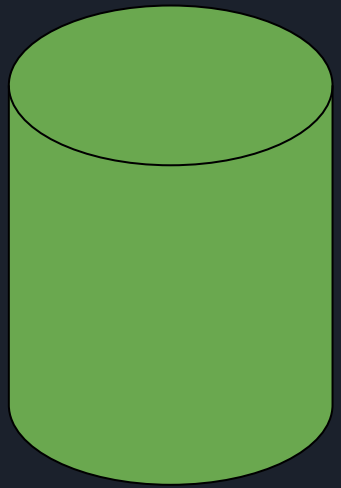
MODELOS DE DATOS

USUARIOS Y PRIVILEGIOS

RECORDERIS...

¿QUÉ APRENDEREMOS HOY?



- Definición de usuario
- Definición de rol
- Administración de usuarios
  - Creación
  - Borrado
  - Cambio de contraseñas
- Tipos de privilegios
- Alcance de cada privilegio
- Limitar recursos a usuarios y roles

DEFINICIÓN: USUARIO

Es una cuenta en la base de datos, la cual puede ser usada por una persona o por una aplicación.

Se compone por un nombre de usuario (login) más una ip o rango de ip en notación CIDR.

Documentación:
https://mariadb.com/kb/en/create-user/

DEFINICIÓN: ROL

Es una forma de agrupar usuarios de una misma área organizacional o con un mismo nivel de permisos.

Facilita la administración en bases de datos con muchos usuarios.

Documentación:
https://mariadb.com/kb/en/roles_overview/

EJEMPLO GRÁFICO

USUARIOS: ¿Para qué?

- Seguridad: Cada usuario tendrá su forma de acceder (generalmente con usuario y contraseña).
- Trazabilidad: Es posible saber quién hace qué, en caso que se requiera una actividad "post-mortem"
- Control de acceso: Es posible definir quién hace qué, y qué límites de ejecución existen. Acá se aplica el famoso "principio del mínimo privilegio"

USUARIOS: ¿Para qué?

- Control: Es posible configurar expiracion de contraseñas, bien sea para forzar el cambio o para que el usuario no pueda volver a conectarse (ej: empleados temporales).

- En posible bloquear o desbloquear cuentas, en algún evento que así lo amerite

ROLES: ¿Para qué?

- Al tener muchos usuarios la administración se vuelve muy compleja. Los roles facilitan esta labor, ya que se administra un grupo en vez de los usuarios de forma individual.

- Permite administrar a los usuarios que cambian de área organizacional o de rol. Ejemplo: Eliminar un desarrollador del rol de desarrollo y luego asignarlo al rol de DBA

IMPORTANTE!!!

Repite conmigo:

"Debería evitar trabajar con el usuario "root", salvo que sea estrictamente necesario!!!"

"NUNCA debo dejar el usuario root sin contraseña o con contraseñas fáciles de obtener, en bases de datos que estén operando en proyectos reales"

¿LO SABÍAS?

Una de las mejores formas de perder el control sobre la base de datos es darle a las personas o a las aplicaciones el usuario root.

Adicionalmente, otra de las mejores formas de perder el control sobre la base de datos es darle a las personas o a las aplicaciones el usuario root.

Por último, por favor, recuerda que una de las mejores formas de perder el control sobre la base de datos es darle a las personas o a las aplicaciones el usuario root.

¿Está claro el mensaje?

¿LO SABÍAS?

NUNCA JAMÁS EN LA VIDA en una base de datos operando, le definas al usuario root contraseñas como "1234", "12345678", "password", "mariadb", "root", "admin", "qwertyuiop", "0000" o similares que sean fáciles de recordar.

Si no me crees, consulta como han sido las formas más comunes de hackeo sobre bases de datos.

¿Está claro el mensaje?

USUARIOS: Estructura

Un usuario en MySQL o MariaDB es:

login + @ dirección ip de conexión (en formato CIDR)

NOTA: Para el alcance de este tema en este curso no se profundizará en el tema de las direcciones IP ni tampoco en las limitantes de ejecución.

Por el momento, tengamos claro que "alguien" y "alguien@%" significan lo mismo, y es para permitir que el usuario "alguien" pueda conectarse desde cualquier lado hacia la base de datos.

IMPORTANTE!!!

Repite conmigo:

"Debo evitar crear usuarios con la ip %, salvo que sea estrictamente necesario!!!"

USUARIOS: Creación

NOTA: Estos 3 ejemplos son sólo ilustrativos, su uso es totalmente desaconsejado

Con conexión desde cualquier lugar (explícito), sin especificar contraseña	`create user 'selena'@'%';`
Con conexión desde cualquier lugar (implícito), sin especificar contraseña	`create user 'selena';`
Con conexión desde cualquier lugar (explícito), especificando contraseña en claro	`create user 'selena'@'%' identified by 'selena123';`

USUARIOS: Creación y borrado

Con expiración automática de contraseña en 30 días	`create user 'selena' password expire interval 30 day;`

Con conexión desde cualquier lugar especificando contraseña en hash (la cual debe generarse primero)	`select password('selena123');`
	
	`create user selena identified by password '*225BBFE6ABC549D00279B8424B453A46F900F9F5'`

Cambiar nombre	`rename user 'selena' to 'selena_gomez';`

Eliminar usuario con acceso desde cualquier parte	`drop user selena;`
	
	`drop user selena@'%';`

USUARIOS: Modificación

Modificar contraseña en claro (totalmente desaconsejado)	`alter user selena identified by 'selena456';`

Modificar contraseña usando hash	`select password('selena456');`
	
	`alter user selena identified by password '*CA05827C8ECB76C61DE69C8457356E652F57276B'`

Bloquear cuenta	`alter user selena account lock;`
Desbloquear cuenta	`alter user selena account unlock;`

ROLES

Crear rol	`create role singers;`
Eliminar rol	`drop role singers;`
Asociar usuario a rol	`grant singers to selena;` `grant singers to selena@'%';`
Desasociar usuario de rol	`revoke singers from selena;` `revoke singers from selena@'%';`

CONSULTAR

NOTA: La consulta de usuarios retorna muchos campos. Se recomienda usar un ide, o cambiar el * por la lista de campos que se deseen mostrar.

Consultar usuarios	`select * from mysql.user;`
Consultar roles	`select * from mysql.roles_mapping;`

USUARIOS: Notas adicionales

- Los usuarios y roles se crean a nivel global en la instancia, y no a una base de datos específica (como por ejemplo pasa con las tablas)
- Un usuario puede pertenecer a varios roles
- Es posible (aunque no recomendable) asignar un rol a otro rol
- No hay control para cantidad de intentos fallidos

DEFINICIÓN: Privilegio

En pocas palabras, determina hasta dónde puede llegar un usuario. Qué puede hacer o que no puede hacer.

La idea es que los usuarios solo puedan realizar tareas DDL (alteración de estructura) o DML (alteración de datos) que les corresponda hacer.

PRIVILEGIOS: ¿Para qué?

- Permite definir límites de lo que cada usuario o role puede hacer
- Protegen el acceso no permitido a secciones de la base de datos
- Ayudan a evitar una posible pérdida de información por error
- Son una de las herramientas que posee el DBA para complementar la administración de los usuarios

PRIVILEGIOS: Clases

Globales	`*.*`	`Instancia completa`
A nivel de base de datos	`world.*`	`Base de datos completa`
A nivel de rutinas	`world.fn_get_city` `world.pr_get_cities`	`Rutinas específicas`
A nivel de tablas	`world.country` `world.city`	`Tablas específicas`
A nivel de columnas en tablas	`world.country.Name` `world.city.Name`	`Columnas específicas de tablas específicas`

¿Que privilegios tiene mi instancia?

```
MariaDB [gd]> show privileges;
+-------------------------+---------------------------------------+-------------------------------------------------------------+
| Privilege               | Context                               | Comment                                                     |
+-------------------------+---------------------------------------+-------------------------------------------------------------+
| Alter                   | Tables                                | To alter the table                                          |
| Alter routine           | Functions,Procedures                  | To alter or drop stored functions/procedures                |
| Create                  | Databases,Tables,Indexes              | To create new databases and tables                          |
| Create routine          | Databases                             | To use CREATE FUNCTION/PROCEDURE                            |
| Create temporary tables | Databases                             | To use CREATE TEMPORARY TABLE                               |
| Create view             | Tables                                | To create new views                                         |
| Create user             | Server Admin                          | To create new users                                         |
| Delete                  | Tables                                | To delete existing rows                                     |
| Delete history          | Tables                                | To delete versioning table historical rows                  |
| Drop                    | Databases,Tables                      | To drop databases, tables, and views                        |
| Event                   | Server Admin                          | To create, alter, drop and execute events                   |
| Execute                 | Functions,Procedures                  | To execute stored routines                                  |
| File                    | File access on server                 | To read and write files on the server                       |
| Grant option            | Databases,Tables,Functions,Procedures | To give to other users those privileges you possess         |
| Index                   | Tables                                | To create or drop indexes                                   |
| Insert                  | Tables                                | To insert data into tables                                  |
| Lock tables             | Databases                             | To use LOCK TABLES (together with SELECT privilege)         |
| Process                 | Server Admin                          | To view the plain text of currently executing queries       |
| Proxy                   | Server Admin                          | To make proxy user possible                                 |
| References              | Databases,Tables                      | To have references on tables                                |
| Reload                  | Server Admin                          | To reload or refresh tables, logs and privileges            |
```

PRIVILEGIOS: Asignación

Permitir conexión a la base de datos y uso (i.e.: comando "use") en toda la instancia	`grant usage on *.* to selena;`
Permitir conexión a la base de datos y uso (i.e.: comando "use") en una base de datos específica	`grant usage on mydb.* to selena;`

NOTA: Usuarios sin el privilegio "usage" no podrán conectarse a una base de datos

PRIVILEGIOS: Asignación

Permitir creación de rutinas (funciones y procedimientos) a un usuario en la base de datos world	`grant create routine on world.* to selena;`
Permitir modificación y borrado de rutinas	`grant alter routine on world.* to selena;`
Permitir creación de tablas, bases de datos e índices	`grant create on world.* to selena;`
Permitir borrado de tablas, bases de datos e índices	`grant drop on world.* to selena;`

PRIVILEGIOS: Asignación

Permitir consultar una tabla	`grant select on world.city to selena;`
Permitir insertar registros en una tabla	`grant insert on world.city to selena;`
Permitir las 4 operaciones usando una única sentencia	`grant insert, update, delete, select on world.city to selena;`
Permitir seleccionar ciertas columnas en una tabla	`grant select(Name, Code) on world.country;`

PRIVILEGIOS: Asignación

Permitir todo lo que sea posible en una base de datos	`grant all privileges on world.* to selena;`
Permitir todo lo que sea posible en una instancia	`grant all privileges on *.* to selena;`

NOTA: Estas 2 opciones se explican a modo ilustrativo y por conocimiento de su existencia, sin embargo son totalmente desaconsejadas por motivos de seguridad de los datos

¿LO SABÍAS?

Al igual que prestar la contraseña del usuario root, otra de las mejores formas de perder el control sobre la base de datos es darle a un usuario "all privileges", en especial con "with grant option".

Adicionalmente, otra de las mejores formas de perder el control sobre la base de datos es darle a un usuario "all privileges", en especial con "with grant option"..

Por último, por favor, recuerda que una de las mejores formas de perder el control sobre la base de datos es darle a un usuario "all privileges", en especial con "with grant option".

¿Está claro el mensaje?

IMPORTANTE!!!

Repite conmigo:

"NUNCA debería ejecutar el comando grant all privileges a un usuario regular, salvo que sea estrictamente necesario"

PRIVILEGIOS: Eliminación

Impedir conexión a la base de datos y uso en toda la instancia	`revoke usage on *.* from selena;`
Impedir modificación y borrado de rutinas	`revoke alter routine on world.* from selena;`
Impedir creación de tablas, bases de datos e índices	`revoke create on world.* from selena;`
Impedir las 4 operaciones DML usando una única sentencia	`revoke insert, update, delete, select on world.city from selena;`
Impedir todo lo que sea posible	`revoke all privileges on *.* from selena;`

IMPORTANTE!!!

Al realizar cualquier tipo de cambio de privilegios es posible que éstos no queden aplicados de forma inmediata. Para solucionar esta situación, asegurarse de ejecutar el comando "flush privileges" después de aplicar grant o revoke.

PRIVILEGIOS: Roles

Asignar 2 usuarios a un rol. Darle permisos al rol. Los usuarios heredarán los permisos del rol (Esta es la verdadera utilidad de los roles)	`create role pop_singers;`
	`grant pop_singers to selena, ariana;`
	
	`grant select, insert, update on world.city to pop_singers;`

Asignar permisos a un rol con opción de propagación	`grant select, update on world.city to pop_singers with grant option;`
Asignar permisos de modificar DDL a un rol (y por ende, a los usuarios que hacen parte de éste)	`grant create on world.* to pop_singers;`

PRIVILEGIOS: Consulta

Consultar privilegios de un usuario o rol	`show grants for selena;` `show grants for pop_singers;`

```
MariaDB [gd]> show grants for selena;
+---------------------------------------------------------------------+
| Grants for selena@%                                                 |
+---------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `selena`@`%`                                  |
| GRANT SELECT ON `gd`.`singers` TO `selena`@`%` WITH GRANT OPTION    |
+---------------------------------------------------------------------+
2 rows in set (0,009 sec)
```

```
MariaDB [gd]> show grants for pop_singers;
+---------------------------------------------------------------------+
| Grants for pop_singers                                              |
+---------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `pop_singers`                                 |
| GRANT CREATE ON `world`.* TO `pop_singers`                          |
| GRANT SELECT ON `gd`.`singers` TO `pop_singers` WITH GRANT OPTION   |
+---------------------------------------------------------------------+
3 rows in set (0,000 sec)
```

IMPORTANTE!!!

RECORDERIS:

La instalación de MariaDB en los PC de las salas de la universidad se hicieron con el usuario root sin contraseña.

Esto está bien para propósitos académicos, pero nunca para usar en un sistema real o productivo.

ENUNCIADO

Se cuenta con el siguiente equipo de trabajo para un proyecto de desarrollo de software:

Nombre completo	Cargo en el proyecto
Albert Einstein	Director del proyecto
Leonardo Da Vinci	Desarrollador
Isaac Newton	Desarrollador
Galileo Galilei	Analista de pruebas
Marie Curie	Analista de pruebas
Charles Darwin	Analista de pruebas
Nikola Tesla	Desarrollador
Graham Bell	Analista de negocio
Nicolas Copernico	Analista de negocio
Michael Faraday	Desarrollador

ENUNCIADO

- El área de TI de la empresa ha determinado que el login del usuario corresponde a la primera letra del nombre y luego los apellidos hasta completar 8 caracteres. Ejemplo: Policarpa Salabarrieta sería "psalabar", Manuel Elkin Patarroyo sería "mpatarro", Selena Gomez sería "sgomez" y así sucesivamente.

- Los roles de los integrantes son "developer" para los desarrolladores, "qa" para pruebas y "business" para analistas de negocio.

- Al director del proyecto no le interesa acceder al sistema.

EJERCICIOS

Escribir las sentencias para satisfacer lo enunciado en las diapositivas anteriores.

Recomendaciones:

- Analizar primero la estructura del login
- Posteriormente analizar qué usuarios deben ir en qué roles

Una vez tenga analizado lo anterior, le será mucho más fácil escribir el código.

EJERCICIOS

Usando los mismos usuarios y roles que acaba de crear, definir los siguientes permisos:

- Rol de desarrolladores:
  - insert, select sobre las tablas "city" y "country"
  - crear, modificar y eliminar rutinas
- Rol de analistas de pruebas:
  - insert, select, update, delete sobre las tablas "city", "country" y "countrylanguage"
- Rol de analistas de negocio:
  - select sobre la instancia completa

EJERCICIOS

Configurar una nueva conexión en el ide de su preferencia (en clases se está usando el mysql workbench), pero no con el usuario root sino con alguno de los usuarios recién creados. Validar que las conexiones se acepten o se denieguen.

Una vez validadas las conexiones del ejercicio anterior, intentar las operaciones de select, update, etc…, conectándose como un usuario desarrollador y como un analista de pruebas.

Validar que las operaciones sean admitidas o denegadas acorde a los permisos del rol.

¿PREGUNTAS?