

Unidad 6

Control de Acceso

6.1 Tipos de usuarios.

6.2 Creación de usuarios.

6.3 Privilegios de usuarios.

6.4 Roles.

6.1 Tipos de Usuarios

Algunos de los tipos de usuario más relevantes que interactúan con las bases de datos en SQL Server son:

- Administradores del sistema (sa=*sysadmin*).
 - Tiene acceso ilimitado a todas las funcionalidades del servidor de SQL Server (el DBA).
- Usuarios de Aplicaciones.
 - Usuarios genéricos utilizados por aplicaciones para acceder a la base de datos. No están asociados directamente a individuos.
- Usuarios regulares.
 - Programadores, DBA junior, usuarios sofisticados. Se les asignan permisos individuales para realizar tareas específicas.

6.1 Tipos de Usuarios

No hay un estándar para la creación de la estructura de usuarios en los DBMS.

En SQL Server y otros DBMS, la creación de usuarios se trata de manera distinta con respecto a unos y otros DBMS.

6.2 Creación de usuarios

Para los ejemplos que estudiaremos, se requiere que SQL Server esté configurado para acceso mixto .

Si no se encuentra configurado para modo mixto, acceda a la siguiente liga para obtener las instrucciones respecto a cómo modificarlo: <https://docs.microsoft.com/es-es/sql/database-engine/configure-windows/change-server-authentication-mode?view=sql-server-2017>

6.2 Creación de usuarios

SQL Server hace una separación de *Login* y *Usuario* para controlar quien puede acceder al servidor y qué puede hacer en cada base de datos.

Aunque son dos conceptos independientes, solo puede haber un *Login* para un *Usuario* y viceversa.

Login/Usuario sa (system administrator) y el usuario de Windows tienen todos los derechos sobre el servidor y sobre las bases de datos.

6.2 Creación de usuarios

Creación de usuarios

- Acceder a SQL Server Managment Studio (***sa***)
- Crear un nuevo acceso (LOGIN)
Security → Logins → Botón Derecho → New Login
- Asignar el nombre al Login (*BillGates*)
- Elegir método mixto de autenticación (*SQL Server Authentication*) y retirar la marca en "*Forzar a políticas de password*".

6.2 Creación de usuarios

Crear un nuevo usuario equivalente al *Login*:

- Acceder a *Security* → *Logins* → *BillGates*
- *Botón Derecho* → *Propiedades*
- En el menú de la izquierda elegir *User Mappings*
- Hacer click en la Base de Datos ITD.
El DBMS sugiere un usuario con el mismo nombre que el *Login*.

6.2 Creación de usuarios

Acceder a SQL Server con el Login *BillGates*

Intente acceder a una de las Bases de Datos de la lista excepto *ITD*

Intente acceder a la Base de Datos *ITD*, observe que tablas están visibles

6.3 Privilegios de usuarios

acceder a SQL Server con el *Login BillGates*

```
USE ITD                --(debe permitir el acceso)
SELECT *
FROM Alumnos           --(debe impedir el acceso)

USE ITD2                --(debe impedir el acceso,
                        --lea el mensaje de error)
```

6.3 Privilegios de usuarios

acceder con el *Login sa*

```
USE ITD
```

```
GRANT
```

```
    SELECT ON Alumnos  
    TO BillGates
```

6.3 Privilegios de usuarios

acceder con el *Login BillGates*

```
SELECT *  
FROM Alumnos --(permitirá el acceso)  
  
INSERT INTO  
    Alumnos  
VALUES  
    (15040001, 'CBTIS 130', 4) --(debe impedirlo, lea  
                                --el mensaje de error)
```

6.3 Privilegios de usuarios

acceder con el *Login sa*

USE ITD

GRANT

INSERT, UPDATE ON Alumnos

TO BillGates

acceda con el *Login BillGates* y verifique si puede realizar las operaciones indicadas.

6.3 Privilegios de usuarios

Verificación de permisos

(1 Usuario y 1 objeto en particular)

```
USE ITD
```

```
EXECUTE AS USER='BillGates'
```

```
SELECT * FROM
```

```
fn_my_permissions('Alumnos', 'OBJECT')
```

6.3 Privilegios de usuarios

acceder con el *Login sa*

USE ITD

REVOKE

UPDATE ON Alumnos
FROM BillGates

acceda con el *Login BillGates* y verifique si puede realizar un update o consulte los permisos que posee.

6.3 Privilegios de usuarios

-- Todos los Usuarios y objetos

Use ITD

SELECT

```
    Usuario          = Usuarios.name,  
    Permiso          = Permisos.permission_name,  
    EstadoPermiso    = Permisos.state_desc,  
    TipoObjeto       = obj.type_desc,  
    NombreObjeto     = OBJECT_NAME(Permisos.major_id)
```

FROM

```
    sys.database_principals Usuarios
```

LEFT JOIN

```
    sys.database_permissions Permisos
```

ON Permisos.grantee_principal_id = Usuarios.principal_id --Permisos

LEFT JOIN

```
    sys.objects obj ON Permisos.major_id = obj.object_id
```

WHERE

```
    Usuarios.type='S'AND obj.type_desc IS NOT NULL
```

ORDER BY Usuarios.Name, OBJECT_NAME(Permisos.major_id)

-- Usuarios.type='S' (se refiere a los Usuarios de SQL Server)

6.4 Roles

En SQL Server se cuenta con roles predeterminados del sistema y se pueden crear roles personalizados.

Los roles de sistema son conjuntos predefinidos de permisos que otorgan capacidades específicas a nivel de servidor y a nivel de base de datos.

6.4 Roles

Roles de Nivel de Servidor de SQL Server

Agrupan los permisos a nivel del Servidor de BD y sus Bases de Datos.

1. **sysadmin:** Control total sobre el servidor SQL Server.
2. **serveradmin:** Administrar la configuración del servidor.
3. **securityadmin:** Administrar la seguridad del servidor, incluyendo logins, usuarios y permisos.
4. **processadmin:** Administrar procesos que están corriendo en el servidor.
5. **setupadmin:** Administrar la configuración de servidores vinculados.
6. **bulkadmin:** Ejecutar operaciones de copia masiva.
7. **diskadmin:** Administrar los discos o SSDs.
8. **dbcreator:** Crear y eliminar bases de datos.

6.4 Roles

Roles de Nivel de Base de Datos

Agrupan los permisos sobre una base de datos específica

1. **db_owner:** Control total sobre una base de datos.
2. **db_securityadmin:** Administra la seguridad de la base de datos (permisos y roles).
3. **db_accessadmin:** Administra los accesos a la base de datos.
4. **db_backupoperator:** Permite realizar copias de seguridad de la base de datos.
5. **db_ddladmin:** Permite ejecutar comandos DDL.
6. **db_datareader:** Permite leer todos los datos en la BD.
7. **db_datawriter:** Permite escribir datos en todas las tablas de la BD.
8. **db_denydatareader:** Niega permisos de lectura.
9. **db_denydatawriter:** Niega permisos de escritura.

6.4 Roles

Ejemplos de Uso

Asignar un usuario al **rol de servidor**:

```
ALTER SERVER ROLE dbcreator  
    ADD MEMBER BillGates
```

Asignar un usuario al **rol de base de datos**:

```
ALTER ROLE db_datareader  
    ADD MEMBER PatoDonald
```

6.4 Roles

ROLES PERSONALIZADOS

A nivel de servidor

1. Crear el rol

```
CREATE SERVER ROLE rol_DBAJunior
```

2. Asignar Permisos

```
GRANT VIEW ANY DATABASE  
TO rol_DBAJunior
```

3. Agregar un usuario al rol

```
ALTER SERVER ROLE rol_DBAJunior  
ADD MEMBER BillGates
```

6.4 Roles

ROLES PERSONALIZADOS

A nivel de base de datos

1. Crear el rol

```
CREATE ROLE rol_ProgJunior
```

2. Asignar permisos al rol

```
GRANT SELECT, UPDATE ON Alumnos  
TO rol_ProgJunior
```

3. Agregar un usuario al rol

```
ALTER ROLE rol_ProgJunior  
ADD MEMBER PatoDonald
```

6.4 Roles

Beneficios al crear roles personalizados

- **Flexibilidad:**

En virtud de que los permisos se adaptan de acuerdo a cada entorno. Es decir, se pueden modificar en cualquier momento de acuerdo a cada situación particular.

- **Fácil Administración:**

Se consigue al agrupar usuarios con necesidades similares.

Ejercicio

En SQL Server acceda a la Base de Datos ITD (considere que contiene por lo menos las tablas y vistas siguientes).

- Alumnos
- AlumnosRelaciones
- Maestros
- Personas
- FormasContacto
- InasistAlum
- Materias
- Grupos
- GruposMaestro
- PagosAlumnos

- vAlumnos.
- vMaestros.
- vGrupos.
- vGruposMaestro.
- vPersonasFContacto

Considere que además de **GRANT CREATE VIEW**, se requiere otorgar privilegios de **ALTER ANY SCHEMA** al usuario o al rol correspondiente.

El ejercicio consiste en crear los roles y usuarios que se indican enseguida, asignar roles u otorgar los privilegios necesarios de acuerdo con sus características.

6.4 Privilegios de usuarios

- **ProgJunior (BillGates, SteveJobs, JeffBezzos)**
 - Solo programará interfaces para obtener reportes.
 - No tiene conocimiento del esquema completo de la BD, solo de las vistas (subesquemas) necesarias para facilitarle la obtención de consultas.
- **ProgSenior (AlanTuring, JohnVonNeumann)**
 - Además de lo que hace un Programador Junior, puede hacer interfaces para modificar la BD, es decir puede escribir programas para realizar altas de nueva información, modificar o eliminar datos de la BD.
- **DBAJunior (AdaLovelace, CharlesBabbage)**
 - Puede realizar las tareas de los programadores.
 - Puede crear y modificar vistas a partir de cualquier tabla u otra vista.

6.4 Privilegios de usuarios

Prepare una propuesta de los privilegios que deben tener cada uno de los usuarios indicados, así como los comandos requeridos para otorgar los privilegios.

Observe que el Administrador del Sistema (sa/root) cuenta con todos los privilegios y los tres usuarios que se proponen solo tendrán los que usted considere de acuerdo al problema planteado.