

## Contenido

Modos de Seguridad en SQL Server .....	2
Comprender los modos de seguridad:.....	2
1. Autenticación de Windows .....	2
2. Autenticación de SQL Server (mixta): .....	3
3. Roles de Servidor:.....	4
4. Usuarios y Esquemas de Base de Datos: .....	4
5. Permisos a Nivel de Base de Datos:.....	5
6. Seguridad Avanzada en SQL Server 2022: .....	5
7. Auditoría de Seguridad: .....	6

---

## Manual de Seguridad SQL Server

SQL Server 2005 (y posteriores) proporciona múltiples capas de seguridad, integrándose estrechamente con Windows y ofreciendo configuraciones detalladas para administrar usuarios y permisos.

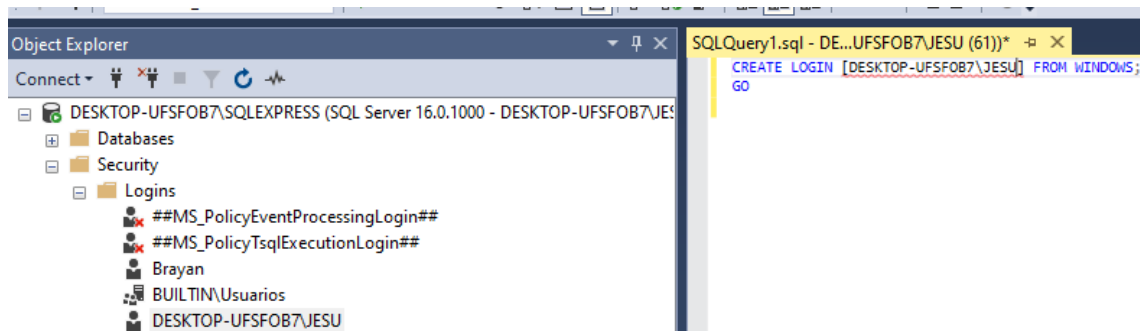
Ejemplo: Un usuario de Windows puede ser validado mediante autenticación integrada o utilizando un inicio de sesión de SQL Server.

## Modos de Seguridad en SQL Server

La seguridad en SQL Server se estructura en varios niveles:

- **Seguridad en Windows:** Utiliza usuarios locales, grupos, y usuarios de dominio para controlar accesos.
- **Seguridad en SQL Server:** Usa inicios de sesión a nivel de servidor y roles para gestionar permisos.
- **Seguridad en la Base de Datos:** Implementa usuarios específicos, roles y esquemas.

Ejemplo: En SQL Server 2022, puedes configurar un inicio de sesión de Windows:



## Comprender los modos de seguridad:

### 1. Autenticación de Windows

Durante la instalación, se crean grupos de seguridad específicos para gestionar permisos, como:

- **SQLServer2005MSSQLUser\$nombreDeEquipo\$nombreDeInstancia:** Para proteger archivos y servicios.

**Ventaja:** Simplifica la gestión de permisos y garantiza una integración sólida con el sistema operativo.

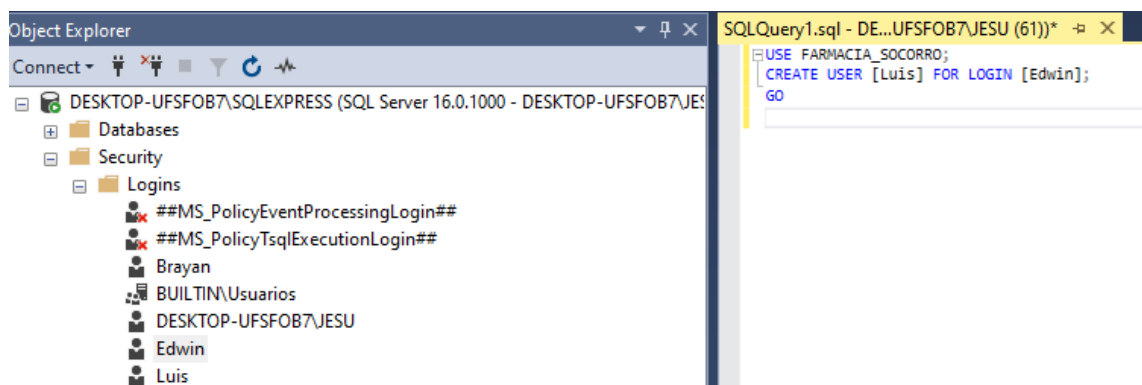
Ejemplo: Un administrador puede permitir que el grupo **SQLServer2005SQLAgentUser\$nombreDeEquipo** ejecute trabajos programados.

SQL Server crea grupos de seguridad durante la instalación, como:

**SQLServer2022MSSQLUser\$Equipo\$Instancia**

Estos grupos simplifican la administración de permisos para servicios y archivos.

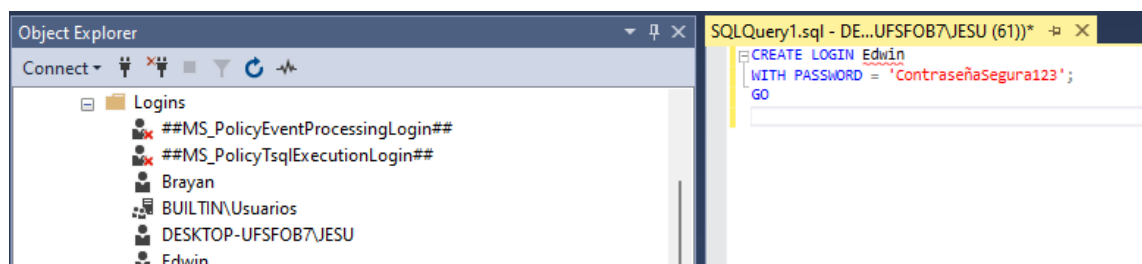
Ejemplo: Para agregar un usuario de dominio a la base de datos:



## 2. Autenticación de SQL Server (mixta):

Permite la validación tanto de usuarios de Windows como de SQL Server. Requiere cambiar la configuración predeterminada.

Ejemplo: Crear un inicio de sesión de SQL Server con autenticación:



La validación mixta (Windows y SQL Server) permite más flexibilidad. Se crean inicios de sesión con T-SQL:

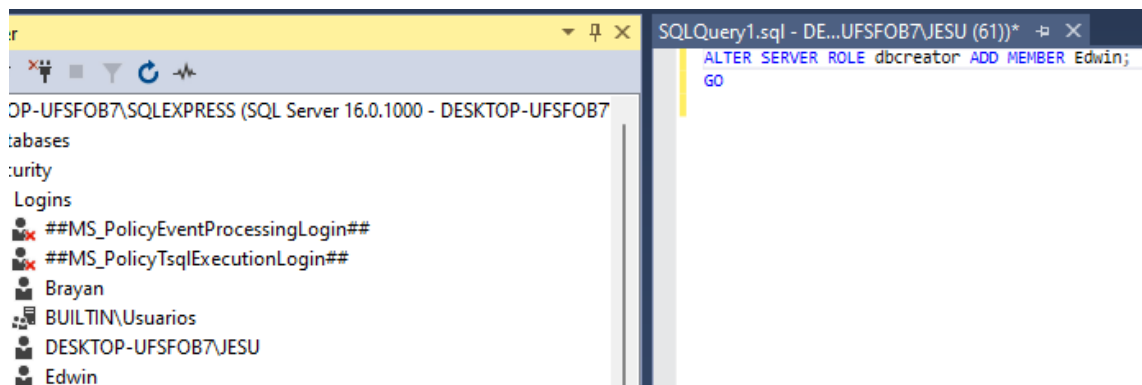
- CREATE LOGIN: Para añadir usuarios.
- ALTER LOGIN: Para modificar usuarios.
- DROP LOGIN: Para eliminar usuarios.

### 3. Roles de Servidor:

SQL Server incluye roles de servidor para administrar permisos globales:

- bulkadmin: Ejecutar operaciones masivas.
- dbcreator: Crear, modificar y eliminar bases de datos.
- securityadmin: Administrar inicios de sesión y permisos.
- sysadmin: Realizar cualquier operación en el servidor.

Ejemplo: Asignar un rol a un usuario:



### 4. Usuarios y Esquemas de Base de Datos:

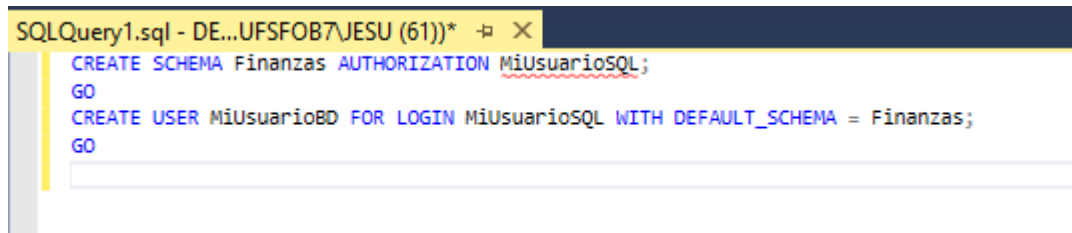
Los usuarios en SQL Server están asociados a un esquema predeterminado. Los esquemas permiten organizar objetos en espacios de nombres independientes.

Ventajas de los Esquemas:

- Varios usuarios pueden compartir un esquema.
- Simplifica la eliminación de usuarios sin afectar los objetos del esquema.

- Facilita el desarrollo de aplicaciones al abstraer la gestión de usuarios.

Ejemplo: Crear un esquema y asignarlo a un usuario:



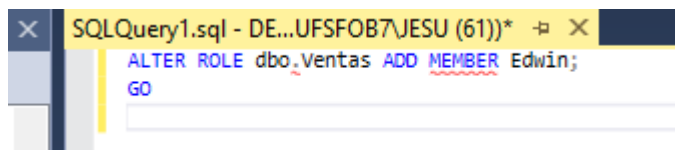
```
SQLQuery1.sql - DE...UFSFOB7\JESU (61))* - + X
CREATE SCHEMA Finanzas AUTHORIZATION MiUsuarioSQL;
GO
CREATE USER MiUsuarioBD FOR LOGIN MiUsuarioSQL WITH DEFAULT_SCHEMA = Finanzas;
GO
```

## 5. Permisos a Nivel de Base de Datos:

SQL Server 2022 ofrece roles predeterminados para gestionar permisos específicos:

- db\_datareader: Leer datos de todas las tablas.
- db\_datawriter : Escribir datos en todas las tablas.
- db\_owner : Control total sobre la base de datos.

Ejemplo: Asignar permisos de lectura a un usuario:



```
SQLQuery1.sql - DE...UFSFOB7\JESU (61))* - + X
ALTER ROLE db_datareader ADD MEMBER Edwin;
GO
```

## 6. Seguridad Avanzada en SQL Server 2022:

SQL Server 2022 introduce características avanzadas como:

- Encriptación Siempre Activa (Always Encrypted): Protege datos sensibles mediante claves almacenadas fuera de la base de datos.
- Acceso Basado en Roles (RBAC): Implementación granular de permisos.
- Auditoría y Seguridad Avanzada: Incluye capacidades como auditorías detalladas y soporte mejorado para políticas de cumplimiento.

Ejemplo: Configurar Always Encrypted para una columna:

```
SQLQuery1.sql - DE...UFSFOB7\JESU (61))* -> X
CREATE COLUMN MASTER KEY MiClaveMaestra
WITH
    KEY_STORE_PROVIDER_NAME = 'AZURE_KEY_VAULT',
    KEY_PATH = 'https://mykeyvault.vault.azure.net/keys/mykey';
GO

CREATE COLUMN ENCRYPTION KEY MiClaveColumna
WITH VALUES
(
    COLUMN_MASTER_KEY = MiClaveMaestra,
    ALGORITHM = 'RSA_OAEP'
);
GO

CREATE TABLE Clientes (
    ID INT PRIMARY KEY,
    Nombre NVARCHAR(100) COLLATE Latin1_General_BIN2 ENCRYPTED WITH
    (
        ENCRYPTION_TYPE = DETERMINISTIC,
        ALGORITHM = 'AEAD_AES_256_CBC_HMAC_SHA_256',
        COLUMN_ENCRYPTION_KEY = MiClaveColumna
    )
);
GO
```

## 7. Auditoría de Seguridad:

La auditoría permite registrar actividades en el servidor y bases de datos. SQL Server 2022 mejora la compatibilidad con herramientas de análisis.

Ejemplo: Crear una auditoría y especificar eventos:

```
SQLQuery1.sql - DE...UFSFOB7\JESU (61))* -> X
CREATE SERVER AUDIT MiAuditoria
TO FILE (FILEPATH = 'C:\Auditorias\')
WITH (QUEUE_DELAY = 1000, ON_FAILURE = CONTINUE);
GO

CREATE SERVER AUDIT SPECIFICATION MiAuditoriaEspecificas
FOR SERVER AUDIT MiAuditoria
ADD (FAILED_LOGIN_GROUP);
GO

ALTER SERVER AUDIT MiAuditoria WITH (STATE = ON);
GO
```