

# **MariaDB Installation, Sanitation, and Verification on CentOS Stream 9**

## **Overview**

This project documents the installation, configuration, sanitation, and validation of MariaDB on a CentOS Stream 9 system. The workflow covers repository setup, package installation with dnf, service management using systemctl, execution of initial sanitation steps, and final version verification to confirm the database server is operational.

## **What Was Accomplished**

Configured a MariaDB YUM repository file under /etc/yum.repos.d/ to support database package management.

Installed the MariaDB server and client packages, with dependencies resolved through the AppStream repository.

Started and enabled the MariaDB service, verifying it is running and ready to accept SQL connections.

Ran the MariaDB secure installation process to review and apply sanitation options, including authentication and default database considerations.

Verified the installed MariaDB version using the MySQL client to confirm proper installation and access.

## **Result**

MariaDB 10.5.27 is successfully installed and running on the system. Core sanitation steps were reviewed during setup, and the database service is stable, enabled at boot, and accessible from the command line.

## **Outcome**

The system is now prepared for continued development or further hardening, demonstrating hands-on experience with Linux package management, database service control, and initial sanitation practices in an enterprise-style environment.

A custom MariaDB 10.7.8 YUM repository configuration is being edited under /etc/yum.repos.d/MariaDB.repo, defining the repository name, archive base URL, GPG key location, and enabling GPG signature verification to ensure secure package installation.

```
egarrido@dev-app-eg3:~ + v [mariadb] name = MariaDB-10.7.8  
baseurl= http://archive.mariadb.org/mariadb-10.7.8/yum/centos/$releasever/$basearch  
gpgkey= https://archive.mariadb.org/PublicKey  
gpgcheck=1  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~/etc/yum.repos.d/MariaDB.repo" 5L, 174B
```

The terminal output shows an initial failure to locate MariaDB-server and MariaDB-client, followed by a warning that the custom /etc/yum.repos.d/MariaDB.repo could not be loaded and was skipped. As a result, dnf falls back to the CentOS Stream 9 appstream repository and proceeds to install the default MariaDB 10.5.27 packages (mariadb, mariadb-server) along with required, weak, and SELinux-related dependencies.

```
egarrido@dev-app-eg3:~$ sudo dnf install mariadb-server mariadb -y
Error: Unable to find a match: MariaDB-server MariaDB-client
[egarrido@dev-app-eg3 ~]$ sudo dnf install mariadb-server mariadb -y
Warning: failed loading '/etc/yum.repos.d/MariaDB.repo', skipping.
Last metadata expiration check: 1:50:52 ago on Fri 26 Sep 2025 10:38:56 AM EDT.
Dependencies resolved.

Package          Architecture Version       Repository      Size
=====
Installing:
mariadb           x86_64      3:10.5.27-1.el9    appstream     1.6 M
mariadb-server    x86_64      3:10.5.27-1.el9    appstream     9.8 M
Installing dependencies:
mariadb-common    x86_64      3:10.5.27-1.el9    appstream     30 k
mariadb-connector-c x86_64      3.2.6-1.el9      appstream     198 k
mariadb-connector-c-config x86_64      3.2.6-1.el9      appstream     11 k
mariadb-errmsg     x86_64      3:10.5.27-1.el9    appstream     219 k
mysql-selinux      noarch      1.0.14-1.el9      appstream     36 k
perl-DBD-MariaDB   x86_64      1.21-17.el9      appstream     148 k
perl-DBI            x86_64      1.643-9.el9      appstream     725 k
perl-File-Copy      noarch      2.34-483.el9      appstream     20 k
perl-Math-BigInt    noarch      1:1.9998.18-460.el9 appstream     190 k
perl-Math-Complex   noarch      1.59-483.el9      appstream     46 k
perl-Sys-Hostname   x86_64      1.23-483.el9      appstream     17 k
Installing weak dependencies:
mariadb-backup      x86_64      3:10.5.27-1.el9    appstream     6.5 M
mariadb-gssapi-server x86_64      3:10.5.27-1.el9    appstream     15 k
mariadb-server-utils x86_64      3:10.5.27-1.el9    appstream     215 k
```

MariaDB 10.5 is successfully installed, enabled, and actively running on the system. The mariadb service has started without errors, initialized its data directory, passed socket and upgrade checks, and is now ready to accept SQL connections, confirming a healthy database server state.

```
[egarrido@dev-app-eg3 yum.repos.d]$ sudo systemctl status mariadb
● mariadb.service - MariaDB 10.5 database server
  Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; preset: disabled)
  Active: active (running) since Fri 2025-09-26 12:42:35 EDT; 6s ago
    Docs: man:mariadb(8)
          https://mariadb.com/kb/en/library/systemd/
   Process: 7290 ExecStartPre=/usr/libexec/mariadb-check-socket (code=exited, status=0/SUCCESS)
   Process: 7312 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir mariadb.service (code=exited, s>
   Process: 7406 ExecStartPost=/usr/libexec/mariadb-check-upgrade (code=exited, status=0/SUCCESS)
 Main PID: 7394 (mariadb)
   Status: "Taking your SQL requests now ... "
     Tasks: 11 (limit: 4605)
    Memory: 94.0M
      CPU: 1.248s
     CGroup: /system.slice/mariadb.service
             └─7394 /usr/libexec/mariadb --basedir=/usr

Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: The second is mysql@localhost>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: you need to be the system >
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: After connecting you can s>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: able to connect as any of >
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: See the MariaDB Knowledgeb>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: Please report any problems>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: The latest information abo>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: Consider joining MariaDB's>
Sep 26 12:42:34 dev-app-eg3.procore.prod1 mariadb-prepare-db-dir[7351]: https://mariadb.org/get-in>
Sep 26 12:42:35 dev-app-eg3.procore.prod1 systemd[1]: Started MariaDB 10.5 database server.
[egarrido@dev-app-eg3 yum.repos.d]$
```

The MariaDB secure installation process has completed successfully. During the interactive hardening steps, remote root login was left enabled, the default test database was retained, and privilege tables were not reloaded. The script finished without errors, confirming the MariaDB installation is operational and the security configuration steps were intentionally skipped based on the selected responses.

```
egarrido@dev-app-eg3:/etc/ ... 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] n
... skipping.

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] n
... skipping.

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

Reload privilege tables now? [Y/n] n
... skipping.

Cleaning up ...

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

Thanks for using MariaDB!
[egarrido@dev-app-eg3 yum.repos.d]$
```

The mysql -V command confirms that the MySQL client is installed and reporting MariaDB 10.5.27 on a 64-bit Linux system, indicating the server and client versions are aligned and properly accessible from the command line.

```
[egarrido@dev-app-eg3 yum.repos.d]$ mysql -V
mysql Ver 15.1 Distrib 10.5.27-MariaDB, for Linux (x86_64) using EditLine wrapper
[egarrido@dev-app-eg3 yum.repos.d]$
```

## Summary

MariaDB was successfully installed and validated on a CentOS Stream 9 system. The process included repository configuration, package installation, service startup and enablement, execution of initial sanitation steps through the secure installation workflow, and verification of the installed version. The database server is running, accessible from the command line, and ready for further configuration or application integration.