Universidad César Vallejo

Pregrado

GESTIÓN DE DATOS E NFORMACIÓN II

30 años

Licenciada por SUNEDU para que puedas salir adelante

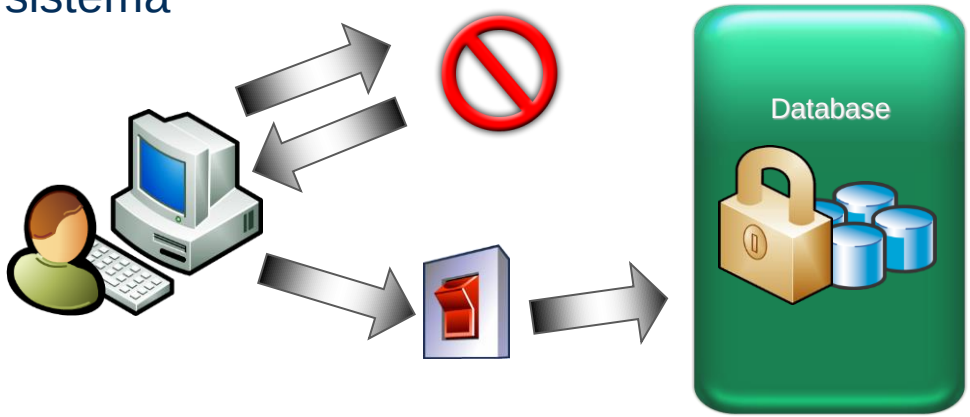
SESIÓN 11

SEGURIDAD DE BASE DE DATOS

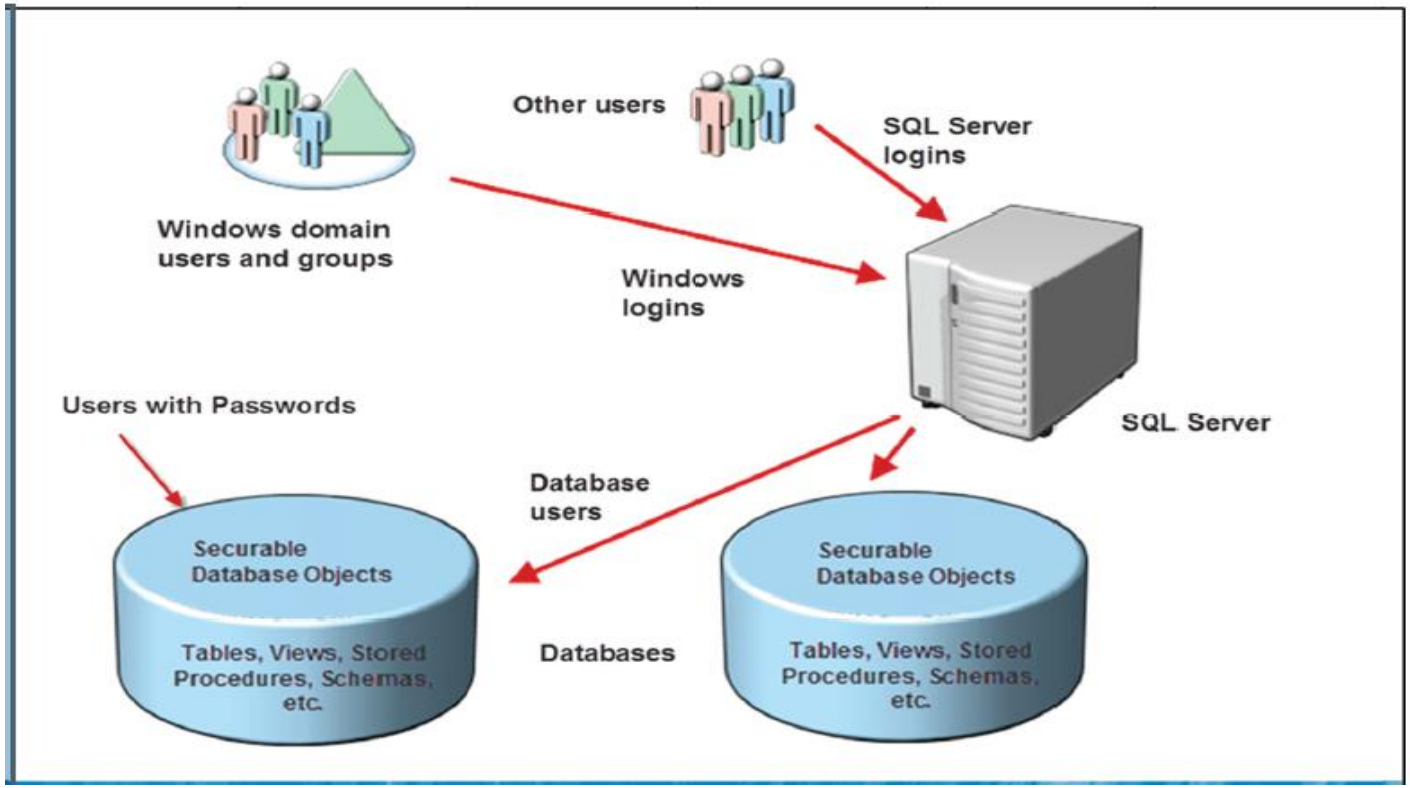
www.ucv.edu.pe

INTRODUCCIÓN A LA SEGURIDAD

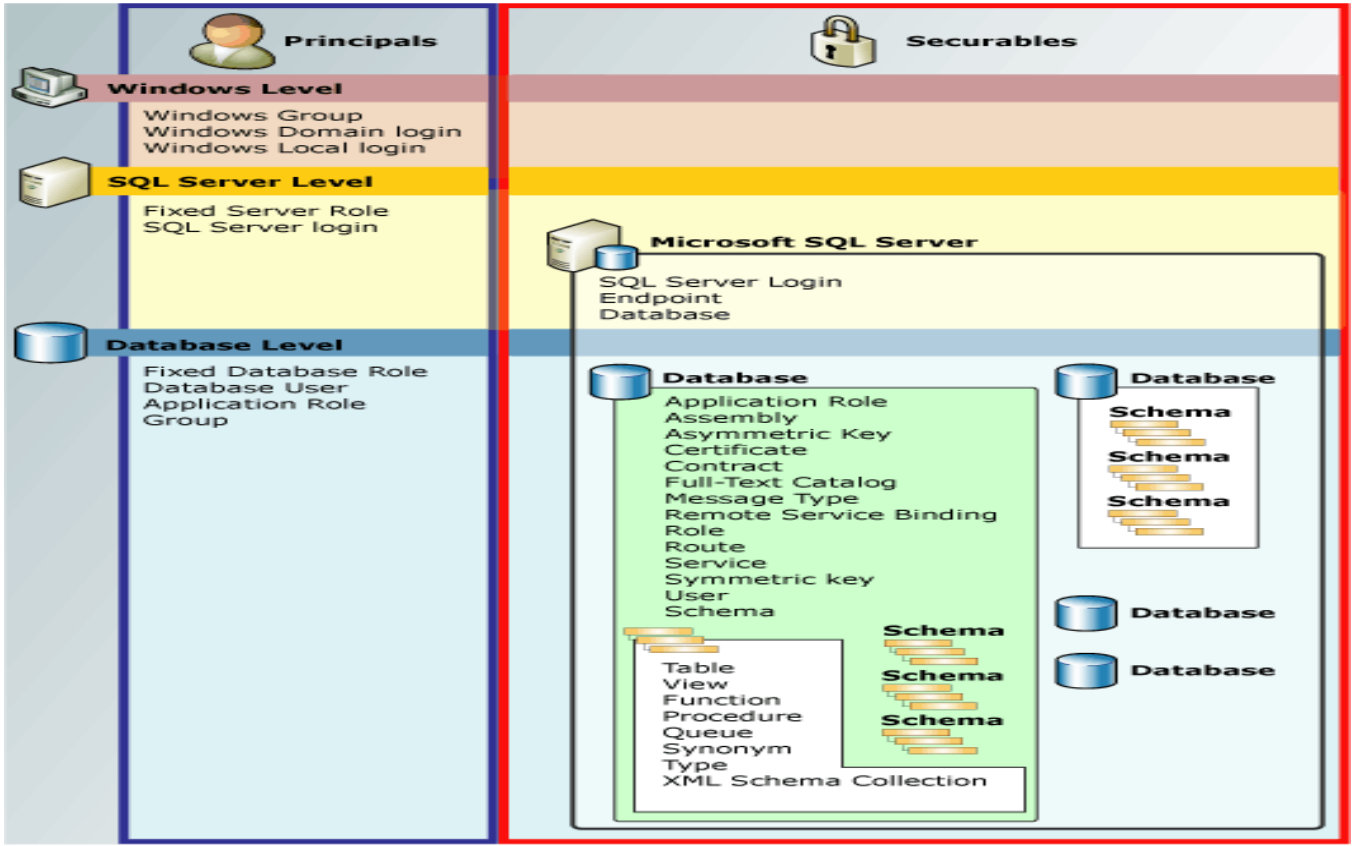
- Seguridad Implica que Personas Autorizadas (actores) usen los Recursos de la Base de Datos
- Los recursos deben ser usados en funcion a las roles que desempeñan los actores en el sistema



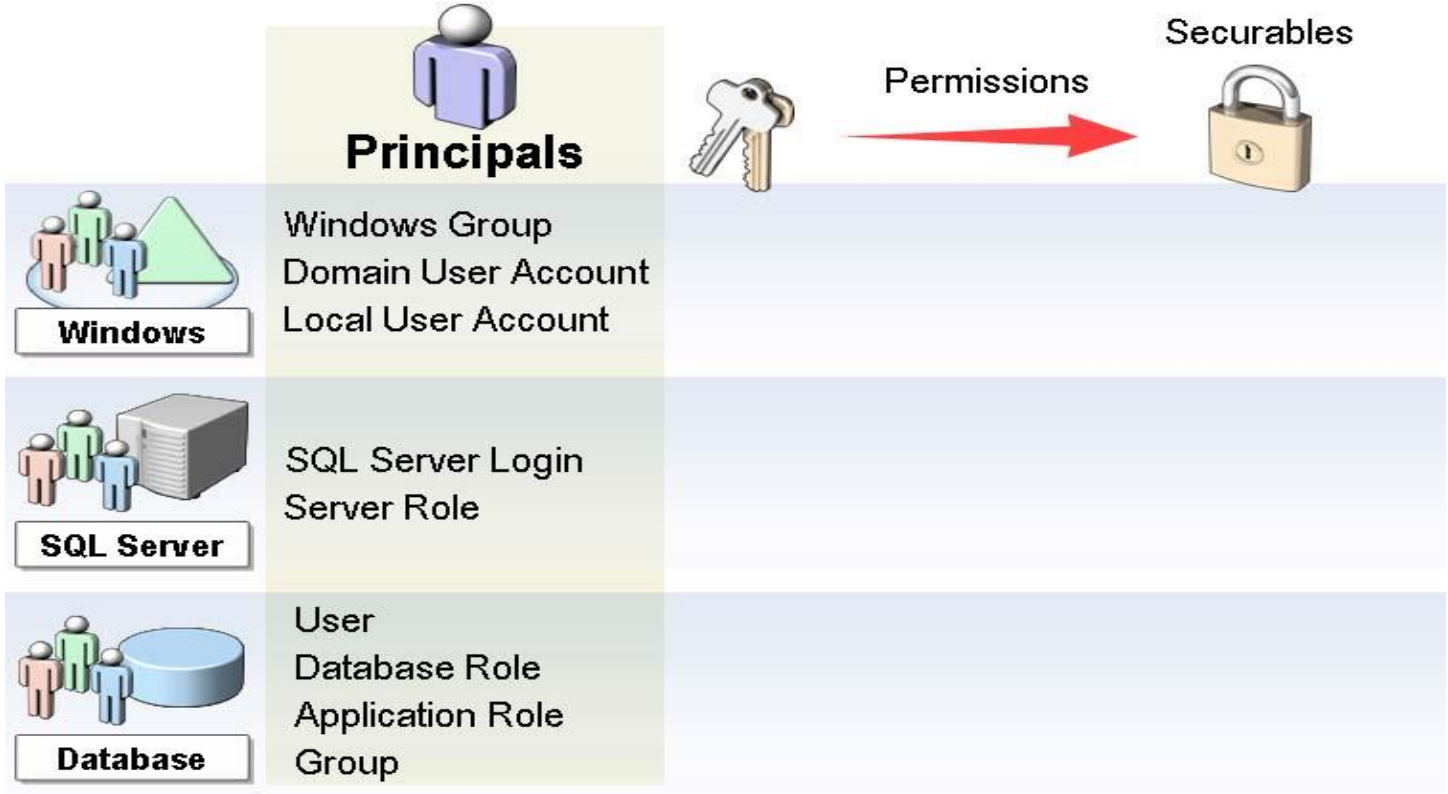
NIVELES DE SEGURIDAD



LA SEGURIDAD SQL SERVER



LA SEGURIDAD SQL SERVER



LA SEGURIDAD SQL SERVER

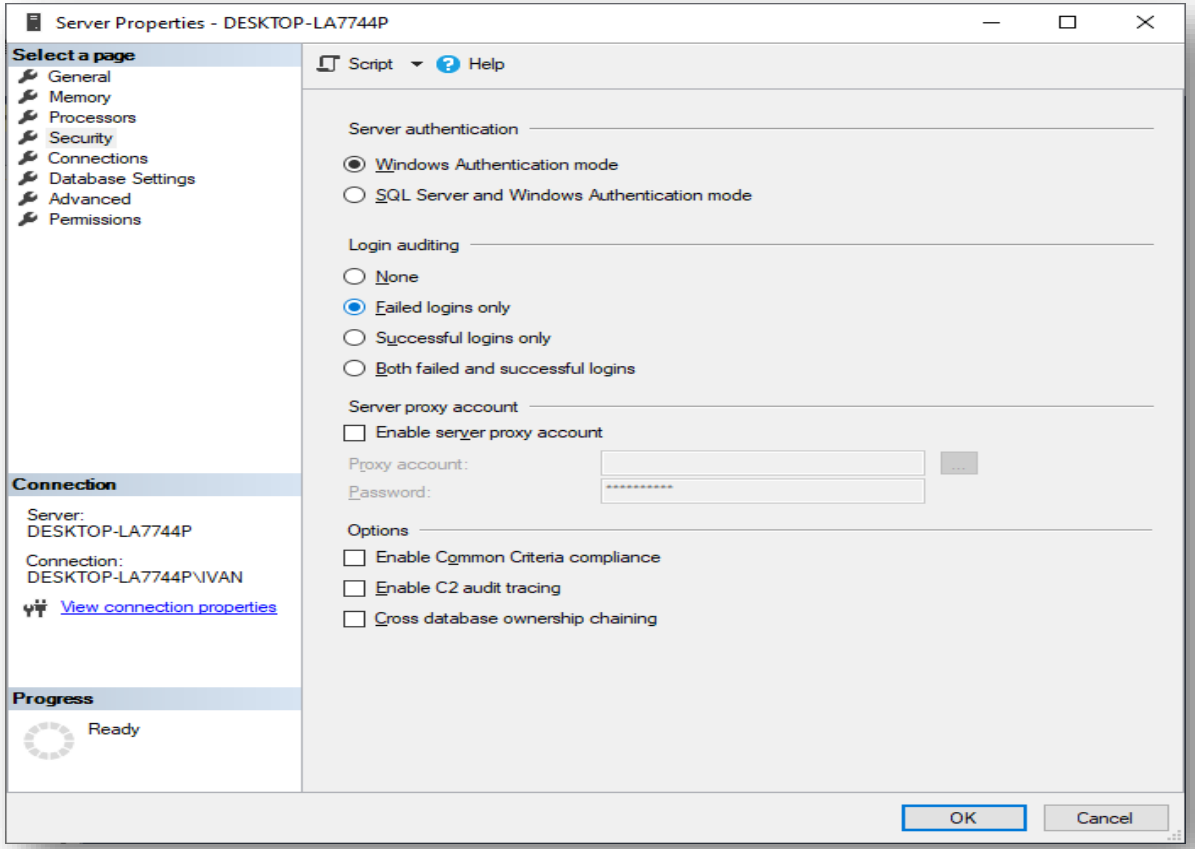
Nivel	Principal
Windows	Cuenta de usuario local Windows Cuenta de usuario de dominio Windows Grupo Windows
SQL Server	SQL Server login SQL Server role
Base de datos	Usuario de base de datos, Rol de base de datos Grupo de base de datos, Rol de la aplicación

LA SEGURIDAD SQL SERVER

- **Son recursos a los cuales el Sistema de Autorizacion Controla el Acceso**
- **Alcances:**

Alcance	Descripción
Servidor	Logins HTTP Encode Certificados
Base de Datos	Usuarios Roles Roles de Aplicacion Certificados, Assemblies
Schema	Tablas Vistas Funciones, Procedimientos Almacenados Reglas, Defaults

MODOS DE AUTENTICACIÓN



AUTENTICACIÓN DE WINDOWS

- ✓ Cuando un usuario se conecta a través de una cuenta de usuario de **Microsoft Windows**, **SQL Server** valida el nombre de cuenta y la contraseña con el token de la entidad de seguridad de Windows del sistema operativo. Esto significa que Windows confirma la identidad del usuario. **SQL Server** no pide la contraseña y no realiza la validación de identidad.

- ✓ ***La autenticación de Windows*** es el modo de autenticación predeterminado y es mucho más seguro que la autenticación de SQL Server .

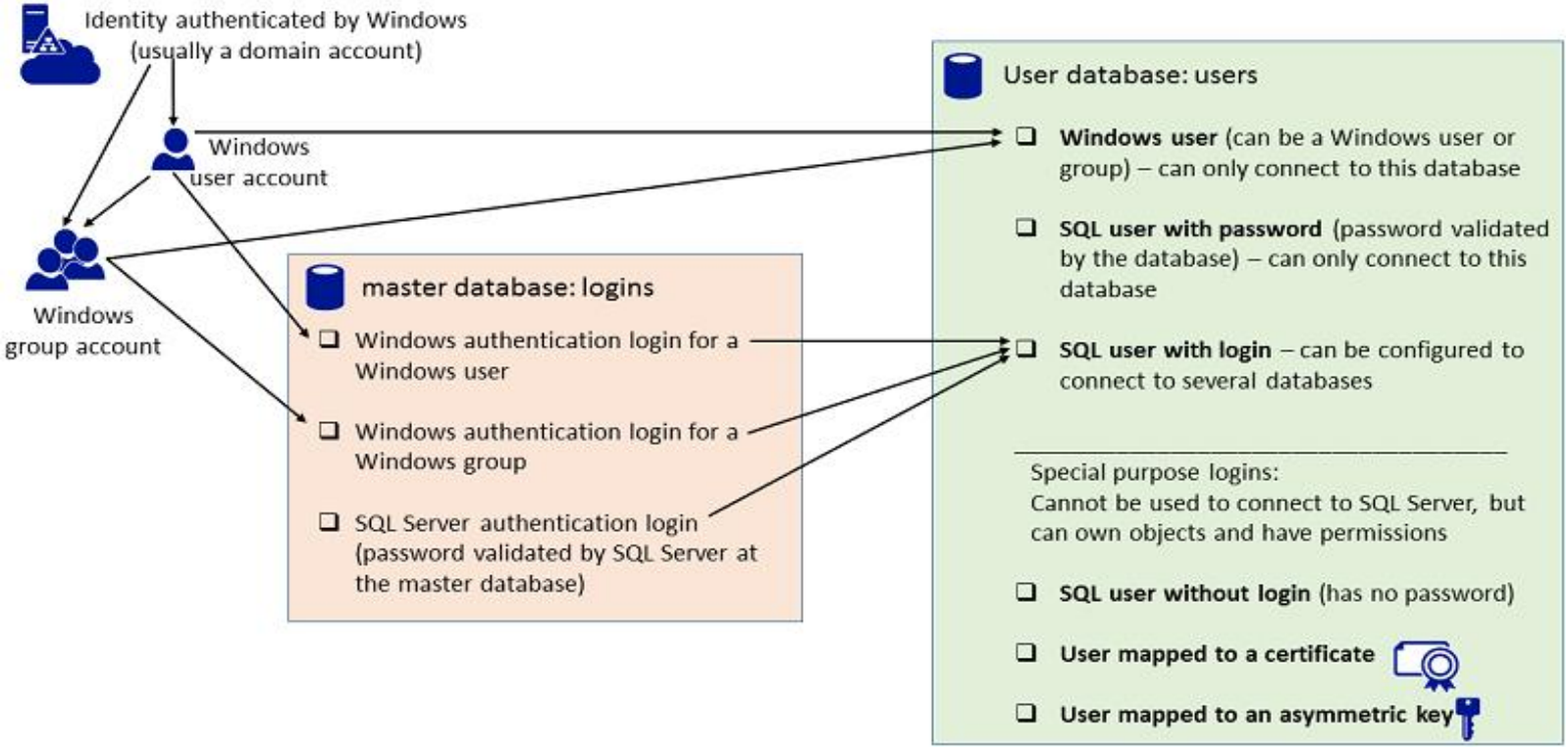
- ✓ ***La autenticación de Windows*** usa el protocolo de seguridad de Kerberos, proporciona la aplicación de directivas de contraseñas en cuanto a la validación de la complejidad de las contraseñas seguras, ofrece compatibilidad para el bloqueo de cuentas y admite la expiración de las contraseñas.

AUTENTICACIÓN DE WINDOWS

- ✓ Una conexión realizada utilizando **la autenticación de Windows** se denomina a veces conexión de confianza, porque SQL Server confía en las credenciales proporcionadas por Windows.

- ✓ Cuando se emplea **la autenticación de Windows**, se pueden crear grupos de Windows en el nivel de dominio y se puede crear un inicio de sesión en SQL Server para todo el grupo. La administración del acceso desde el nivel de dominio puede simplificar la administración de cuentas.

AUTENTICACIÓN DE WINDOWS



AUTENTICACIÓN SQL SERVER

- ✓ Cuando se utiliza la ***autenticación de SQL Server*** , los inicios de sesión se crean en SQL Server y no se basan en cuentas de usuario de Windows. El nombre de usuario y la contraseña se crean utilizando SQL Server y se almacenan en SQL Server.

- ✓ Los usuarios que se conectan usando la autenticación de SQL Server deben indicar sus credenciales (inicio de sesión y contraseña) cada vez que se conectan. Al utilizar la autenticación de SQL Server , debe establecer contraseñas seguras para todas las cuentas de SQL Server .

- ✓ Microsoft SQL Server crea una cuenta de administrador por defecto denominada SA. Esta cuenta dispone de privilegios de administrador completos así como de propiedad de tablas de sistema.

ROLES DE NIVEL SERVIDOR

SQL Server proporciona roles de nivel de servidor para ayudarle a administrar los permisos de un servidor. Estos roles son entidades de seguridad que agrupan otras entidades de seguridad. Los roles de nivel de servidor se aplican a todo el servidor en lo que respecta a su ámbito de permisos.

ROLES DE NIVEL SERVIDOR

sysadmin Los miembros del rol fijo de servidor **sysadmin** pueden realizar cualquier actividad en el servidor.

serveradmin Los miembros del rol fijo de servidor **serveradmin** pueden cambiar opciones de configuración en el servidor y cerrar el servidor.

processadmin Los miembros del rol fijo de servidor **processadmin** pueden finalizar los procesos que se ejecutan en una instancia de SQL Server.

setupadmin Los miembros del rol fijo de servidor **setupadmin** pueden agregar y quitar servidores vinculados mediante instrucciones de Transact-SQL. (Es necesaria la pertenencia a **sysadmin** cuando se usa Management Studio).

ROLES DE NIVEL SERVIDOR

Bulkadmin Los miembros del rol fijo de servidor **bulkadmin** pueden ejecutar la instrucción BULK INSERT.

Diskadmin El rol fijo de servidor **diskadmin** se usa para administrar archivos de disco.

Dbcreator Los miembros del rol fijo de servidor **dbcreator** pueden crear, modificar, quitar y restaurar cualquier base de datos.

ROLES DE NIVEL SERVIDOR

Securityadmin Los miembros del rol fijo de servidor securityadmin administran los inicios de sesión y sus propiedades. Pueden administrar los permisos de nivel de servidor **GRANT, DENY, y REVOKE**. También pueden administrar los permisos de nivel de base de datos **GRANT, DENY y REVOKE** si tienen acceso a una base de datos. Asimismo, pueden restablecer contraseñas para inicios de sesión de SQL Server .

IMPORTANTE: La capacidad de conceder acceso a Motor de base de datos y configurar los permisos de usuario permite que el administrador de seguridad asigne la mayoría de los permisos de servidor. El rol **securityadmin** se debe tratar como equivalente al rol **sysadmin** .

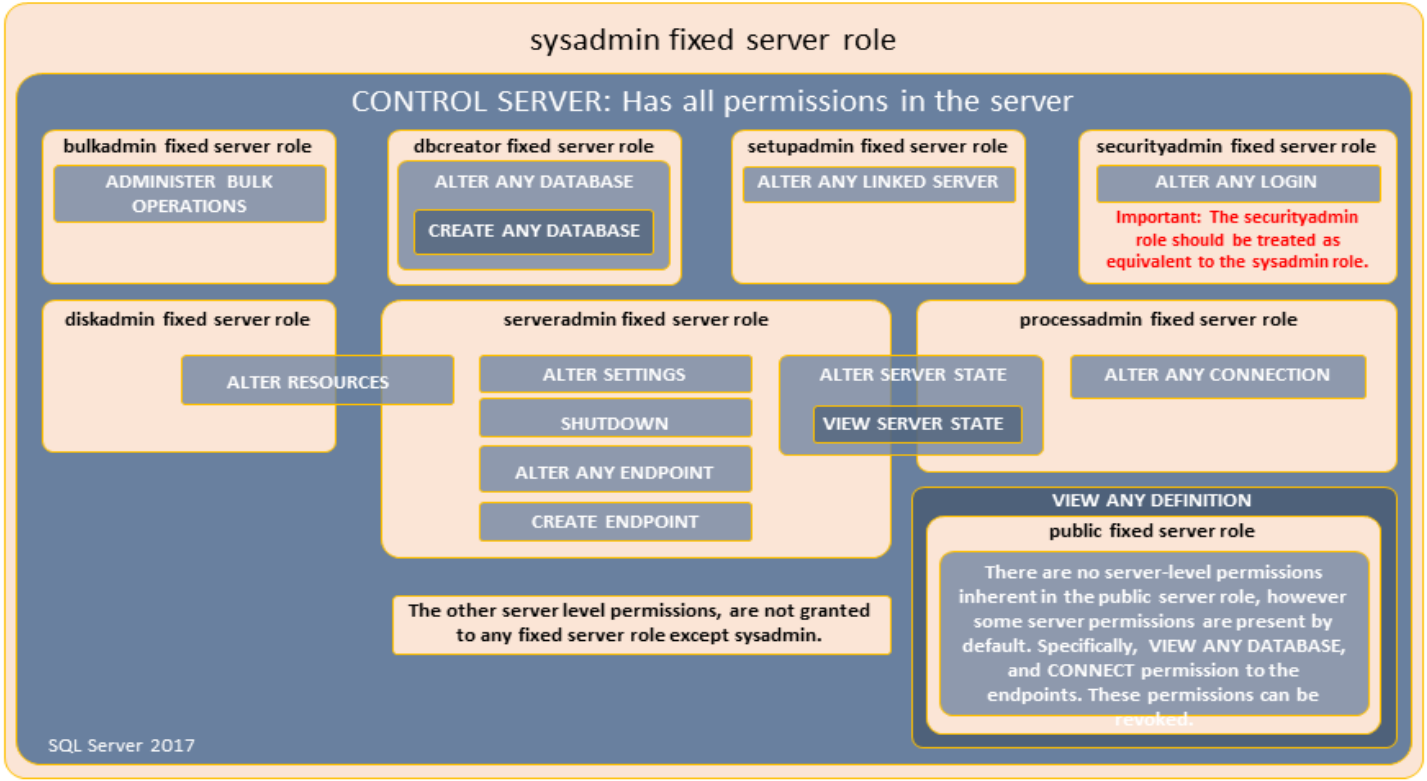
ROLES DE NIVEL SERVIDOR

Public Cada inicio de sesión de SQL Server pertenece al rol de servidor **public.** Cuando a una entidad de seguridad de servidor no se le han concedido ni denegado permisos específicos para un objeto protegible, el usuario hereda los permisos concedidos al rol pública para ese elemento. Solo asigne los permisos públicos en cualquier objeto cuando desee que el objeto esté disponible para todos los usuarios. No puede cambiar la pertenencia en **public.**

Nota: public se implementa de manera diferente a otros roles, y los permisos se pueden conceder, denegar o revocar desde los roles fijos de servidor públicos.

ROLES DE NIVEL SERVIDOR

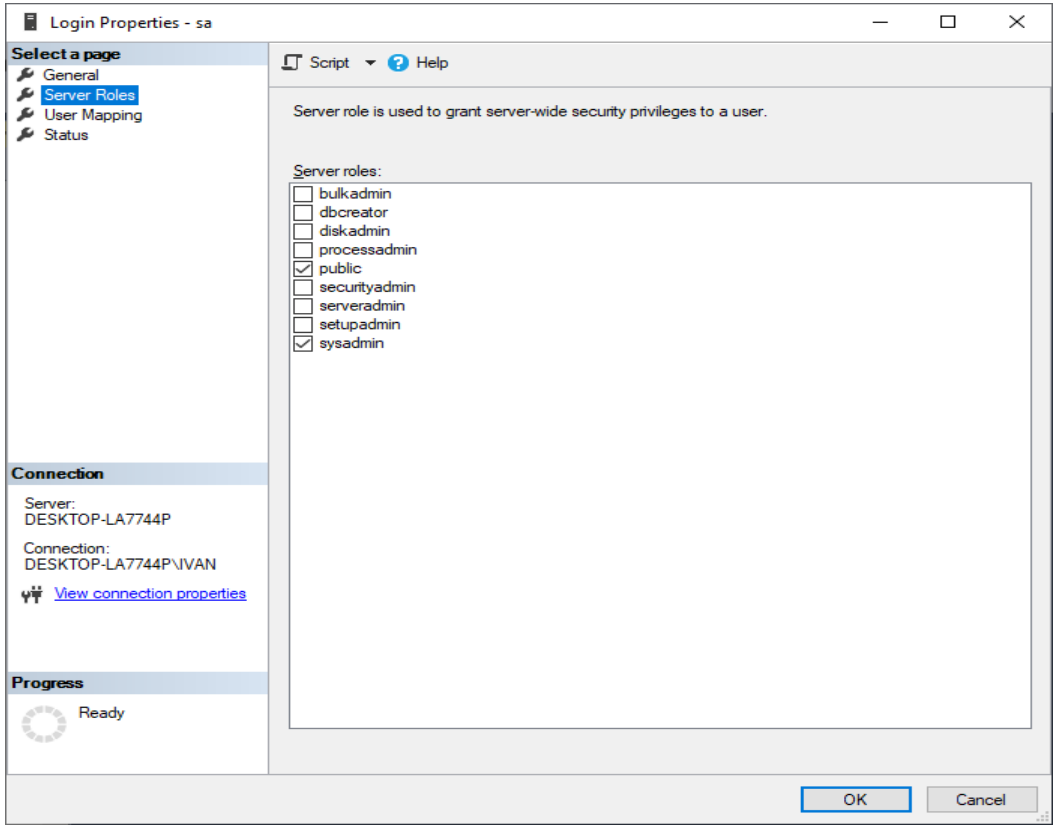
SERVER LEVEL ROLES AND PERMISSIONS: 9 fixed server roles, 34 server permissions



ROLES DE NIVEL SERVIDOR

Pregrado

Ingeniería de Sistemas



PERMISOS A NIVEL SERVIDOR

- Permite accesar objetos mismos de servidor, logins y base de datos

Asegurable	Permiso	Descripción
Server	CONNECT_SQL	Conecta al servidor.
	CREATE LOGIN	Crea un login.
	ALTER ANY LOGIN	Altera cualquier login en el rango del servidor.
	CONTROL SERVER	Control completo de administración del sistema.
Login	ALTER	Altera el login.
	IMPERSONATE	Impersonar el login.
Base de Datos	CREATE TABLE	Crea una tabla en la base de datos.
	ALTER ANY USER	Altera cualquier usuario en la base de datos.
	CONTROL	Control completo de la base de datos.

ROLES DE NIVEL DE BASE DE DATOS

Para administrar con facilidad los permisos en las bases de datos, SQL Server proporciona varios roles , que son las entidades de seguridad que agrupan a otras entidades de seguridad. Son como los grupos del sistema operativo Microsoft Windows. Los roles de nivel de base de datos se aplican a toda la base de datos en lo que respecta a su ámbito de permisos.

ROLES FIJOS DE BASE DE DATOS

db_owner Los miembros del rol fijo de base de datos **db_owner** pueden realizar todas las actividades de configuración y mantenimiento en la base de datos y también pueden quitar la base de datos en SQL Server. (En SQL Database y Azure Synapse, algunas actividades de mantenimiento requieren permisos de nivel de servidor y los roles **db_owners** no las pueden realizar).

db_securityadmin Los miembros del rol fijo de base de datos **db_securityadmin** pueden modificar la pertenencia a roles únicamente para roles personalizados y administrar permisos. Los miembros de este rol pueden elevar potencialmente sus privilegios y se deben supervisar sus acciones.

db_accessadmin Los miembros del rol fijo de base de datos db_accessadmin pueden agregar o quitar el acceso a la base de datos para inicios de sesión de Windows, grupos de Windows e inicios de sesión de SQL Server .

ROLES FIJOS DE BASE DE DATOS

db_backupoperator Los miembros del rol fijo de base de datos **db_backupoperator** pueden crear copias de seguridad de la base de datos.

db_ddladmin Los miembros del rol fijo de base de datos db_ddladmin pueden ejecutar cualquier comando del lenguaje de definición de datos (DDL) en una base de datos.

db_datawriter Los miembros del rol fijo de base de datos db_datawriter pueden agregar, eliminar o cambiar datos en todas las tablas de usuario.

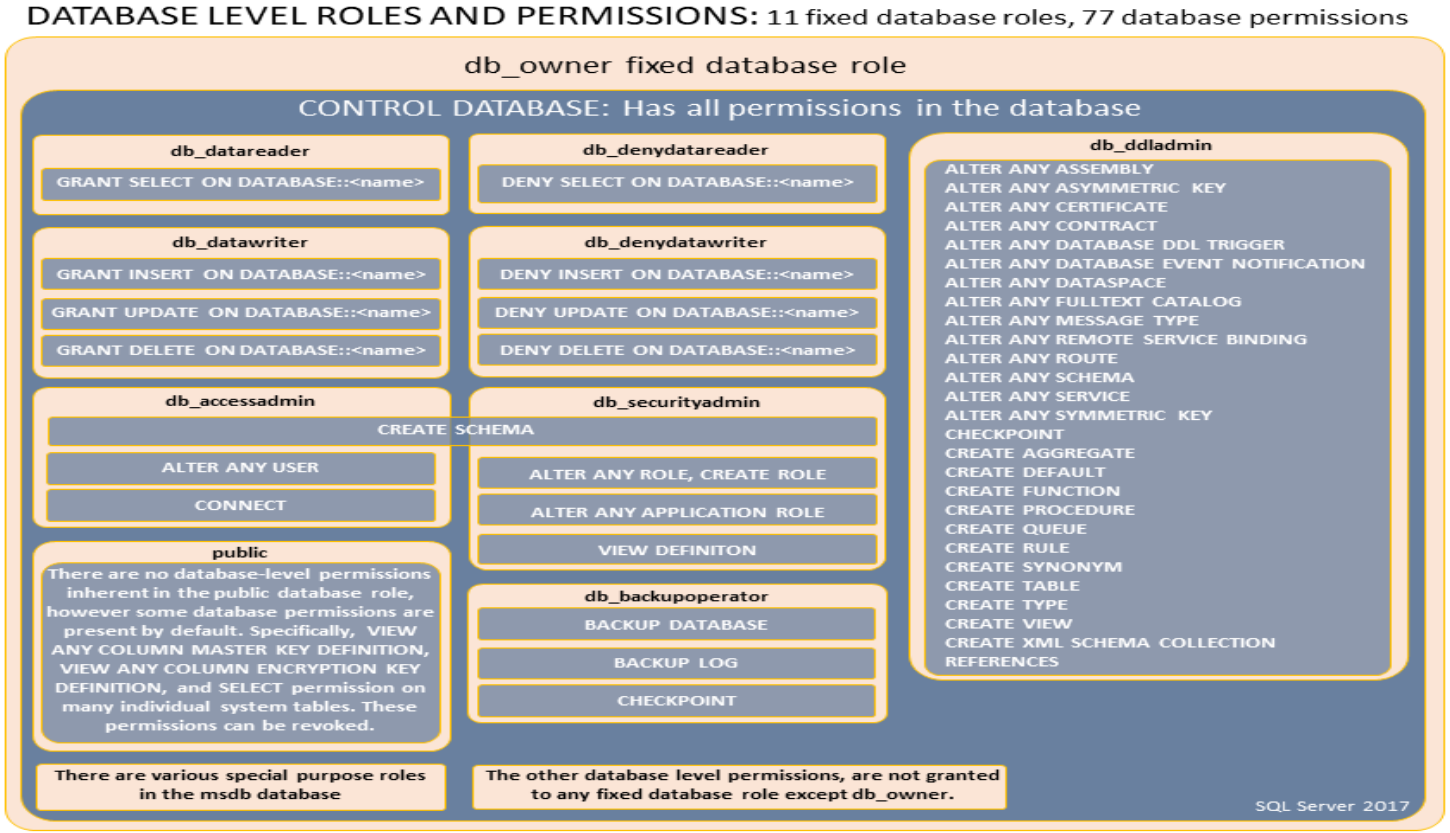
db_datareader Los miembros del rol fijo de base de datos db_datareader pueden leer todos los datos de todas las tablas y vistas de usuario. Los objetos de usuario pueden existir en cualquier esquema, excepto sys e INFORMATION_SCHEMA.

ROLES FIJOS DE BASE DE DATOS

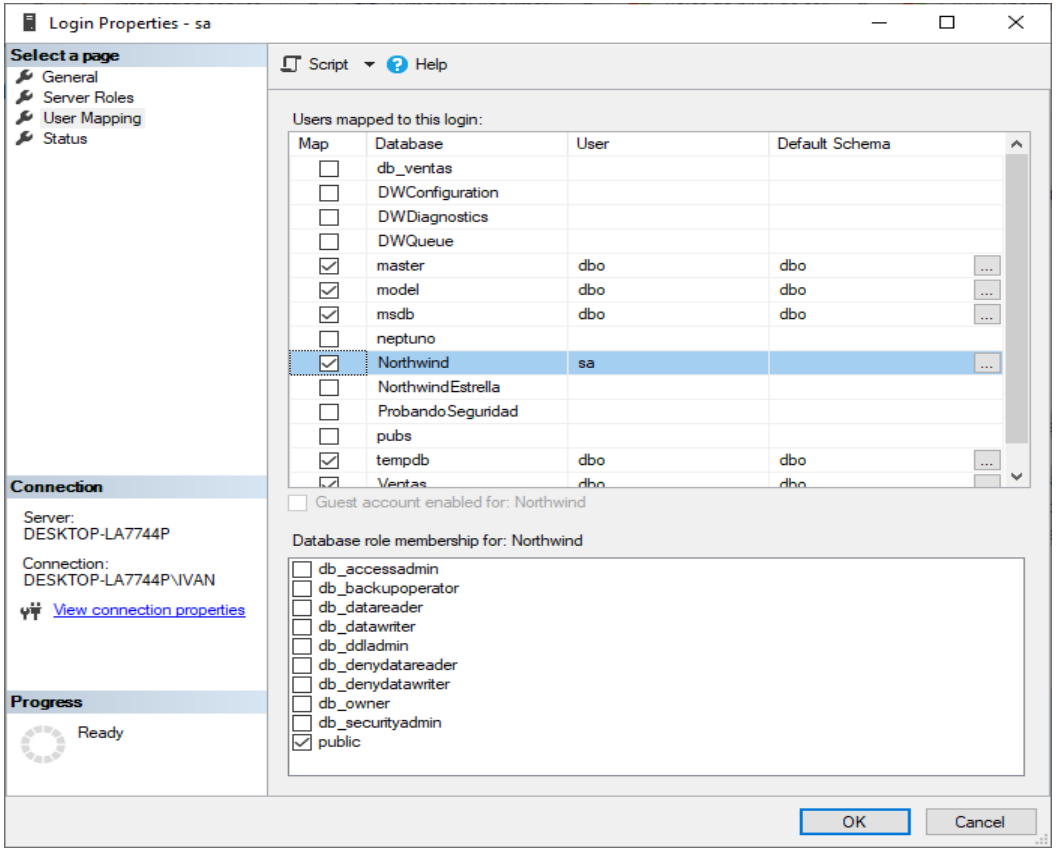
db_denydatawriter Los miembros del rol fijo de base de datos db_denydatawriter no pueden agregar, modificar ni eliminar datos de tablas de usuario de una base de datos.

db_denydatareader Los miembros del rol fijo de base de datos **db_denydatareader** no pueden leer datos de las tablas y vistas de usuario dentro de una base de datos.

ROLES FIJOS DE BASE DE DATOS



ROLES FIJOS DE BASE DE DATOS



PERMISOS A NIVEL DE BASE DE DATOS

Permite accesar objetos en la misma BD, tablas, SP

Asegurable	Permiso	Descripción
Usuario	ALTER	Altera el usuario especificado.
Schema	SELECT	Selecciona rows para cualquier objeto del schema.
	ALTER	Altera cualquier objeto en el schema.
	TAKE OWNERSHIP	Toma propiedad del schema.
Tabla	SELECT	Selecciona rows de la tabla.
	ALTER	Altera la tabla.
	INSERT	Agregar registros a la tabla
	UPDATE	Actualizar registros
	DELETE	Eliminar registros
	CONTROL	Control total de la tabla.
Stored Procedures	EXECUTE	Permite ejecutar un Stored Procedure

CUENTA DE USUARIO dbo

El **dbo** o base de datos owner, es una cuenta de usuario que tiene permisos implícitos para realizar todas las actividades en la base de datos. Los miembros del **sysadmin** rol de servidor fijo se asignan automáticamente a dbo.

La **dbo** cuenta de usuario se confunde con frecuencia con el **db_owner** rol fijo de la base de datos. El alcance de **db_owner** es una base de datos; el alcance de **sysadmin** es todo el servidor. La pertenencia al **db_owner** rol no confiere **dbo** privilegios de usuario.

CUENTA DE USUARIO dbo

El **dbo** o base de datos owner, es una cuenta de usuario que tiene permisos implícitos para realizar todas las actividades en la base de datos. Los miembros del **sysadmin** rol de servidor fijo se asignan automáticamente a dbo.

La **dbo** cuenta de usuario se confunde con frecuencia con el **db_owner** rol fijo de la base de datos. El alcance de **db_owner** es una base de datos; el alcance de **sysadmin** es todo el servidor. La pertenencia al **db_owner** rol no confiere **dbo** privilegios de usuario.

LA SEGURIDAD SQL SERVER - PERMISOS

Asegurable	Permiso	Descripción
Server	CONNECT_SQL	Conecta al servidor.
	CREATE LOGIN	Crea un login.
	ALTER ANY LOGIN	Altera cualquier login en el rango del servidor.
	CONTROL SERVER	Control completo de administración del sistema.
Login	ALTER	Altera el login.
	IMPERSONATE	Impersonar el login.
Base de Datos	CREATE TABLE	Crea una tabla en la base de datos.
	ALTER ANY USER	Altera cualquier usuario en la base de datos.
	CONTROL	Control completo de la base de datos.
Usuario	ALTER	Altera el usuario especificado.
Schema	SELECT	Selecciona rows para cualquier objeto del schema.
	ALTER	Altera cualquier objeto en el schema.
	TAKE OWNERSHIP	Toma propiedad del schema.
Tabla	SELECT	Selecciona rows de la tabla.
	ALTER	Altera la tabla.
	CONTROL	Control total de la tabla.

EJEMPLOS

CREACIÓN DE LOGIN

```
CREATE LOGIN [LoginFinanciera] WITH PASSWORD = N'Protect2020'
GO
CREATE LOGIN [LoginFinanciera] WITH PASSWORD = N'Protect2020',
DEFAULT_DATABASE=[FinancieraDemoBD]
-- creación de login desde Windows
CREATE LOGIN [LoginFinanciera] FROM WINDOWS WITH
DEFAULT_DATABASE=[FinancieraDemoBD]
```

ASIGNACIÓN DE ROLES A LOGIN

```
-- Agregar rol dbcreator al login
EXEC sp_addsrvrolemember 'LoginFinanciera', 'dbcreator'
```

EJEMPLOS

CREACIÓN DE USUARIO DE BASE DE DATOS

```
USE FinancieraDemoBD
GO
CREATE USER [UsuarioBDFinanciera] FOR LOGIN [LoginFinanciera]
GO
-- conceder permisos al usuario para que realice Select sobre una tabla de la BD.
GRANT SELECT ON dbo.CLIENTE TO UsuarioBDFinanciera
GO
```

ELIMINAR LOGIN Y USUARIO DE BASE DE DATOS

```
-- Eliminar Login y Usuario
DROP LOGIN LoginFinanciera
DROP USER UsuarioBDFinanciera
```

EJEMPLOS

CREACIÓN DE LOGIN Y USUARIO DE BASE DE DATOS, PREVIA VERIFICACIÓN DE EXISTENCIA

```
-- CREACION DE LOGIN
USE [master]
GO
IF NOT EXISTS (SELECT name FROM sys.server_principals WHERE name =
N'LoginFinanciera')
CREATE LOGIN [LoginFinanciera] WITH PASSWORD = N'Protect2020',
DEFAULT_DATABASE=[FinancieraDemoBD]
-- CREACION DE USUARIO DE BASE DE DATOS
USE [FinancieraDemoBD]
GO
IF NOT EXISTS(SELECT name FROM sys.sysusers WHERE name = 'UsuarioBDFinanciera')
CREATE USER [UsuarioBDFinanciera] FOR LOGIN [LoginFinanciera]
GO
```

EJEMPLOS

ASIGNACIÓN DE ROLES A USUARIO DE BASE DE DATOS Y OTORGAR PERMISOS A OBJETOS: PROCEDIMIENTOS ALMACENADOS, TABLAS

```
-- ASIGNACION DE PERMISOS
EXEC sp_addrolemember N'db_datareader', N'UsuarioBDFinanciera'
GO
EXEC sp_addrolemember N'db_datawriter', N'UsuarioBDFinanciera'
GO
GRANT EXECUTE TO [UsuarioBDFinanciera]
--DAR PERMISOS AL USUARIO DE BASE DE DATOS SOBRE PROCEDIMIENTOS ALMACENADOS Y TABLAS
USE [FinancieraDemoBD]
GO
GRANT EXECUTE ON [spListar_Clientes] TO [UsuarioBDFinanciera];
GRANT EXECUTE ON [spInsertar_Cliente] TO [UsuarioBDFinanciera];
GRANT EXECUTE ON [spListar_Proveedor] TO [UsuarioBDFinanciera];
GRANT EXECUTE ON [spInsertar_Proveedor] TO [UsuarioBDFinanciera];
GRANT SELECT,INSERT,UPDATE,DELETE ON [CLIENTE] TO [UsuarioBDFinanciera];
GRANT SELECT,INSERT,UPDATE,DELETE ON [PROVEEDOR] TO [UsuarioBDFinanciera];
GRANT SELECT,INSERT,UPDATE,DELETE ON "dbo"."COLABORADOR" TO "UsuarioBDFinanciera";
-- PERMISO ESPECIAL --> EVALUAR CONSIDERACIONES DE SEGURIDAD
GRANT ALTER ON "dbo"."CLIENTE" TO "UsuarioBDFinanciera";
GO
```

EJEMPLOS

CONSULTAR LOGIN Y USUARIOS DE BASE DE DATOS

```
-- Ver los inicios de Sesion
select * from sys.server_principals

-- *** Ver los Inicios de Sesión de la Instancia de SQL Server ***
select sid, name, dbname, password, loginname
from master..syslogins

-- *** Ver los Usuarios de la Base de Datos actual de SQL Server ***
use FinancieraDemoBD
GO
select uid, name, sid, *
from sysusers
where islogin=1

select * from sys.database_principals
```

¿QUÉ HEMOS APRENDIDO HOY?

Para que reflexionen y entiendan la importancia de los temas tratados y el mejoramiento de su propio proceso de aprendizaje.

Universidad César Vallejo

Licenciada por Sunedu
para que puedas salir adelante