**NOMBRE: HELEN FARINA RENTERIA RENTERIA**

Actividad noviembre 19 de 2024

Estructura del Taller

1. Conceptos básicos

Antes de comenzar, repasa estos conceptos:

* Usuario: Identidad que se conecta a la base de datos.
* Permisos: Conjunto de reglas que determinan qué puede hacer un usuario.
* Rol: Grupo de permisos que se puede asignar a varios usuarios.
* Host: Define desde dónde puede conectarse un usuario (local o remoto).

2. Ejercicios prácticos (teóricos)

Sigue los pasos con atención para imaginar o planificar cómo configurarías cada escenario.

**Parte 1: Crear usuarios**

Actividad 1: Crear usuarios con diferentes restricciones

Crea un usuario local (analista) que solo pueda conectarse desde el servidor.

|  |
| --- |
| **Create user “analista”@”localhosht” identified by “contraseña”;** |

**Pregunta: ¿Por qué usarías localhost en lugar de %?**

Cuando utilizamos localhosht las conexiones solo están en la el computador donde está el servidor y esto ayuda a mejorar la seguridad.

Crea un usuario remoto (remoto) que pueda conectarse desde cualquier lugar.

|  |
| --- |
| **Create user “analista”@”%” identified by “contraseña”;** |

**Pregunta: ¿Qué riesgos puede tener esta configuración?**

Cuando utilizamos e **%** nos permite las conexiones desde cualquier lugar lo que aumenta el riesgo de ataques remotos los cuales pueden causar muchos daños desde diferentes perspectivas.

**Parte 2: Otorgar permisos**

Actividad 2: Permitir acceso limitado

Permitir al usuario analista consultar datos de una tabla **específica (empleados).**

|  |
| --- |
| **Gran select on datos\_empeados to “analista”@”localhost”;** |

Permitir al usuario remoto insertar y consultar datos.

|  |
| --- |
| **Granr select, insert on datos\_empleados to “remoto”@”%”;** |

Restricción adicional: Revoca el permiso de inserción al usuario remoto.

|  |
| --- |
| **Revoke onsert on datos\_empleados from “remoto” @ “%”;** |

**Pregunta: ¿Qué sucede si remoto intenta insertar datos después de revocar el permiso?**

Le aparecerá error porque MySQL confirma los permisos en cada interacción por lo que no será necesario que cierre la sesión para que se apliquen los cambios.

**COMANDOS DE PERMISOS**

|  |  |
| --- | --- |
| **COMANDO** | **DESCRIPCIÓN** |
| GRANT SELECT ON ... | Da permiso para consultar datos. |
| GRANT INSERT ON ... | Da permiso para insertar datos. |
| REVOKE INSERT ON ... | Revoca el permiso de insertar datos. |
| SHOW GRANTS FOR 'usuario'@'host'; | Muestra los permisos actuales asignados a un usuario. |

**Parte 3: Uso de roles**

Actividad 3: Crear y asignar roles

Crea un rol llamado lectura\_escritura con permisos para leer e insertar en cualquier tabla

|  |
| --- |
| **Create ROLE lectura\_escritura;**  **Gran select, insert on \*.\* to “lectura\_escritura”;** |

Asigna el rol lectura\_escritura al usuario remoto.

|  |
| --- |
| **Grant lectura\_escritura to “remoto”@”%”;** |

**Pregunta: Si eliminamos el rol, ¿los permisos del usuario remoto se mantienen?**

NO porque los permisos están guardados en lectura\_escritura y el usuario remoto hereda estos permisos si lectura\_escritura no existe entonces no habría ningún permiso

**Parte 4: Políticas avanzadas (conceptual)**

Configura seguridad por filas: Imagina que solo deseas que analista pueda ver empleados

con salarios menores a 6000.

Política conceptual:

|  |
| --- |
| CREATE POLICY analista\_policy  ON empleados  FOR SELECT TO analista  USING (salario < 6000); |

**Pregunta: ¿Qué registros esperas que pueda ver analista después de aplicar esta política?**

Cuando apliquemos esta política, analista solo podrá ver los empleados con un salario menor a 6000 y los salarios mayores estarán ocultos para analista.

NOTA: tenemos que asegurarnos de habilitar la seguridad por filas en la tabla

|  |
| --- |
| Alter table empleados ENABLE row level segurity; |