

Usuarios y permisos en base de datos

Creación de usuarios

Cuando se empieza a usar MySQL se recibe un usuario y contraseña y sus credenciales iniciales serán las del root, sin embargo, no es lo adecuado para todos los usuarios.

Para crear un usuario se accede a la cmd e ingresar al servidor MySQL, posteriormente el script devolverá un resultado que confirma que estas accediendo a un servidor MySQL.

Luego se ejecuta el comando *CREATE USER 'nuevo_usuario'@'localhost' IDENTIFIED BY 'contraseña';*

En el cual *nuevo_usuario* es el nombre de la nueva cuenta de usuario e *IDENTIFIED BY 'contraseña'* establece la contraseña para el usuario.

Ahora bien, hasta este punto el usuario no cuenta con permisos en la base de datos, por lo cual se deben agregar sus credenciales.

Eliminación de usuarios

Puedes eliminar varios usuarios con un solo comando *DROP USER*. Debe ser un super usuario o tener los permisos necesarios para ejecutar el comando.

DROP USER [IF EXISTS] name [, ...]

La cláusula *IF EXISTS* significa que si el usuario no existe, entonces el comando no debe realizar cambios, y *name* hace referencia al nombre de usuario que se eliminara, puede ser uno solo o varios, siempre y cuando se separen con una coma.

Tipos de permisos

Los permisos más comunes en MySQL son los siguientes:

- **ALL PRIVILEGES:** Todos los privilegios.
- **CREATE:** Permite crear nuevas tablas o bases de datos.
- **DROP:** Permite eliminar tablas o bases de datos.
- **DELETE:** Permite eliminar filas de las tablas.
- **INSERT:** Permite insertar filas en las tablas.
- **SELECT:** Les permite consultar información de una base de datos o tabla.
- **UPDATE:** Permite actualizar las filas de las tablas.
- **GRANT OPTION:** Permite otorgar o eliminar privilegios a otros usuarios.

Asignación de permisos

Para otorgar todos los privilegios de la base de datos para un usuario recién creado, se ejecuta el siguiente comando:

```
GRANT ALL PRIVILEGES ON *.* TO 'nuevo_usuario'@'localhost';
```

Los asteriscos en este comando se refieren a la base de datos y la tabla (respectivamente) a los que pueden acceder, por lo cual este comando permite al usuario leer, editar, ejecutar y realizar todas las tareas en todas las bases de datos y tablas.

Una vez que haya finalizado los permisos que desea configurar para sus nuevos usuarios, es necesario volver a cargar los privilegios, con el siguiente comando:

```
FLUSH PRIVILEGES;
```

Ahora, si queremos proporcionar un permiso a un usuario específico, se puede ocupar el siguiente comando:

```
GRANT      type_of_permission      ON      database_name.table_name      TO  
'username'@'localhost';
```

Revocación de permisos

La estructura es muy similar a la de otorgar permisos:

```
REVOKE      type_of_permission      ON      database_name.table_name      FROM  
'username'@'localhost';
```

Se debe tener en cuenta que cuando se revocan los permisos, la sintaxis requiere que utilice *FROM* en lugar de *TO*.

Mostrar privilegios de un usuario

Para encontrar que privilegios se han otorgado a cierto usuario se ocupa el siguiente comando:

```
SHOW GRANTS FOR 'nombre_usuario'@'localhost';
```

Roles

Los roles se utilizan para gestionar con mayor facilidad los privilegios de los grupos de usuarios, pues simplifican el proceso de gestión de privilegios, ya que se pueden otorgar privilegios a un rol y luego otorgar el rol a usuarios. Y cuando desee revocar privilegios para un usuario, simplemente se revoca la autorización del usuario, en lugar de revocar cada privilegio de manera individual.

Los roles se crean y descartan utilizando el mismo proceso para realizar cualquier cambio de objeto de base de datos.

Ejemplos

En el siguiente ejemplo se crea el usuario pedro, sin ninguna credencial.

```
CREATE USER 'pedro'@'localhost' IDENTIFIED BY 'password';
```

Ahora se le asignan los permisos para una base de datos específica y se actualizan los privilegios.

```
GRANT ALL PRIVILEGES ON `basededatos` . * TO 'pedro'@'localhost';
```

```
FLUSH PRIVILEGES;
```

Por último se eliminará el usuario pedro si existe, si no es así, no se realiza ninguna acción y se devuelve un mensaje.

```
DROP USER IF EXISTS pedro;
```

Bibliografía

- Deyimar, A. (2020, julio 29). Cómo crear un usuario MySQL y otorgar privilegios: una guía para principiantes. Tutoriales Hostinger. <https://www.hostinger.mx/tutoriales/como-crear-usuario-mysql>
- IBM Data Studio 4.1.1. (2021, marzo 17). Ibm.com. <https://www.ibm.com/docs/es/data-studio/4.1.1?topic=management-database-roles>
- Mauricio. (2022, agosto 31). ¿Cómo crear usuario mysql? guía básica. Tutoriales Dongee. <https://www.dongee.com/tutoriales/crear-usuario-mysql/>
- Sverdlov, E. (2014, diciembre 3). Cómo crear un nuevo usuario y otorgar permisos en MySQL. Digitalocean.com; DigitalOcean. <https://www.digitalocean.com/community/tutorials/crear-un-nuevo-usuario-y-otorgarle-permisos-en-mysql-es>
- (S/f-a). Amazon.com. Recuperado el 30 de abril de 2024, de https://docs.aws.amazon.com/es_es/redshift/latest/dg/r_DROP_USER.html
- (S/f-b). Clouding.io. Recuperado el 30 de abril de 2024, de <https://help.clouding.io/hc/es/articles/4408406585618-C%C3%B3mo-administrar-usuarios-y-permisos-en-MySQL>
- (S/f-c). Codigofacilito.com. Recuperado el 30 de abril de 2024, de <https://codigofacilito.com/articulos/asignar-permisos-mysql>