Tarea 3.2: Asignación de permisos en MySQL

RA3: Implanta métodos de control de acceso utilizando asistentes, herramientas gráficas y comandos del lenguaje del sistema gestor.

Criterios de Evaluación:

- c) Se han definido y eliminado cuentas de usuario.
- d) Se han identificado los privilegios sobre las bases de datos y sus elementos.
- f) Se han asignado y eliminado privilegios a usuarios.
- h) Se ha garantizado el cumplimiento de los requisitos de seguridad.

Entrega un documento llamado **Tarea3.2**.**pdf**

Ejercicios:

1. Crea un usuario 'tengopermisos' y otórgale permisos para que pueda crear usuarios.

2. Conectado como 'tengopermisos' crea un nuevo usuario 'user1'.

3. Conectado como 'root', otórgale permisos al usuario 'user1' para que pueda crear tablespaces.

Conectado como 'user1' comprueba que dispone de dichos permisos ejecutando sentencias SQL que necesiten tener el permiso otorgado.

4. Conectado como 'root', muestra los permisos que tiene el usuario 'user1'.

Conectado como 'user1' muestra los permisos que posee y comprueba que son los mismos a los de la orden anterior.

5. Conectado como 'root' crea un usuario 'matador' que tenga permisos para poder eliminar conexiones activas y pueda ver las conexiones activas de todos los usuarios con el servidor.

6. Conectado como 'root' otorga permiso de creación y borrado de procedimientos, así como de ejecución al usuario 'user1' sobre una base de datos llamada ‘pruebas’ creada previamente.

Conectado como 'user1' crea un procedimiento almacenado en la base de datos indicada en el paso anterior, con las siguientes órdenes:

```
DELIMITER $$
 DROP PROCEDURE IF EXISTS test_mysql`$$
 CREATE PROCEDURE test_mysql()
 BEGIN
      DECLARE x  INT;
      SET x = 1;
```

```
      WHILE x  >= 0 DO
        set x = x+1;
      END WHILE;
 END$$
DELIMITER ;
```

Este procedimiento crea un bucle infinito. Conéctate como 'user1' y ejecuta el procedimiento con la orden SQL: call test_mysql_while_loop

Nota: Si te da algún tipo de error, prueba a cerrar la conexión y volver a conectarte.

7. Conectado como 'matador' identifica el proceso e intenta matarlo. ¿Puedes?. ¿Por qué?. Mátalo con el usuario que puede hacerlo.

8. Conectado como 'root' crea un usuario de nombre 'creartablas' que tenga permisos para crear, borrar y modificar tablas de la base de datos creada previamente llamada ‘pruebas’.

9. Conectado como 'creartablas' crea un tabla sencilla de al menos dos columnas.

10. Conectado como 'root' crea un usuario de nombre 'accesoglobal' que pueda realizar operaciones de selección y inserción sobre todas las tablas de todas las bases de datos.

11. Conectado como 'accesoglobal' añade una fila a la tabla creada anteriormente. Intenta borrar la fila creada. ¿ Puedes ?

12. Conectado como 'root' crea un usuario de nombre 'accesolocal' que pueda seleccionar todas las tablas de la base de datos anterior.

Conéctate como 'accesolocal' y comprueba que puedes selecciona la fila añadida anteriormente.

13. Conectado como 'root' crea un usuario de nombre 'accesolimitado' que pueda realizar operaciones de inserción, actualización y selección sobre la primera columna de la tabla creada previamente.

Conéctate como 'accesolimitado' y comprueba que tienes los permisos ejecutando las órdenes SQL SELECT, UPDATE e INSERT.

Comprueba qué permisos tienes.

14. Quita los permisos 'específicos' otorgados a cada uno de los usuarios anteriores, comprobando con la orden SQL SHOW GRANTS que realmente fueron eliminados.