
Installing MariaDB Debian Packages on Ubuntu 10.10

Peter Lavin

Table of Contents

Packages to Install Prior to the MariaDB .debs	1
The MariaDB Packages	1
Post-Installation	3
About the Author	4

This article is a follow-up article to [Installing MariaDB on Ubuntu 10.10 \(Maverick Meerkat\)](#). The Debian packages for Ubuntu 10.10 are now available. It may seem like overkill to document package installation. However, the MariaDB packages aren't available from a repository so installation is not simply a matter of adding an entry to the `/etc/apt/sources.list` file and then using your package manager. You'll need to download the required packages and install them locally. You'll also need additional packages that aren't available on the MariaDB mirrors. This article documents those steps.

Packages to Install Prior to the MariaDB .debs

The MariaDB packages (like the MySQL packages) require the Perl database interface common to all databases and the database interface to the MySQL database. You'll also need `psmisc` a collection of utilities that use the `proc` file system. You can install these packages using your package manager or go to the command line and issue the command:

```
shell> sudo apt-get install libdbi-perl libdbd-mysql-perl psmisc
```

The MariaDB Packages

Navigate to <http://askmonty.org/wiki/MariaDB:Download#Packages> and find a link to the Meerkat package suitable for your architecture. Choose your mirror and then select the `binary` directory. You should see something similar to the following:

Figure 1. Debian package files

	Parent Directory			-
	Packages.gz	06-Dec-2010	06:14	3.2K
	libmariadbclient-dev_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	1.6M
	libmariadbclient16_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	1.4M
	libmariadbdev_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	6.1M
	libmysqlclient16_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	4.2K
	mariadb-5.2_5.2.4-mariadb94.dsc	06-Dec-2010	06:14	1.4K
	mariadb-5.2_5.2.4-mariadb94.tar.gz	06-Dec-2010	06:14	24M
	mariadb-5.2_5.2.4-mariadb94_amd64.changes	06-Dec-2010	06:14	6.0K
	mariadb-client-5.2_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	1.7M
	mariadb-client-core-5.2_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	91K
	mariadb-client_5.2.4-mariadb94_all.deb	06-Dec-2010	06:14	4.1K
	mariadb-server-5.2_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	13M
	mariadb-server-core-5.2_5.2.4-mariadb94_amd64.deb	06-Dec-2010	06:14	4.2M
	mariadb-server_5.2.4-mariadb94_all.deb	06-Dec-2010	06:14	4.2K
	mariadb-test-5.2_5.2.4-mariadb94_all.deb	06-Dec-2010	06:14	9.3M
	mariadb-test_5.2.4-mariadb94_all.deb	06-Dec-2010	06:14	4.0K
	mysql-common_5.2.4-mariadb94_all.deb	06-Dec-2010	06:14	10K

As already noted, the various mirrors aren't package repositories. You'll need to download the packages you need and install them locally. You can download all the files (with the exception of Packages.gz) and install them all, however the required files are:

- libmysqlclient16_5.2.4-mariadb94_amd64.deb
- libmariadbclient16_5.2.4-mariadb94_amd64.deb
- mariadb-client-5.2_5.2.4-mariadb94_amd64.deb
- mariadb-client_5.2.4-mariadb94_all.deb
- mariadb-server_5.2.4-mariadb94_all.deb
- mariadb-server-5.2_5.2.4-mariadb94_amd64.deb
- mysql-common_5.2.4-mariadb94_all.deb
- mariadb-client-core-5.2_5.2.4-mariadb94_amd64.deb
- mariadb-server-core-5.2_5.2.4-mariadb94_amd64.deb

Note: The 64 bit files are shown above. Be sure to choose the files suitable to your architecture.

Once the files have been downloaded, navigate to your download directory and use the Debian package manager command **dpkg**:

```
shell> sudo dpkg --install mysql-common_5.1.42-mariadb73_all.deb
shell> sudo dpkg --install libmariadbclient16_5.1.42-mariadb73_amd64.deb \
mariadb-client-core-5.2_5.2.4-mariadb94_amd64.deb \
libmysqlclient16_5.1.42-mariadb73_amd64.deb \
mariadb-client_5.1.42-mariadb73_all.deb \
mariadb-client-5.1_5.1.42-mariadb73_amd64.deb \
mariadb-server_5.1.42-mariadb73_all.deb \
mariadb-server-core-5.2_5.2.4-mariadb94_amd64.deb \
mariadb-server-5.1_5.1.42-mariadb73_amd64.deb
```

During installation you will be asked to add a password for the root user.

Use the following command to confirm that the files have been installed:

```
shell> apt-cache search maria
```

You should see output similar to the following:

```
...
maria-doc - documentation of Maria
mariadb-client-5.2 - MariaDB database client binaries
mariadb-server-5.2 - MariaDB database server binaries
mariadb-server - MariaDB database server
mariadb-client - MariaDB database client
mariadb-server-core-5.2 - MariaDB database core server files
mysql-common - MariaDB database common files (e.g. /etc/mysql/my.cnf)
libmariadbclient16 - MariaDB database client library
libmysqlclient16 - Virtual package to satisfy external depends
mariadb-client-core-5.2 - MariaDB database core client binaries
```

Post-Installation

When you install MariaDB using the Debian packages, the `mysql` client and other utilities are installed to the `/usr/bin` directory, a directory that is in the default path, so there is no need to add a directory to your `PATH` variable. You can use the shell command `which mysql` to confirm this.

Also, the package installer configures the database server to start automatically and runs the **`mysql_secure_installation`** script. However, you may want to add a user and grant that user privileges. If you do this you'll probably also want to add a `.my.cnf` file. To perform these tasks see the Post-Installation section of [Installing MariaDB on Ubuntu 10.10 \(Maverick Meerkat\)](#).

If you want remote access to your server there is another change that you will have to make. The package files install the `/etc/mysql/my.cnf` configuration file with the `bind-address` option set to `127.0.0.1`. With this setting, attempting remote access results in the following error:

```
shell> mysql -u username -h ip_or_hostname -p
Enter password:
ERROR 2003 (HY000): Can't connect to MySQL server on 'ip_or_hostname' (61)
```

Change the `bind-address` option to `0.0.0.0` and, providing you've created a user with remote access, you'll be able to log in to your server from anywhere. (You will of course, also need the **`mysql`** client

installed on the remote computer.) After making this change don't forget to restart the server. You can do this by issuing the command `sudo /etc/init.d/mysql restart`.

About the Author

Peter Lavin is a technical writer who has been published in a number of print and online magazines. He is the author of Object Oriented PHP, published by No Starch Press and a contributor to PHP Hacks by O'Reilly Media.

Please do not reproduce this article in whole or part, in any form, without obtaining written permission.