

Cómo instalar MySQL en Ubuntu 18.04

Introducción

MySQL es un sistema de gestión de bases de datos de código abierto, que generalmente está instalado como parte de la popular combinación LAMP (Linux, Apache, MySQL, PHP/Python/Perl). Gestiona sus datos usando una base de datos relacional y SQL (Lenguaje de consulta estructurada).

La versión breve de la instalación es simple: Actualice el índice de su paquete, instale el paquete mysql-server y luego ejecute el script de seguridad incluido.

```
sudo apt update
```

```
sudo apt install mysql-server
```

```
sudo mysql_secure_installation
```

Este tutorial le explicará cómo instalar la versión 5.7 de MySQL en un servidor Ubuntu 18.04.

Paso 1 — Instalar MySQL

Únicamente la última versión de MySQL se incluye en el repositorio de paquete APT de forma predeterminada en Ubuntu 18.04. Al momento de escribir esto, esa sería la versión MySQL 5.7.

Para instalarla, actualice el índice del paquete en su servidor con apt:

```
sudo apt update
```

Luego, instale el paquete predeterminado:

```
sudo apt install mysql-server
```

Esto instalará MySQL, pero no le pedirá que cree una contraseña ni que haga ningún otro cambio de configuración. Dado a que esto deja su instalación de MySQL insegura, vamos a abordarlo a continuación.

Paso 2 — Configurar MySQL

Para las instalaciones recientes, querrá ejecutar el script de seguridad que viene incluido. Esto cambia algunas de las opciones predeterminadas menos seguras para cosas como inicios de sesión root remotos y usuarios de ejemplo. Para las versiones antiguas de

MySQL, también deberá inicializar el directorio de datos manualmente, pero ahora esto se hace automáticamente.

Ejecute el script de seguridad:

```
sudo mysql_secure_installation
```

Esto hará que pase por una serie de indicaciones en las que puede hacer algunos cambios en las opciones de seguridad de su instalación de MySQL. La primera indicación le preguntará si quiere configurar el plugin de Validación de Contraseña, la que puede usarse para probar la solidez de su contraseña de MySQL. Independientemente de lo que seleccione, la siguiente indicación será establecer una contraseña para el usuario **root** de MySQL. Ingrese y luego confirme una contraseña segura de su elección.

Desde este punto, puede presionar Y y luego ENTER para aceptar las configuraciones predeterminadas para todas las preguntas siguientes. Esto eliminará algunos usuarios anónimos y la base de datos de prueba, deshabilitará los inicios de sesión root remotos y cargará estas nuevas reglas para que MySQL respete los cambios que haya realizado inmediatamente.

Para iniciar el directorio de datos de MySQL, debe usar `mysql_install_db` para versiones anteriores a 5.7.6, y `mysqld --initialize` para la versión 5.7.6 y posteriores. No obstante, si instaló MySQL desde la distribución Debian, como se describió en el Paso 1, el directorio de datos se inició automáticamente; no es necesario que haga nada. Si trata de ejecutar el comando igual, se le mostrará el siguiente error:

Output

```
mysqld: Can't create directory '/var/lib/mysql/' (Errcode: 17 - File exists)
...
2018-04-23T13:48:00.572066Z 0 [ERROR] Aborting
```

Note que, si bien estableció una contraseña para el usuario **root** de MySQL, este usuario no está configurado para autenticarse con una contraseña al conectarse al shell de MySQL. Si quiere, puede ajustar esta configuración siguiendo el Paso 3.

Paso 3 — (Opcional) Ajustar la autenticación y los privilegios del usuario

Para los sistemas Ubuntu que estén usando MySQL 5.7 (y las versiones posteriores), el usuario **root** de MySQL está configurado, de forma predeterminada, para autenticarse usando el plugin `auth_socket` en vez de una contraseña. En muchos casos, esto permite que la seguridad y usabilidad sea mayor pero también puede complicar las cosas cuando deba permitir que un programa externo (tal como phpMyAdmin) tenga acceso al usuario.

Deberá cambiar su método de autenticación de `auth_socket` a `mysql_native_password` para usar una contraseña para conectarse a MySQL como **root**. Para hacerlo, abra la indicación de MySQL desde su terminal:

```
sudo mysql
```

Posteriormente, consulte cuál método de autenticación usa cada una de sus cuentas de usuario de MySQL usando el siguiente comando:

```
SELECT user,authentication_string,plugin,host FROM mysql.user;
```

En este ejemplo, puede ver que el usuario **root** verdaderamente se autentica usando el plugin `auth_socket`. Para configurar la cuenta **root** para autenticarse usando una contraseña, ejecute el siguiente comando `ALTER USER`. Asegúrese de cambiar `password` (contraseña) a una contraseña segura de su elección y sepa que este comando cambiará la contraseña de **root** que estableció en el Paso 2:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password';
```

Luego, ejecute `FLUSH PRIVILEGES` (purgar privilegios), lo que le dice al servidor que vuelva a cargar las tablas `grant` e implemente sus nuevos cambios:

```
FLUSH PRIVILEGES;
```

Vuelva a verificar los métodos de autenticación que usa cada uno de sus usuarios para confirmar que **root** ya no se autentica usando el plugin `auth_socket`:

```
SELECT user,authentication_string,plugin,host FROM mysql.user;
```

En este resultado de ejemplo, puede ver que, ahora, el usuario **root** de MySQL se autentica usando una contraseña. Una vez que confirme esto en su propio servidor, puede salir del shell de MySQL:

```
exit
```

Alternativamente, para otras personas puede adaptarse mejor a su flujo de trabajo si se conectan a MySQL con un usuario dedicado. Para crear tal usuario, vuelva a abrir el shell de MySQL nuevamente:

```
sudo mysql
```

Nota: Si tiene la autenticación de contraseña habilitada para `root` según se describió en los párrafos de arriba, deberá usar un comando diferente para acceder al shell de MySQL. Lo que se indica a continuación ejecutará su cliente MySQL con privilegios de usuario regular, y solamente tendrá privilegios de administrador dentro de la base de datos una vez que haga la autenticación:

```
mysql -u root -p
```

Desde ese punto, cree un nuevo usuario y use una contraseña sólida:

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

Luego, dele a su nuevo usuario los privilegios adecuados. Por ejemplo, puede concederle al usuario privilegios a todas las tablas dentro de la base de datos, así como autoridad para agregar, cambiar y eliminar privilegios de usuario, mediante este comando:

```
GRANT ALL PRIVILEGES ON *.* TO 'sammy'@'localhost' WITH GRANT OPTION;
```

Note que, en este punto, no necesita volver a ejecutar el comando FLUSH PRIVILEGES. Solamente necesita este comando al modificar las tablas grant utilizando declaraciones como INSERT, UPDATE o DELETE. Dado a que creó un usuario nuevo en vez de modificar uno existente, no es necesario que use FLUSH PRIVILEGES aquí.

Después de esto, salga del Shell de MySQL:

```
exit
```

Por último, vamos a probar la instalación de MySQL.

Paso 4 — Probar MySQL

Independientemente de cómo lo instaló, MySQL debería haber empezado a ejecutarse automáticamente. Para probar esto, verifique su estado.

```
systemctl status mysql.service
```

Verá un resultado parecido al de abajo:

Output

- mysql.service - MySQL Community Server
Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: en
Active: **active (running)** since Wed 2018-04-23 21:21:25 UTC; 30min ago
Main PID: 3754 (mysqld)
Tasks: 28
Memory: 142.3M
CPU: 1.994s
CGroup: /system.slice/mysql.service
└─3754 /usr/sbin/mysqld

Si MySQL no se está ejecutando, puede iniciarlo usando `sudo systemctl start mysql`.

Para una verificación adicional, puede tratar de conectarse a la base de datos usando la herramienta `mysqladmin`, que es un cliente que le permite ejecutar comandos

administrativos. Por ejemplo, este comando dice que se conecte a MySQL como **root** (-u root), pida una contraseña (-p) y devuelva la versión.

```
sudo mysqladmin -p -u root version
```

Debería ver un resultado parecido a este:

Output

```
mysqladmin Ver 8.42 Distrib 5.7.21, for Linux on x86_64  
Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.
```

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

```
Server version      5.7.21-1ubuntu1  
Protocol version    10  
Connection          Localhost via UNIX socket  
UNIX socket         /var/run/mysqld/mysqld.sock  
Uptime:             30 min 54 sec
```

```
Threads: 1 Questions: 12 Slow queries: 0 Opens: 115 Flush tables: 1 Open tables: 34  
Queries per second avg: 0.006
```

Esto quiere decir que MySQL está funcionando.

¿Cómo crear un nuevo usuario y otorgarle permisos en MySQL?

MySQL es un software de código abierto para la gestión de bases de datos que ayuda a los usuarios a almacenar y organizar datos para poder obtenerlos posteriormente. Tiene una gran variedad de opciones para otorgar permisos particulares a usuarios específicos dentro de las tablas y bases de datos. Este tutorial aborda brevemente algunas de estas opciones.

Crear un nuevo usuario

En la mayoría de los casos en los cuales se crea un usuario MySQL, se requieren restricciones con respecto a los permisos que tendrán estos usuarios dentro de la base de datos o tablas. Hay varias maneras de crear usuarios con permisos personalizados.

Primero ingrese en su Shell MySQL; para ingresar como usuario *root* ejecute el siguiente comando:

```
mysql -u root -p
```

Se le solicitará el ingreso de su contraseña de usuario *root* MySQL (recuerde no confundir con la contraseña del usuario root del servidor).

Ahora continuemos con la creación de un usuario desde el *Shell* MySQL:

```
CREATE USER 'nuevousuario'@'localhost' IDENTIFIED BY 'contraseña';
```

En este punto, *nuevousuario* no tiene permisos para hacer nada con la base datos. De hecho, si *nuevousuario* intenta iniciar sesión (con la contraseña contraseña) no podrá ni siquiera llegar hasta el *shell* MySQL.

Por lo tanto, lo primero que debe hacer es proporcionar al usuario el acceso a la información que necesita.

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

Los asteriscos en este comando se refieren a la base de datos y a la tabla (respectivamente) a las que el usuario puede acceder; este comando específico le permitiría a *nuevousuario* leer, editar, ejecutar y realizar todas las tareas dentro de todas las tablas y bases de datos. ¡Esto es debido a que el asterisco *** representa todo!, en este caso: todas las bases de datos y todas las tablas.

Una vez que haya terminado de asignar los permisos que quiere establecer para sus nuevos usuarios, siempre asegúrese de volver a cargar todos los privilegios.

```
FLUSH PRIVILEGES;
```

Ahora sus cambios surtirán efecto.

Cómo otorgar permisos diferentes a los usuarios

A continuación una lista breve de otras posibilidades de permisos comunes que un usuario podría tener:

- **ALL PRIVILEGES:** tal como vimos antes, esto permitiría que un usuario MySQL accediera completamente a la base de datos designada (o si no se selecciona una base de datos en particular, el usuario tendría acceso a todo el sistema);
- **CREATE:** les permite a los usuarios crear nuevas tablas o bases de datos;
- **DROP:** les permite eliminar tablas o bases de datos;
- **DELETE:** les permite eliminar filas de tablas;
- **INSERT:** les permite insertar filas en tablas;
- **SELECT:** les permite usar el comando **SELECT** para consultar información en bases de datos;

- **UPDATE:** les permite actualizar filas en tablas;
- **GRANT OPTION:** les permite otorgar o suprimir privilegios de otros usuarios de la manera que vimos anteriormente, esto le daría a un usuario MySQL todo el acceso a una base de datos designada (o si no se selecciona una base de datos en particular, a todo el sistema).

Para proporcionar un permiso específico a un usuario, puede usar la siguiente estructura:

```
GRANT [tipo de permiso] ON [nombre de la base de datos].[nombre de la tabla] TO  
'[nombredeusuario]'@'localhost';
```

Si usted desea otorgar acceso a cualquier base de datos o a cualquier tabla, asegúrese de poner un asterisco (*) en lugar del nombre de la base de datos o el nombre de la tabla.

Cada vez que actualice o cambie un permiso, asegúrese de utilizar el comando FLUSH PRIVILEGES mencionado anteriormente.

Si usted necesita revocar un permiso, la estructura es casi idéntica a la primera:

```
REVOKE [tipo de permiso] ON [nombre de la base de datos].[nombre de la tabla]  
FROM '[nombredeusuario]'@'localhost';
```

Tal como se pueden eliminar bases de datos usando DROP, puede usar DROP para borrar un usuario completamente:

```
DROP USER 'usuariodeejemplo'@'localhost';
```

Para probar un usuario nuevo, cierre sesión usando:

```
quit
```

Y vuelva a ingresar con este comando en su terminal:

```
mysql -u [nombredeusuario]-p
```

Las combinaciones para el otorgamiento o revocación de permisos son innumerables y dependerán de la configuración de sus bases de datos y de los usuarios que tendrán acceso a estas. Puede hacer distintas pruebas antes de que su plataforma entre en producción, una vez ahí debe tener más cuidado con los comandos que ejecute y los permisos que otorgue en MySQL.

MySQL Workbench un entorno gráfico para trabajar con MySQL

Para realizar la instalación de esta herramienta en nuestro sistema, **podremos hacerlo directamente desde los repositorios de Ubuntu.**

Por lo que podemos apoyarnos con nuestro centro de software, synaptic o desde la misma terminal **podremos realizar la instalación ejecutando el siguiente comando:**

```
sudo apt install mysql-workbench
```

Para quienes prefieran realizar la instalación utilizando el ultimo paquete deb estable de esta, **pueden dirigirse a la pagina web oficial del proyecto y en su sección de descargas encontraran los enlaces.**

En estos momentos la actual versión estable es la 8.0.15 por lo que si prefieren pueden hacer uso de alguno de los siguientes comandos para realizar la descarga.

Usuarios de Ubuntu 18.10 y derivados, deben ejecutar el siguiente comando en una terminal:

```
wget  
https://dev.mysql.com/get/Downloads/MySQLGUITools/mysql-workbench-community_8.0.15-1ubuntu18.10_amd64.deb
```

Mientras que para los que son **usuarios de Ubuntu 18.04 LTS o derivados de esta versión de Ubuntu**, el paquete que deben descargar es el siguiente:

```
wget  
https://dev.mysql.com/get/Downloads/MySQLGUITools/mysql-workbench-community_8.0.15-1ubuntu18.04_amd64.deb
```

Hecha la descarga, pueden realizar la instalación del paquete con ayuda de su gestor de paquetes preferido o desde la misma terminal **ejecutando el siguiente comando:**

```
sudo dpkg -i mysql-workbench*.deb
```

Y solucionamos cualquier problema con las dependencias ejecutando en una terminal lo siguiente:

```
sudo apt -f install
```

Para comenzar a hacer uso de esta herramienta **debemos ejecutarla, por lo que debes buscar su lanzador en tu menú de aplicaciones o desde la terminal teclear el comando:**

```
mysql-workbench
```


Desinstalar todo

```
sudo apt-get remove --purge *mysql*
```

(Les va a preguntar si quieren eliminar todas las bases creadas selecciones Yes)

```
sudo apt-get remove --purge libcanberra-gtk-module
```

```
sudo apt-get autoremove
```

```
sudo apt-get autoclean
```