UNIVERSIDAD TÉCNICA NACIONAL

SEDE SAN CARLOS

ÁREA DE DOCENCIA

INGENIERÍA EN TECNBOLOGIAS DE LA INFORMACIÓN

ADMINISTRACIO DE BASES DE DATOS

TAREA BD

TEMA

ASIGNACION DE ROLES

PROFESOR

VICTOR IVAN ZUÑIGA ZUÑIGA

ESTUDIANTE

JUSTIN RODRIGUEZ GONZALEZ

AÑO: 2024

Roles nativos

Respaldo: Este rol es el que se encarga de como dice la palabra respaldar la información de la empresa de tener guardado y seguro la información de esta también la posibilidad de poder restaurar información de suma importancia para la empresa donde es de vital importancia llevar un control de versiones y controlar bien cualquier altercado que se pueda ejecutar.

```
BACKUP DATABASE [nombreBD] TO DISK ‘C:\backups\NombreDeLaBaseDeDatos.bak'
WITH FORMAT, MEDIANAME = 'SQLServerBackups', NAME = 'Full Backup of
NombreDeLaBaseDeDatos’.
```

Este fragment esta especificamente utilizado para hacer tareas que se usen en un usuario de respaldo el BACKUP DATABASE esto lo que hace es mostrar que se va a hacer un respaldo de las bases de datos y dentro el nombre el TO DISK indica donde se va a guardar MEDIANAME esto hace un nombre descriptivo para el respaldo

Administración: Este rolo lo que hace es darles permisos a los demás usuarios según la necesidad que necesite cada uno de los usuarios o trabajadores para que puedan interactuar en las bases de datos de una manera clara y segmentada según sus permisos es un rol esencial para llevar un control muy bueno en las bases de datos.

```
USE [NombreDeLaBaseDeDatos];

CREATE USER [NombreDeUsuario] FOR LOGIN [NombreDeUsuarioSQLServer];

ALTER ROLE db_owner ADD MEMBER [NombreDeUsuario];
```

Este lo que hace es otorgarle permisos específicos a un usuario en este caso le da todos los permisos, pero un usuario puede especificar cuales dar y cuales son más necesarios según el rol del trabajador

Seguridad: Es de los roles mas importante ya que ellos pueden ver los permisos de los trabajadores en una empresa también pueden asignar estos, ellos están a cargo por supuesto de la seguridad de una base de datos sabiendo que lo mas importante es el filtrado de la información y que esta no se expuesta ya que para cualquier empresa puede ser de sumo riesgo.

```
OPEN SYMMETRIC KEY MiClaveColumna

DECRYPTION BY CERTIFICATE MiCertificadoConClave

SELECT ClienteID, Nombre,

CONVERT (NVARCHAR, DecryptByKey(NumeroTarjetaCredito)) AS
NumeroTarjetaCredito

FROM Clientes;

CLOSE SYMMETRIC KEY MiClaveColumna
```

Este código lo que hace es crear una clave maestra que lo que hace es proteger otras claves y esto lo que hace es encriptar las claves maestras y para poder consultar los datos se tiene que usar una función que lo que hace es descifrar estos datos.

Cifrado de datos

El cifrado de datos lo que hace es proteger lo datos con los que cuenta la empresa los mas que se pueda con diferentes algoritmos que son implementados lo cuales le an una dificultada más alta para las personas que buscan como acceder a esta información esto ayuda a resguardar de diferentes maneras la información de la empresa y hacerla segura y confiable.

1. AES ((Advanced Encryption Standard)
2. Triple DES (3DES)
3. SHA (Secure Hash Algorithm):

Aplicaciones del cifrado

Uso de las llaves: El uso lo que hacer es que normalmente se tiene un clave publica para poder acceder a cualquier dato lo que se hace en estos casos es cifrar estos datos públicos con una clave privada la cual se necesite descifrar para poder acceder a estos datos o sea que es de suma importancia para agregar mucha más seguridad a nuestra base de datos esto lo que haces es por supuesto generar esta clave.

```
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'StrongPassword!123';
-- Crear una llave simétrica
 CREATE SYMMETRIC KEY MySymmetricKey WITH ALGORITHM = AES_256
ENCRYPTION BY PASSWORD = 'AnotherStrongPassword!123';
```

Uso de Certificados: este funciona de manera diferente ya que cifra los datos con un clave que es publica, pero para poder acceder a ellos se tiene que utilizar una clave privada.

```
USE Practica13_grupo2;
-- Crear una base de datos maestra y un certificado
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'StrongPassword!123';
-- Crear un certificado
CREATE CERTIFICATE MyCertificate
WITH SUBJECT = 'My Encryption Certificate';
```

CRUD sobre Datos Cifrados: Este lo que hace es por ejemplo en unas bases de datos se realizan muchas operaciones CRUD y es muy común así que lo que este hace es que mientras las operaciones se estén haciendo los datos están descifrados pero una vez el proceso no esté en ejecución se vuelve a cifrar de una manera segura.

```
-- Cifrar y crear un nuevo registro
OPEN SYMMETRIC KEY MySymmetricKey
DECRYPTION BY CERTIFICATE MyCertificate;
DECLARE @PlainText NVARCHAR(100) = 'SensitiveData';
DECLARE @EncryptedData VARBINARY(MAX);
SET @EncryptedData = ENCRYPTBYKEY(KEY_GUID('MySymmetricKey'),
@PlainText);
INSERT INTO SensitiveTable (EncryptedColumn)
VALUES (@EncryptedData);
CLOSE SYMMETRIC KEY MySymmetricKey;
```

Eliminar el Cifrado de Datos Existentes: esto lo que hace es que al eliminar el cifrado este se quita t se vuelven los datos a su estado original.

```
-- Eliminar un dato cifrado
DELETE FROM SensitiveTable
WHERE SomeCondition = 'Condition';
GO
```

Material extra que investigue:

Propiedad	Descripción
`RoleAssignmentName` `name`	Nombre de la asignación de roles, que es un identificador único global (GUID).
`RoleAssignmentId` `id`	Identificador único de la asignación de roles, que incluye el nombre.
`Scope` `scope`	Identificador de recursos de Azure al que se limita la asignación de roles.
`RoleDefinitionId` `roleDefinitionId`	Identificador único del rol.
`RoleDefinitionName` `roleDefinitionName`	Nombre del rol.
`ObjectId` `principalId`	Identificador de objeto de Microsoft Entra de la entidad de seguridad que tiene asignado el rol.
`ObjectType` `principalType`	Tipo de objeto de Microsoft Entra que representa la entidad de seguridad. Los valores válidos son `User`, `Group` y `ServicePrincipal`.
`DisplayName`	Para las asignaciones de roles de los usuarios, el nombre para mostrar del usuario.
`SignInName` `principalName`	El nombre principal de usuario (UPN) o el nombre de la aplicación asociada a la entidad de servicio.
`Description` `description`	Descripción de la asignación de roles.
`Condition` `condition`	Instrucción de condición compilada mediante una o varias acciones de definición de roles y atributos.
`ConditionVersion` `conditionVersion`	Número de versión de una condición. El valor predeterminado es 2.0 y es la única versión admitida.
`CanDelegate` `canDelegate`	Sin implementar.

Referencias:

https://www.pandasecurity.com/es/mediacenter/cifrado-aes-guia/#:~:text=El%20cifrado%20AES%2C%20o%20advanced,una%20encriptaci%C3%B3n%20f%C3%A1cil%20de%20usar.

https://ciberseguridad.com/guias/prevencion-proteccion/criptografia/cifrado-3des/

https://www.ibm.com/docs/es/db2/11.5?topic=backup-examples

https://docs.plesk.com/es-ES/obsidian/administrator-guide/servidores-de-bases-de-datos/permisos-y-roles-de-los-usuarios-de-bases-de-datos.74697/

https://learn.microsoft.com/es-es/azure/analysis-services/analysis-services-database-users

https://learn.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles?view=sql-server-ver15

https://administraciondesistemas.com/secure-hash-algorithm-sha/

https://www.keepersecurity.com/blog/es/2023/05/09/what-is-a-hardware-security-key-and-how-does-it-work/

Implementación de RBAC: Gestión de Accesos por Roles (tecnetone.com)