò (role)	Ý nghĩa
Db_owner	Sở hữu CSDL (dbo), có thể thực hiện mọi thao tác trên CSDL này
Db_accessadmin	Quản trị người sử : Thêm/ xóa người sử dụng trong CSDL này
Db_datareader	Có thể thực hiện select trên các bảng dữ liệu của người dùng khác trong CSDL.
Db_dataWriter	Có thể Insert, update, delete trên các bảng của người dùng khác trong CSDL
Db_ddladmin	Có thể thêm, xoá hoặc sửa đổi các đối tượng CSDL
Db_securityadmin	Quản lý các Roles và Members trong CSDL này
Db_backupoperator	Có thể thực hiện chức năng sao lưu (backup) dữ liệu
Db_denydareader	Không thể sử dụng phát biểu select trên tất cả các bảng trong CSDL
Db_denydawriter	Không thể thực hiện insert, update, delete trên tất cả các bảng trong CSDL

Server 2000 ID	Server 2005 ID	Allocated To
0	0	public
1	1	dbo
2	2	guest
3	3	INFORMATION_SCHEMA
4	4	SYSTEM_FUNCTION_SCHEMA -2000;sys-2005
5 - 16383	5 - 16383	Users, aliases, application roles
16384	16384	db_owner
16385	16385	db_accessadmin
16386	16386	db_securityadmin
16387	16387	db_ddladmin
16389	16389	db_backupoperator
16390	16390	db_datareader
16391	16391	db_datawriter
16392	16392	db_denydatareader
16393	16393	db_denydatawriter
16394 - 16399	16394 - 16399	Reserved
16400 - 32767		Roles
	16400 - 2,147,483,647	Users, roles, application roles, aliases

Tạo mới user database

```
CREATE USER user_name
   [ { { FOR | FROM }
     {
       LOGIN login_name
       | CERTIFICATE cert_name
       | ASYMMETRIC KEY asym_key_name
     }
     | WITHOUT LOGIN
   ]
   [ WITH DEFAULT_SCHEMA = schema_name ]
```

create user Userus1 for login Loginus1

Thêm user vào các nhóm bằng T-SQL

- Thêm vào Server Roles
  - Cú pháp:
    - sp_addsrvrolemember [@loginame=] '*login*' , [@rolename =] '*role*'
  - Ví dụ:
    - EXEC sp_addsrvrolemember 'user1', 'sysadmin';

Thêm vào các Database Roles

- Cú pháp:
  - sp_addrolemember [@rolename =] '*role*', [@membername =] '*security_account*'
- Ví dụ:
  - EXEC sp_addrolemember 'db_accessadmin', 'user1db';

Application roles

- App Roles được dùng cho việc bảo mật các ứng dụng riêng lẻ. App roles khác với DB roles ở các điểm sau:
  - App Roles yêu cầu password để kích hoạt.
  - App Roles không có members. Ứng dụng sẽ cung cấp tên của App Role và Password.
  - SQL Server ignores all other user permissions when the application role is activated.
  - SQL Server bỏ qua tất cả các quyền khác khi mà App Roles được kích hoạt

App Roles (2)

- To create an application role, administrators should use sp_addapprole:
  - Cú pháp:
    - **sp_addapprole** [**@rolename =**] '*role*' **,** [**@password =**] '*password*'
  - Ví dụ:
    - Exec sp_addapprole @rolename = 'Accounting', @password = 'password'

App Roles (3)

- A client application should first log in to SQL Server and then activate the application role using sp_setapprole:
  - Cú pháp:

```
sp_setapprole [ @rolename = ] 'role',
        [ @password = ] { encrypt N'password'  | 'password' }
        [ , [ @encrypt = ] { 'none' | 'odbc' } ]
        [ , [ @fCreateCookie = ] true | false ]
    [ , [ @cookie = ] @cookie OUTPUT ]
```

  - Ví dụ:
    - Exec sp_setapprole @rolename = 'Accounting', @password = 'password'

App Roles(4)

```
Use test go
Exec sp_addapprole @rolename = 'Approle', @password = '1111' go
SELECT USER_NAME();  go
DECLARE @cookie varbinary(8000);
EXEC sp_setapprole 'Approle ', '1111',
 @fCreateCookie = true, @cookie = @cookie OUTPUT;
-- The application role is now active.
SELECT USER_NAME();
-- This will return the name of the application role, Approle.
EXEC sp_unsetapprole @cookie;
-- The application role is no longer active. -- The original context has now been restored.
SELECT USER_NAME();
-- This will return the name of the original user.
```

Thủ tục sp_helpuser

- **A. Listing all users**
  - The following example lists all users in the current database: EXEC sp_helpuser
- **B. Listing information for a single user**
  - The following example lists information about the user database owner (dbo): EXEC sp_helpuser 'dbo'
- **C. Listing information for a database role**
  - The following example lists information about the db_securityadmin fixed database role.
  - EXEC sp_helpuser 'db_securityadmin'

Bảo mật ứng dụng bằng Stored Procedures (lecture3-grant.sql)

Khi user được cấp phát thực hiện trên các đối tượng như Stored Procedures, User-defined Functions, and Views thì không cần các quyền trên các đối tượng bên trong thủ thục, hàm, views (nếu chúng cùng schema).

```
Create Database Test
go
Create login Maria WITH PASSWORD = 'My,password',
    DEFAULT_DATABASE = Test
```

Bảo mật ứng dụng bằng Stored Procedures (lecture3-grant.sql)

```
Use Test
CREATE USER Maria
Create Table dbo.aTable (Id int identity(1,1),Description Varchar(20) )
Create Procedure dbo.ap_aTable_List
as    select user_name() 'User in proc'
     Select * from dbo.aTable --cung schema nen ko kiem tra quyen tren
    cac objects duoc tham chieu
go
Create Procedure dbo.ap_aTable_Insert
   @SDesc varchar(20)
as     Insert Into dbo.aTable (Description)     Values (@SDesc)
go
Deny Select, Insert, Update, Delete On dbo.aTable To Public
Grant Execute On dbo.ap_aTable_Insert To Public
Grant Execute On dbo.ap_aTable_List To Public
```

Bảo mật ứng dụng bằng Stored Procedures… (2)

- Login vào SQL Server 2005 theo user: Maria
- Hai lệnh sau sẽ không thực hiện được
  - select * from atable
  - insert into atable(Description) values ('test')

- Hai lệnh sau lại thực hiện tốt
  - exec ap_aTable_Insert 'test'
  - exec dbo.ap_aTable_List
  - Revert

Liệt kê các quyền của user

```
USE test  -- The database the user has permissions in.
--Set the session context to the user.
--SELECT * FROM fn_my_permissions (NULL, 'DATABASE');
go
EXECUTE AS User = 'Mary';
GO
-- Get the user's permissions on the current database
SELECT * FROM fn_my_permissions (NULL, 'DATABASE');
SELECT * FROM fn_my_permissions ('aTable', 'OBJECT');
GO
-- Set the session context back to you.
REVERT; (Lecture3-fn-my-permissions.sql)
```

Lệnh **EXECUTE AS**

- Trong SQL 2005 chúng ta có thể thiết lập user thông qua lệnh **EXECUTE AS**.
- Cú pháp:

  EXECUTE AS <context_specification>

  <context_specification>::=

  { LOGIN | USER } = 'name' [WITH {NO REVERT | COOKIE INTO @varbinary_variable }]

  | CALLER}
- Revert

  REVERT [WITH COOKIE = @varbinary_variable]
- Các ví dụ trong lecture3.doc (I. , II.)

Grant, Revoke, Deny

- Grant – là lệnh dùng để cấp pháp quyền thực thi các thao tác hoặc là quyền truy cập đến đối tượng trên SQL Server.
- Revoke – dùng để đòi lại các quyền mà user đã được cấp phát.
- Deny – cấm không cho thực thi các thao tác hoặc truy cập đến một đối tượng nào đó

Grants permissions on a schema

Cấp quyền đối với mỗi schema:

GRANT permission [,...n] ON SCHEMA :: schema_name

TO database_principal [,...n]

[WITH GRANT OPTION]

[AS granting_principal]

Ví dụ:

Grants permissions on objects (table, view, proc,…)

GRANT <permission> [,...n] ON

[OBJECT ::][schema_name]. object_name [(column [,...n])]

TO <database_principal> [,...n]

[WITH GRANT OPTION]

Grants permissions on a server

GRANT *permission* [,...n] TO <login> [,...n]

Ví dụ:

```
use master  go
create login LoginUs1 with password='123'
create login LoginUs2 with password='123’ go
grant create any database to LoginUs1
execute as login='LoginUs1'
create database test1
 revert
execute as login='LoginUs2'
create database test2 --fail
 revert
drop database test1
Drop login LoginUs1
```

Revokes permissions on a schema

- REVOKE [GRANT OPTION FOR] *permission* [,...n] ON SCHEMA **::** *schema_name* { TO | FROM } *database_principal* [,...n] [CASCADE]

- GRANT OPTION FOR: Đòi lại quyền WITH GRANT OPTION đã cấp phát
- CASCADE: Đòi lại các quyền đã phát.
  - Ví dụ:
    - user sa -> usr1 (**WITH GRANT OPTION**)->usr2
    - User sa đòi lại quyền của usr1(cascade) thì quyền truy cập của usr2 cũng tự động bị đòi lại

Removes server-level GRANT and DENY permissions

REVOKE [GRANT OPTION FOR]

permission [,...n] { TO | FROM } <login>

[,...n] [CASCADE]

Revokes permissions on objects

```
REVOKE [ GRANT OPTION FOR ] <permission> [ ,...n ] ON
   [ OBJECT :: ][ schema_name ]. object_name [ ( column [ ,...n ] ) ]
      { FROM | TO } <database_principal> [ ,...n ]
   [ CASCADE ]
```

Deny

- Denies permissions on a server:
  - DENY *permission* [**,**...n] TO <login> [**,**...n] [CASCADE]
- Denies permissions on a schema:
  - DENY *permission* [,...n] } ON SCHEMA **::** *schema_name* TO *database_principal* [,...n] [CASCADE]
- Denies permissions on objects:
  - DENY <permission> [**,**...n] ON [OBJECT **::**][*schema_name*]**.** *object_name* [**(** *column* [**,**...n] **)**] TO <database_principal> [**,**...n] [CASCADE]

Lệnh Create Schema

Trong SQL 2005 sử dụng lược đồ để gom nhóm các đối tượng phục vụ cho việc dễ dàng quản lý các CSDL lớn.

Người dùng trong SQL 2005 được gán một giản đồ mặc định.

Có thể cấp quyền hoặc cấm quyền truy cập đến lược đồ cho user hoặc group.

CREATE SCHEMA schema_name_clause [<schema_element> [, ...n]]

<schema_name_clause> ::=

{ schema_name

| AUTHORIZATION owner_name

| schema_name AUTHORIZATION owner_name

}

<schema_element> ::=

{ table_definition | view_definition | grant_statement

revoke_statement | deny_statement

}