**Control de accesos**

1. Primero que todo tiene que utilizar sus bases de datos locales.
2. Debe iniciar sesión con Windows Authentication.
3. Clic derecho sobre el servidor, ir a propiedades, luego seguridad y cambiar de Windows Authentication mode a SQL Server and Windows Authentication mode.
4. Clic derecho sobre el servidor de nuevo, darle restart, y aceptar y esperar a que finalice, cierre sesión y vuelva a iniciar.
5. Los primeros 4 puntos solo se realizan una vez para configurar el funcionamiento.
6. Ir a Security de manera general, luego ir a crear un login nuevo y llena La información necesaria.
7. Se van a descargar la BD de prueba que vamos a hacer ([AdventureWorks sample databases - SQL Server | Microsoft Docs](https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver15&tabs=ssms)).
8. Realizan un back up.
9. Luego se debe crear un usuario y se le debe asignar el login creado, además en Membership a la izquierda pueden agregarlos específicamente a algún rol ya existente.

Existen 2 niveles diferentes donde podemos implementar un esquema de seguridad:

|  |  |
| --- | --- |
| **Servidor de Base de datos** | **Base de datos individual** |
| Security | Security |
| Logins se les administra una contraseña | Users no se les administra una contraseña |
| Server Roles | Roles |
| Permisos de lectura/escritura/vista | Permisos de lectura/escritura/vista |

[Entidades de seguridad (motor de base de datos) - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/relational-databases/security/authentication-access/principals-database-engine?view=sql-server-ver15)

[Roles de nivel de servidor - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/relational-databases/security/authentication-access/server-level-roles?view=sql-server-ver15)

[Roles de nivel de base de datos - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/relational-databases/security/authentication-access/database-level-roles?view=sql-server-ver15)

Procedimientos almacenados para saber quienes tiene algún rol o permisos

Exec sp\_help para ver el general

Exec sp\_helplogins ‘sa’

Exec sp\_helpuser

Exec sp\_helpsrvrole ‘sysadmin’

Exec sp\_helpsrvrolemember ‘sysadmin’

En la BD que se realizó Backup:

* Databases Roles: es donde se trabaja creando los diferentes roles que se requieran. Aquí estos permisos son específicos para la propia BD.

GRANT (para dar permisos, los permisos pueden ser heredados por esto puede denegarse otros) [GRANT (Transact-SQL) - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/t-sql/statements/grant-transact-sql?view=sql-server-ver15)

REVOKE (para remover permisos) [REVOKE (permisos de extremo de Transact-SQL) - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/t-sql/statements/revoke-endpoint-permissions-transact-sql?view=sql-server-ver15)

DENY (para denegar permisos) [DENY (permisos de servidor de Transact-SQL) - SQL Server | Microsoft Docs](https://docs.microsoft.com/es-es/sql/t-sql/statements/deny-server-permissions-transact-sql?view=sql-server-ver15)

En Database Roles, clic derecho new Database Role,llena la información necesaria y luego va a Securables al lado izquierdo, luego search y objetos específicos, buscar los tipos de objetos y seleccionar tablas, browse y elegir una tabla nada más. Debajo de eso aparecen los permisos.

Los permisos que se le den a algún usuario depende de las necesidades que cada uno requiera, un contador o un auditor, o etc

Si quieren repasar les dejo este link:

[SQL Server Parte 1: Control de Acceso. - YouTube](https://www.youtube.com/watch?v=piIy-bA6Qlk)