

# MariaDB

## Installation

```
apk add mariadb mariadb-common mariadb-client
```

## Initialisation

Il faut lancer la commande pour initialiser le service et créer la base SQL de départ.

```
alpine:~# rc-service mariadb setup
* Creating a new MySQL database ...
Installing MariaDB/MySQL system tables in '/var/lib/mysql' ...
OK
```

To start mysqld at boot time you have to copy  
support-files/mysql.server to the right place for your system

Two all-privilege accounts were created.

One is root@localhost, it has no password, but you need to  
be system 'root' user to connect. Use, for example, sudo mysql  
The second is mysql@localhost, it has no password either, but  
you need to be the system 'mysql' user to connect.

After connecting you can set the password, if you would need to be  
able to connect as any of these users with a password and without sudo

See the MariaDB Knowledgebase at <http://mariadb.com/kb> or the  
MySQL manual for more instructions.

You can start the MariaDB daemon with:

```
cd '/usr' ; /usr/bin/mysqld_safe --datadir='/var/lib/mysql'
```

You can test the MariaDB daemon with mysql-test-run.pl

```
cd '/usr/mysql-test' ; perl mysql-test-run.pl
```

Please report any problems at <http://mariadb.org/jira>

The latest information about MariaDB is available at <http://mariadb.org/>.

You can find additional information about the MySQL part at:

<http://dev.mysql.com>

Consider joining MariaDB's strong and vibrant community:

<https://mariadb.org/get-involved/>

[ ok ]

## Inscription du service au boot

```
alpine:~# rc-update add mariadb default
```

```
* service mariadb added to runlevel default
```

## Lancement du service

```
alpine:~# rc-service mariadb start
```

```
* Starting mariadb ...
```

```
200606 17:17:43 mysqld_safe Logging to syslog.
```

```
200606 17:17:43 mysqld_safe Starting mysqld daemon with databases from
```

```
/var/lib/mysql
```

```
[ ok ]
```

## Configuration

Il y a une petite subtilité, c'est **mariadb-secure-installation** même si *mysql\_secure\_installation* est toujours présent.

```
alpine:~# mariadb-secure-installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

In order to log into MariaDB to secure it, we'll need the current password for the root user. If you've just installed MariaDB, and haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):

OK, successfully used password, moving on...

Setting the root password or using the unix\_socket ensures that nobody can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix\_socket authentication [Y/n] y

Enabled successfully!

Reloading privilege tables..

... Success!

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y

New password:

Re-enter new password:

Password updated successfully!

Reloading privilege tables..

... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? [Y/n] y

... Success!

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y

... Success!

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n] y

- Dropping test database...

... Success!

- Removing privileges on test database...

... Success!

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? [Y/n] y

... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!

---

Revision #1

Created 6 June 2020 17:05:04 by Nidouille

Updated 6 June 2020 17:29:35 by Nidouille