**Assignment**

## **ASSIGNMENT on** - Mariadb

## **Prepared by:** Naman Verma

**Dated:** 28 Nov’2022

## **Assigned by:** Varad Sir

**Google link :** <https://www.one.com/en/hosting/what-is-mariadb>

<https://mariadb.com/kb/en/authentication-plugin-pam/>

<https://mariadb.com/kb/en/authentication-with-pluggable-authentication-modules-pam/>

# **What is MariaDB?**

* MariaDB will efficiently enable you to meet all your workload
* Mariadb is a **database**. MariaDB is very similar to **MySQL** (a database management system). The MariaDB database is used for various purposes such as **data warehousing, e-commerce, enterprise-level features, and logging applications.**
* MariaDB will efficiently enable you to meet all your workload; it works in any cloud database and works at any scale – small or large.

# **What is a database?**

# A database is a **place to store information** that you can quickly retrieve and use where you need it.

# **Mariadb database server**

DATABASES — TABLES — RECORDS

# **Why is MariaDB important?**

MariaDB is **open source** which means that it can be used for free. MariaDB hosting comes at a **generally low cost**; it’s also well-documented, meaning that if you need to solve a problem, there is plenty of help online.

# **Service profile :**

**Package -** mariadb-server

**Daemon -** mariadb

**Port no. -** 3306

**Config. File -** /etc/my.cnf

**Log File** - /var/log/mariadb/mariadb.log

Installation and configuration of

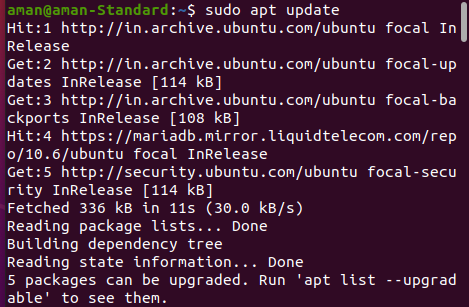
# Mariadb server using PAM in UBUNTU 20.04

## 

## **Step 1: Install MariaDB.**

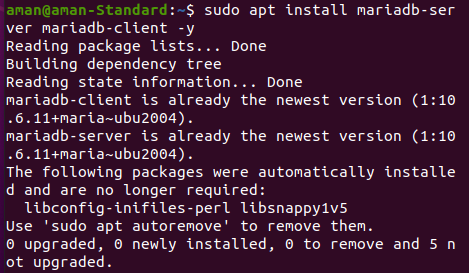
To get started, log into your Ubuntu 20.04 server and refresh the local repositories.

**Syntax :** sudo apt update



Ubuntu repositories already provide MariaDB packages. The installation is straightforward using the PAT package manager as follows.

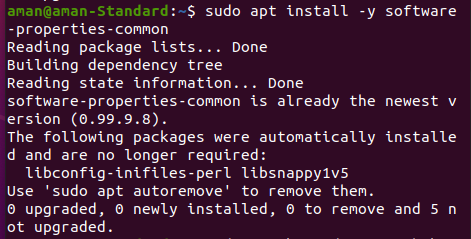
**Syntax :** sudo apt install mariadb-server mariadb-client -y



However, the version of MariaDB provided by Ubuntu repositories is not the latest one. This installs MariaDB 10.3.24 at the time of penning down this guide. To install the latest version, use the official MariaDB repository.

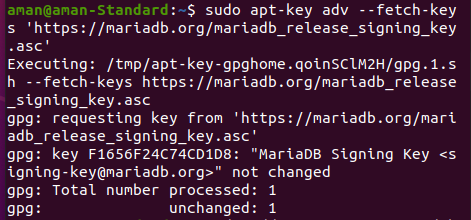
First, install the prerequisite package.

**Syntax :** sudo apt install -y software-properties-common



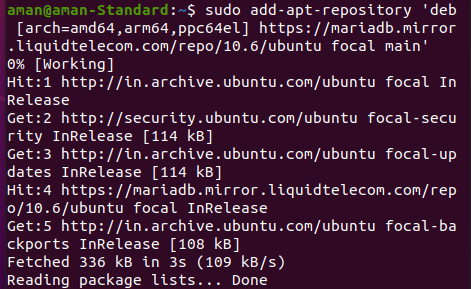
Next, import the GPG signing key.

**Syntax :** sudo apt-key adv --fetch-keys '<https://mariadb.org/mariadb_release_signing_key.asc>'



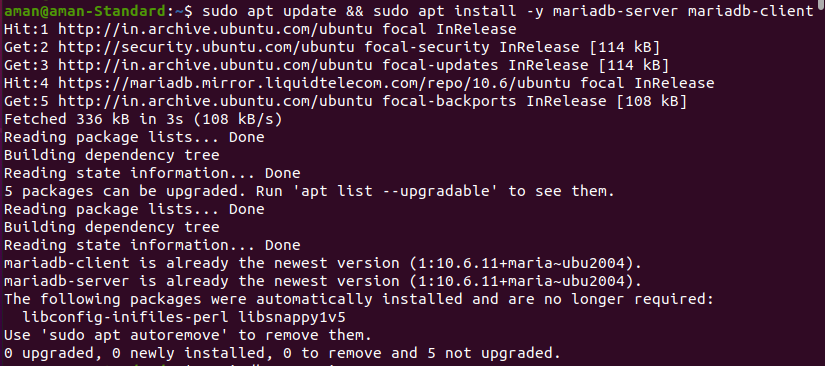
Once the GPG key is in place, add the MariaDB APT repository. The following repository is enabled for the MariaDB version 10.6. You may replace this value with the version that you intend to install.

**Syntax :** sudo add-apt-repository 'deb [arch=amd64,arm64,ppc64el] https://mariadb.mirror.liquidtelecom.com/repo/10.6/ubuntu focal main'



Finally, refresh the local repositories and install the MariaDB server and client by using the APT package manager.

**Syntax :** sudo apt update && sudo apt install -y mariadb-server mariadb-client



The command installs the specified version of the MariaDB database engine and client, alongside other additional packages and dependencies.

You can confirm the installed version of MariaDB as shown.

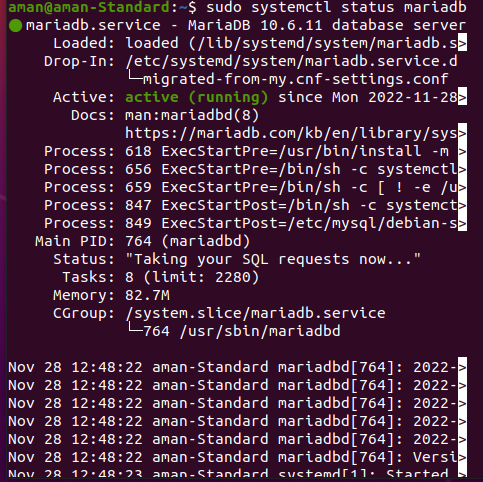
**Syntax :** mariadb --version



## **Step 2: Start And Enable MariaDB.**

By default, the MariaDB database engine starts automatically upon installation. You can verify this by running the command:

**Syntax :** sudo systemctl status mariadb



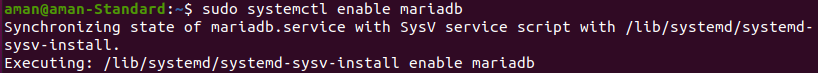
From the above output, you can see that the MariaDB service is running. If for whatever reason, you find that the service is inactive or not running, you can start it as follows.

**Syntax :** sudo systemctl start mariadb



Additionally, consider enabling MariaDB to start every time on system startup as shown.

**Syntax :** sudo systemctl enable mariadb

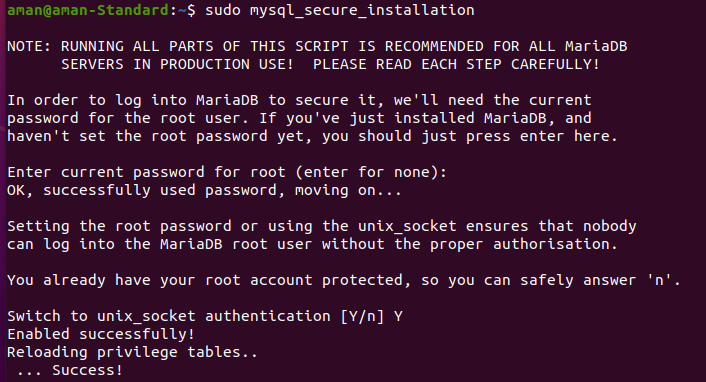


The above command ensures that MariaDB starts automatically every time the server is rebooted or powered on.

## **Step 3: Configure MariaDB.**

To remediate this issue, you need to go an extra step and harden your MariaDB instance. To improve the security of the MariaDB database engine, you need to run the mysql\_secure\_installation shell script as shown.

**Syntax :** sudo mysql\_secure\_installation



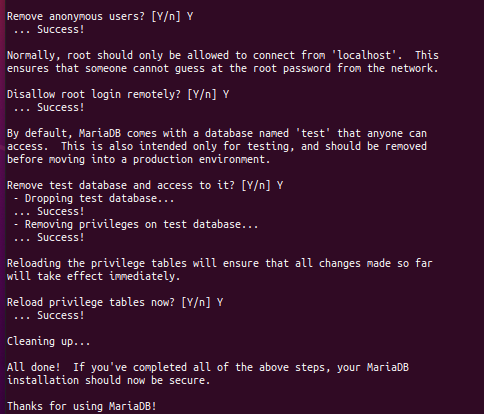
The script walks you through a series of prompts that will require you to make changes to the security options that involve the MariaDB database engine.

The first prompt asks you to provide the current root password, and since none has been set up yet, simply hit ENTER on your keyboard.

Next, you will be prompted for the database root password. This ensures that no one can log in as the root user without authentication. So, type ‘Y’ and provide the database root password and confirm it.

Then press ‘Y’ for the subsequent prompts in order to configure the database engine according to the best security practices. This does the following:

* Removes anonymous users from the database server
* Disables remote root login. This ensures that the root user can only log in to the database server from “localhost”
* Remove the test database which comes with MariaDB by default.
* Reloads privilege tables for the changes t take effect immediately.

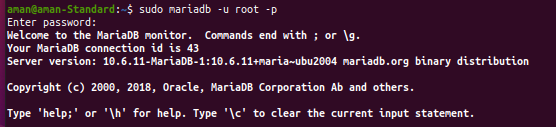


At this point, you have successfully completed the initial security configuration for MariaDB.

## **Step 4: Configure A Password-authenticated Administrative User.**

In this section, you are going to create a new user account in the database server with password authentication and later assign administrative privileges to the user. So, login as the root user as follows:

**Syntax :** sudo mariadb -u root -p



Next, create a regular user. Here, we are creating a user called admin\_user. Be sure to replace secret\_password with your preferred user’s password.

**Syntax :** CREATE USER 'admin\_user'@'localhost' IDENTIFIED BY 'secret\_password';



Next, grant all privileges to admin\_user. This effectively assigns all the database root user’s permissions to the user.

**Syntax :** GRANT ALL PRIVILEGES ON \*.\* TO 'admin\_user'@'localhost';



The \*.\* wildcard notation implies that the user has permission to execute any task on any database in the database server.

To apply the changes, flush the privileges.

**Syntax ;** FLUSH PRIVILEGES;



Finally, exit the database server.

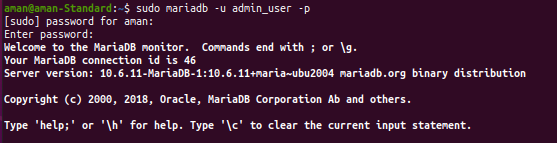
**Syntax :** EXIT;



## **Step 5: Test MariaDB.**

With the MariaDB database server configured, you can now log in and run a few queries. Proceed and log in to the database server using the administrative user that you have just created.

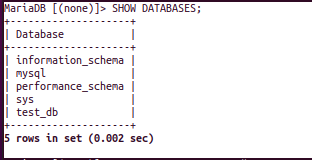
**Syntax :** sudo mariadb -u admin\_user -p



Provide the password for the administrative user and hit ENTER. Just above the MariaDB shell some details about the database server including the connection id, server version, and copyright information will be displayed.

To check the existing databases, run the command:

**Syntax :** SHOW DATABASES;



To create a new database, run the following query. In this example, we are creating a database called test\_db.

**Syntax :** CREATE DATABASE test\_db;

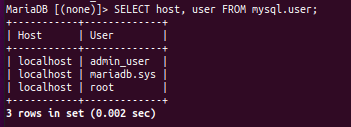


To apply all the changes made without restarting or reloading MariaDB run the query:

**Syntax :** FLUSH PRIVILEGES

To list all the existing users in your database engine, run the following query.

**Syntax :** SELECT host, user FROM mysql.user;



To exit the MariaDB prompt, simply run:

**Syntax :** QUIT



## **Step -6 Configuring PAM Authentication and User Mapping with MariaDB**

## **Set up the user mapper PAM plugin**

**MariaDB’s git repository has a simple** [**user mapper PAM plugin**](https://github.com/MariaDB/server/blob/10.1/plugin/auth_pam/mapper/pam_user_map.c)**. Downloading, compiling, and installing it is simple:**

**Syntax used given below:**

* wgethttps://raw.githubusercontent.com/MariaDB/server/10.1/plugin/auth\_pam/mapper/pam\_user\_map.c
* gcc pam\_user\_map.c -shared -lpam -fPIC -o pam\_user\_map.so
* sudo install --mode=0755 pam\_user\_map.so /lib64/security/

## **Set up the PAM policy**

We want to configure the PAM policy so that:

* Users authenticate with their Linux user names and passwords (i.e. use the *pam\_unix.so* PAM module);
* Login attempts go into the system’s audit logs;
* “Real” user names will be mapped to MariaDB user names (i.e. use the *pam\_user\_map.so* PAM module).

We can create a PAM policy to do all of the above with:

**Syntax :** sudo tee /etc/pam.d/mysql <<EOF

auth required pam\_unix.so audit

account required pam\_unix.so audit

auth required pam\_user\_map.so

EOF

## **Create some test accounts**

Let’s create some Linux accounts to test things out:

# generic "dba" account to map other users to

* sudo useradd dba

# a "real" account for Alice

* sudo useradd alice
* sudo passwd alice

# a "real" account for Bob

* sudo useradd bob
* sudo passwd bob

## **Configuring the user account mapping**

By default, the *pam\_user\_map.so* module looks at */etc/security/user\_map.conf* for the mappings. Let’s map both “alice” and “bob” to the “dba” user:

**Syntax :** sudo tee /etc/security/user\_map.conf <<EOF

alice: dba

bob: dba

EOF

## **Open up access to /etc/shadow**

The *pam\_unix.so* PAM module usually uses the *unix\_chkpwd* utility to handle the authentication. This utility requires read access to */etc/shadow*, which is usually unreadable for security reasons. To get PAM authentication to work with MariaDB, you will probably have to allow the *mysql* user to read this file. This is very easy to do:

**Syntax used given below:**

* sudo groupadd shadow
* sudo usermod -a -G shadow mysql
* sudo chown root:shadow /etc/shadow
* sudo chmod g+r /etc/shadow

## **Set up everything in MariaDB**

Finally, let’s set up everything in MariaDB:

-- Install the plugin

* INSTALL SONAME 'auth\_pam';

-- Create the "dba" user

* CREATE USER 'dba'@'%' IDENTIFIED BY 'strongpassword';
* GRANT ALL PRIVILEGES ON \*.\* TO 'dba'@'%';

-- Create an anonymous catch-all user that will use the PAM plugin and the mysql policy

* CREATE USER ''@'%' IDENTIFIED VIA pam USING 'mysql';

-- Allow the anonymous user to proxy as the dba user

* GRANT PROXY ON 'dba'@'%' TO ''@'%';

Since we changed the *mysql* user’s group membership, we also have to restart the MariaDB service:

**Syntax :** sudo service mysql restart

## **OUTPUT :**

To check the user to get login in mariadb:

**Syntax :** cat /etc/passwd

aman@aman-Standard:~$ **cat /etc/passwd**

alice:x:1017:1021:,,,:/home/alice:/bin/bash

bob:x:1018:1022:,,,:/home/bob:/bin/bash

To login in Mariadb as root:

**Syntax :** mysql -uroot -p

aman@aman-Standard:~$ **mysql -uroot -p**

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 60

Server version: 10.6.11-MariaDB-1:10.6.11+maria~ubu2004 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> **SELECT USER(), CURRENT\_USER();**

+----------------+----------------+

| USER() | CURRENT\_USER() |

+----------------+----------------+

| root@localhost | root@localhost |

+----------------+----------------+

1 row in set (0.001 sec)

MariaDB [(none)]> Ctrl-C -- exit!

Aborted

To login in Mariadb as **alice** user

**Syntax :** mysql -u alice -p

aman@aman-Standard:~$ **mysql -u alice -p**

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 61

Server version: 10.6.11-MariaDB-1:10.6.11+maria~ubu2004 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> **SELECT USER(), CURRENT\_USER();**

+-----------------+----------------+

| USER() | CURRENT\_USER() |

+-----------------+----------------+

| alice@localhost | dba@% |

+-----------------+----------------+

1 row in set (0.000 sec)

MariaDB [(none)]> Ctrl-C -- exit!

Aborted

To login in Mariadb as **bob** user

**Syntax :** mysql -u bob -p

aman@aman-Standard:~$ **mysql -u bob -p**

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 62

Server version: 10.6.11-MariaDB-1:10.6.11+maria~ubu2004 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>  **SELECT USER(), CURRENT\_USER();**

+---------------+----------------+

| USER() | CURRENT\_USER() |

+---------------+----------------+

| bob@localhost | dba@% |

+---------------+----------------+

1 row in set (0.000 sec)

MariaDB [(none)]> Ctrl-C -- exit!

Aborted

aman@aman-Standard:~$ **mysql -uroot -p**

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 63

Server version: 10.6.11-MariaDB-1:10.6.11+maria~ubu2004 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> **SELECT User FROM mysql.user;**

+-------------+

| User |

+-------------+

| |

| dba |

| admin\_user |

| mariadb.sys |

| root |

| foo2 |

+-------------+

6 rows in set (0.012 sec)

MariaDB [(none)]> **SELECT host, user FROM mysql.user;**

+-----------+-------------+

| Host | User |

+-----------+-------------+

| % | |

| % | dba |

| localhost | admin\_user |

| localhost | mariadb.sys |

| localhost | root |

| test | foo2 |

+-----------+-------------+

6 rows in set (0.002 sec)

MariaDB [(none)]> **DESC mysql.user;**

+------------------------+---------------------+------+-----+----------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------------------+---------------------+------+-----+----------+-------+

| Host | char(255) | NO | | | |

| User | char(128) | NO | | | |

| Password | longtext | YES | | NULL | |

| Select\_priv | varchar(1) | YES | | NULL | |

| Insert\_priv | varchar(1) | YES | | NULL | |

| Update\_priv | varchar(1) | YES | | NULL | |

| Delete\_priv | varchar(1) | YES | | NULL | |

| Create\_priv | varchar(1) | YES | | NULL | |

| Drop\_priv | varchar(1) | YES | | NULL | |

| Reload\_priv | varchar(1) | YES | | NULL | |

| Shutdown\_priv | varchar(1) | YES | | NULL | |

| Process\_priv | varchar(1) | YES | | NULL | |

| File\_priv | varchar(1) | YES | | NULL | |

| Grant\_priv | varchar(1) | YES | | NULL | |

| References\_priv | varchar(1) | YES | | NULL | |

| Index\_priv | varchar(1) | YES | | NULL | |

| Alter\_priv | varchar(1) | YES | | NULL | |

| Show\_db\_priv | varchar(1) | YES | | NULL | |

| Super\_priv | varchar(1) | YES | | NULL | |

| Create\_tmp\_table\_priv | varchar(1) | YES | | NULL | |

| Lock\_tables\_priv | varchar(1) | YES | | NULL | |

| Execute\_priv | varchar(1) | YES | | NULL | |

| Repl\_slave\_priv | varchar(1) | YES | | NULL | |

| Repl\_client\_priv | varchar(1) | YES | | NULL | |

| Create\_view\_priv | varchar(1) | YES | | NULL | |

| Show\_view\_priv | varchar(1) | YES | | NULL | |

| Create\_routine\_priv | varchar(1) | YES | | NULL | |

| Alter\_routine\_priv | varchar(1) | YES | | NULL | |

| Create\_user\_priv | varchar(1) | YES | | NULL | |

| Event\_priv | varchar(1) | YES | | NULL | |

| Trigger\_priv | varchar(1) | YES | | NULL | |

| Create\_tablespace\_priv | varchar(1) | YES | | NULL | |

| Delete\_history\_priv | varchar(1) | YES | | NULL | |

| ssl\_type | varchar(9) | YES | | NULL | |

| ssl\_cipher | longtext | NO | | | |

| x509\_issuer | longtext | NO | | | |

| x509\_subject | longtext | NO | | | |

| max\_questions | bigint(20) unsigned | NO | | 0 | |

| max\_updates | bigint(20) unsigned | NO | | 0 | |

| max\_connections | bigint(20) unsigned | NO | | 0 | |

| max\_user\_connections | bigint(21) | NO | | 0 | |

| plugin | longtext | NO | | | |

| authentication\_string | longtext | NO | | | |

| password\_expired | varchar(1) | NO | | | |

| is\_role | varchar(1) | YES | | NULL | |

| default\_role | longtext | NO | | | |

| max\_statement\_time | decimal(12,6) | NO | | 0.000000 | |

+------------------------+---------------------+------+-----+----------+-------+

47 rows in set (0.016 sec)

MariaDB [(none)]> **SELECT User, Db, Host from mysql.db;**

Empty set (0.001 sec)

MariaDB [(none)]> Ctrl-C -- exit!

Aborted

## 

## 

## 

## 