**How to install a MariaDB/MySql service.**

Last updated on [July 24, 2018](https://www.certdepot.net/rhel7-install-mariadbmysql/) (53,067 views) - CertDepot — [7 Comments ↓](https://www.certdepot.net/rhel7-install-mariadbmysql/#comments)

Share this link

[Twitter0](https://twitter.com/share?original_referer=https%3A%2F%2Fwww.certdepot.net%2F&source=tweetbutton&text=RHEL7%3A+How+to+install+a+MariaDB%2FMySql+service.&url=https%3A%2F%2Fwww.certdepot.net%2Frhel7-install-mariadbmysql%2F&via=CertDepot)

[LinkedIn0](https://www.linkedin.com/cws/share?url=https://www.certdepot.net/rhel7-install-mariadbmysql/)

Note: This is an [RHCE 7 exam objective](https://www.certdepot.net/rhel7-rhce-exam-objectives/).

**Presentation of MariaDB**

**MariaDB** is a binary replacement for **MySql**, the famous database system.

**Prerequisites**

As the default **MariaDB** installation uses the **/var/lib/mysql** directory to store your databases, keep in mind that the partition or logical volume associated with **/var** needs adequate space.

**Installation Procedure**

To install it, apply the following steps:

Install the **MariaDB** packages:

# **yum install -y mariadb mariadb-server**

Start and activate at boot the **MariaDB** service:

# **systemctl start mariadb** && **systemctl enable mariadb**

Execute the basic setup:

# **mysql\_secure\_installation**

/usr/bin/mysql\_secure\_installation: line 379: find\_mysql\_client: command not found

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

you haven't set the root password yet, the password will be blank,

so you should just press enter here.

Enter current password for root (enter for none):

OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB

root user without the proper authorisation.

Set 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!

If you need to access the database from a different server, add the **MySql** service to the firewall configuration and reload it:

# **firewall-cmd --permanent --add-service=mysql**

success

# **firewall-cmd --reload**

success

The [Centminmod website](https://community.centminmod.com/) warns of [MariaDB restart issues after RHEL 7.4/CentOS 7.4 upgrade](https://community.centminmod.com/threads/mariadb-mysql-not-restarting-after-centos-7-4-update-fix.12863/).

**Initial Configuration**

To help you define the initial configuration according to the specifications of your system (memory size, number of CPU, type of activity, etc), you can find useful examples of **my.cnf** files in the **mariadb-server** package:

# **rpm -ql mariadb-server | grep my-**

/usr/share/mysql/my-huge.cnf

/usr/share/mysql/my-innodb-heavy-4G.cnf

/usr/share/mysql/my-large.cnf

/usr/share/mysql/my-medium.cnf

/usr/share/mysql/my-small.cnf

Start by reading the **/usr/share/mysql/my-innodb-heavy-4G.cnf** file because it is full of useful comments.

Then, download [Major Hayden’s mysqltuner.pl](https://github.com/major/MySQLTuner-perl). After running this script, you will get a rough evaluation of your configuration and basic recommentations to improve it.

The **innotop** package contains a command with the same name that behaves like the **top** command **on MariaDB** servers, providing a lot of details about the current activity (cache, locks, replication status, etc). This package is available in the **EPEL** repository.

**Configuration Tip**

To get an explanation of all the server options, all the configuration parameters and their current values, type:

# **/usr/libexec/mysqld –verbose –help**

/usr/libexec/mysqld Ver 5.5.52-MariaDB for Linux on x86\_64 (MariaDB Server)

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Starts the MariaDB database server.

Usage: /usr/libexec/mysqld [OPTIONS]

Default options are read from the following files in the given order:

/etc/mysql/my.cnf /etc/my.cnf ~/.my.cnf

The following groups are read: mysqld server mysqld-5.5 mariadb mariadb-5.5 client-server

The following options may be given as the first argument:

--print-defaults Print the program argument list and exit.

--no-defaults Don't read default options from any option file.

--defaults-file=# Only read default options from the given file #.

--defaults-extra-file=# Read this file after the global files are read.

--allow-suspicious-udfs

Allows use of UDFs consisting of only one symbol xxx()

without corresponding xxx\_init() or xxx\_deinit(). That

also means that one can load any function from any

library, for example exit() from libc.so

-a, --ansi Use ANSI SQL syntax instead of MySQL syntax. This mode

will also set transaction isolation level 'serializable'.

--archive[=name] Enable or disable ARCHIVE plugin. Possible values are ON,

OFF, FORCE (don't start if the plugin fails to load).

...

--xtradb-admin-command[=name]

Enable or disable XTRADB\_ADMIN\_COMMAND plugin. Possible

values are ON, OFF, FORCE (don't start if the plugin

fails to load).

Variables (--variable-name=value)

and boolean options {FALSE|TRUE} Value (after reading options)

------------------------------------------------- ------------------------

allow-suspicious-udfs FALSE

archive ON

aria ON

aria-block-size 8192

aria-checkpoint-interval 30

aria-checkpoint-log-activity 1048576

aria-force-start-after-recovery-failures 0

aria-group-commit none

...

verbose TRUE

wait-timeout 28800

xtradb-admin-command ON

To see what values a running MySQL server is using, type

'mysqladmin variables' instead of 'mysqld --verbose --help'.