A Quick Guide to Using the MySQL Yum Repository

Abstract

The MySQL Yum repository provides RPM packages for installing the MySQL server, client, and other components on Linux platforms. The packages also upgrade and replace any third-party MySQL packages installed from the Linux distros' native software repositories, if replacements for them are available from MySQL.

The MySQL Yum repository supports the following Linux distributions:

- EL6 (8.0), EL7, EL8, EL9, and EL10-based platforms (for example, the corresponding versions of Oracle Linux, Red Hat Enterprise Linux, and CentOS)

- Fedora 41 and 42

> **Note**
>
> MySQL Server 8.0 and 8.4 are supported, while MySQL Server 5.7 (as of v5.7.30) is not. The MySQL Innovation Series is also supported.

> **Note**
>
> Not all versions of MySQL are supported on all these Linux distributions. See Selecting a Release Series on how to determine if your Linux distribution supports a particular version.

This is a quick guide to using the MySQL Yum repository. For more information, see Further Readings.

For legal information, see the Legal Notices.

For help with using MySQL, please visit the MySQL Forums, where you can discuss your issues with other MySQL users.

Document generated on: 2025-12-09 (revision: 84099)

Table of Contents

Steps for a Fresh Installation of MySQL

> **Note**
>
> The following instructions assume that MySQL is not already installed on your system using a third-party-distributed RPM package; if that is not the case, follow the instructions given in Replacing a Native Third-Party Distribution of MySQL.

Adding¹the MySQL Yum Repository

First, add the MySQL Yum repository to your system's repository list. Follow these steps:

a. Go to the download page for MySQL Yum repository configuration files at https://dev.mysql.com/downloads/repo/yum/.

   Although this is not required for each update, it does update MySQL repository information to include the current information. For example, when a new LTS series is added.

   The beginning part of the configuration file name, such as `mysql84`, describes the default MySQL series that is enabled for installation. In this case, the subrepository for MySQL 8.4 LTS is enabled by default. It also contains other subrepository versions, such as MySQL 8.0 and the MySQL Innovation Series.

b. Select and download the release package for your platform.

c. Install the downloaded release package with the following command, for example:

```
$> sudo yum localinstall platform-and-version-specific-package-name.rpm
```

   For example, for version n of the package for EL8-based systems, the command is:

```
$> sudo yum localinstall mysql84-community-release-el8-n.noarch.rpm
```

> **Note**
>
> Once the release package is installed on your system, any system-wide update by the `yum update` command (or `dnf upgrade` for dnf-enabled systems) will automatically upgrade MySQL packages on your system and also replace any native third-party packages, if Yum finds replacements for them in the MySQL Yum repository. See Upgrading MySQL with the MySQL Yum Repository and Replacing a Native Third-Party Distribution of MySQL for details.

Selecting a Release Series

When using the MySQL Yum repository, the latest MySQL LTS series is selected for installation by default. If this is what you want, you can skip to the next step, Installing MySQL with Yum.

Within the MySQL Yum repository (https://repo.mysql.com/yum/), different release series of the MySQL Community Server are hosted in different subrepositories. The subrepository for the latest LTS series (currently MySQL 8.4) is enabled by default, and the subrepositories for all other series (for example, the Innovation track or MySQL 8.0) are disabled by default. Use this command to see all the subrepositories in the MySQL Yum repository and to note which of them are enabled or disabled (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> yum repolist all | grep mysql
```

The latest LTS series is installed by default. To install from a different series, such as the innovation track or MySQL 8.0, disable the LTS series subrepository and enable the innovation track subrepository before executing installation commands. If your platform supports the `yum-config-manager` or `dnf config-manager` command, here are examples that disables the 8.4 series subrepository for both Server and Tools, for both depending on your desire; pick one:

Change MySQL 8.4 to 8.0

```
$> sudo yum-config-manager --disable mysql-8.4-lts-community
$> sudo yum-config-manager --disable mysql-tools-8.4-lts-community

$> sudo yum-config-manager --enable mysql80-community
$> sudo yum-config-manager --enable mysql-tools-community
```

For dnf-enabled platforms:

```
$> sudo dnf config-manager --disable mysql-8.4-lts-community
$> sudo dnf config-manager --disable mysql-tools-8.4-lts-community

$> sudo dnf config-manager --enable mysql80-community
$> sudo dnf config-manager --enable mysql-tools-community
```

Change MySQL 8.4 to the Innovation Series

```
$> sudo yum-config-manager --disable mysql-8.4-lts-community
$> sudo yum-config-manager --disable mysql-tools-8.4-lts-community

$> sudo yum-config-manager --enable mysql-innovation-community
$> sudo yum-config-manager --enable mysql-tools-innovation-community
```

For dnf-enabled platforms:

```
$> sudo dnf config-manager --disable mysql-8.4-lts-community
$> sudo dnf config-manager --disable mysql-tools-8.4-lts-community

$> sudo dnf config-manager --enable mysql-innovation-community
$> sudo dnf config-manager --enable mysql-tools-innovation-community
```

Besides using `yum-config-manager` or the `dnf config-manager` command, you can also select a series by editing the `/etc/yum.repos.d/mysql-community.repo` file. An example entry:

```
[mysql80-community]
name=MySQL 8.0 Community Server
baseurl=http://repo.mysql.com/yum/mysql-8.0-community/el/9/$basearch/
enabled=1
gpgcheck=1
```

```
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql-2023
```

Find the entry for the subrepository you want to configure, and edit the `enabled` option. Specify `enabled=0` to disable a subrepository, or `enabled=1` to enable a subrepository. For example, to install MySQL 8.0, make sure you have `enabled=0` for the other MySQL series entries and `enabled=1` for MySQL 8.0 as demonstrated above.

You should only enable subrepository for one release series at any time.

Verify that the correct subrepositories have been enabled and disabled by running the following command and checking its output (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> yum repolist enabled | grep mysql
```

Disabling the Default MySQL Module

(EL8 systems only) EL8-based systems such as RHEL8 and Oracle Linux 8 include a MySQL module that is enabled by default. Unless this module is disabled, it masks packages provided by MySQL repositories. To disable the included module and make the MySQL repository packages visible, use the following command (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> sudo yum module disable mysql
```

Installing MySQL

Install MySQL by the following command (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> sudo yum install mysql-community-server
```

This installs the package for the MySQL server, as well as other required packages.

Starting the MySQL Server

Start the MySQL server with the following command:

```
$> systemctl start mysqld
```

You can check the status of the MySQL server with the following command:

```
$> systemctl status mysqld
```

If the operating system is systemd enabled, standard `systemctl` (or alternatively, `service` with the arguments reversed) commands such as `stop`, `start`, `status`, and `restart` should be used to manage the MySQL server service. The `mysqld` service is enabled by default, and it starts at system reboot. See Managing MySQL Server with systemd for additional information.

MySQL Server Initialization (as of MySQL 5.7): At the initial start up of the server, the following happens, given that the data directory of the server is empty:

- The server is initialized.

- An SSL certificate and key files are generated in the data directory.

- The validate_password plugin is installed and enabled.

- A superuser account `'root'@'localhost'` is created. A password for the superuser is set and stored in the error log file. To reveal it, use the following command:

```
$> sudo grep 'temporary password' /var/log/mysqld.log
```

Change the root password as soon as possible by logging in with the generated, temporary password and set a custom password for the superuser account:

```
$> mysql -uroot -p
```

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'MyNewPass4!';
```

> **Note**
>
> MySQL's validate_password plugin is installed by default. This will require that passwords contain at least one uppercase letter, one lowercase letter, one digit, and one special character, and that the total password length is at least 8 characters.

> **Note**
>
> *For EL7-based platforms:* See Compatibility Information for EL7-based platforms.

Installing Additional MySQL Products and Components

You can use Yum to install and manage individual components of MySQL. Some of these components are hosted in subrepositories of the MySQL Yum repository. Use the following command to list the packages for all the MySQL components available for your platform from all subrepositories in the MySQL Yum repository (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> yum --disablerepo=\* --enablerepo='mysql*-community*' list available
```

Install any packages of your choice with the following command, replacing `package-name` with name of the package (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> sudo yum install package-name
```

For example, to install MySQL Workbench on Fedora:

```
$> sudo dnf install mysql-workbench-community
```

Upgrading MySQL with the MySQL Yum Repository

> **Note**
>
> • Before performing any update to MySQL, follow carefully the instructions in Upgrading MySQL. Among other instructions discussed there, it is especially important to back up your database before the update.

Use the MySQL Yum repository to perform an in-place update (that is, replacing the old version and then running the new version using the old data files) for your MySQL installation by following these steps (they assume you have installed MySQL with the MySQL Yum repository or with an RPM package directly downloaded from MySQL Developer Zone's MySQL Download page; if that is not the case, following the instructions in Replacing a Native Third-Party Distribution of MySQL instead):

Selecting a Target Series

By default, the MySQL Yum repository updates MySQL to the latest version in the release series you have chosen during installation (see Selecting a Release Series for details) which means, for example, that an 8.0.x installation does *not* automatically update to an 8.4.x release. To update to another release series, you need to first disable the subrepository for the series that has been selected

(by default, or by yourself) and enable the subrepository for your target series. To do that, see the general instructions given in Selecting a Release Series for editing the subrepository entries in the `/etc/yum.repos.d/mysql-community.repo` file. For upgrading from MySQL 8.0 to 8.4, disable the subrepository for the MySQL 8.0 series and enable that for the MySQL 8.4 series.

> **Important**
>
> - For important information about upgrading from MySQL 5.7 to 8.0, see Upgrading from MySQL 5.7 to 8.0.
>
> - For important information about upgrading from MySQL 8.0 to 8.4, see Upgrading from MySQL 8.0 to 8.4.
>
> - In-place downgrading of MySQL is not supported by the MySQL Yum repository. Follow the instructions in Downgrading MySQL.

Upgrading MySQL

Upgrade MySQL and its components by the following command, for platforms that are not dnf-enabled:

```
$> sudo yum update mysql-server
```

For dnf-enabled systems:

```
$> sudo dnf --refresh upgrade mysql-server
```

Alternatively, you can update MySQL by telling Yum to update everything on your system, which might take considerably more time. For platforms that are not dnf-enabled:

```
$> sudo yum update
```

> **Note**
>
> If the update fails due to an expired GPG key, import the most recent key using the following command:
>
> ```
> rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2023
> ```

For dnf-enabled systems:

```
$> sudo dnf upgrade
```

Restarting MySQL

The MySQL server always restarts after an update by Yum. Prior to MySQL 8.0.16, run `mysql_upgrade` after the server restarts to check and possibly resolve any incompatibilities between the old data and the upgraded software. `mysql_upgrade` also performs other functions; see mysql_upgrade — Check and Upgrade MySQL Tables for details. As of MySQL 8.0.16, this step is not required, as the server performs all tasks previously handled by `mysql_upgrade`.

You can also update only a specific component. Use the following command to list all the installed packages for the MySQL components (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> sudo yum list installed | grep "^mysql"
```

After identifying the package name of the component of your choice, update the package with the following command, replacing `package-name` with the name of the package. For platforms that are not dnf-enabled:

```
$> sudo yum update package-name
```

For dnf-enabled systems:

```
$> sudo dnf upgrade package-name
```

Replacing a Native Third-Party Distribution of MySQL

To replace third-party distributions of MySQL that were installed from the supported Linux platforms' native software repositories with the latest bug release (from the MySQL 8.4 series currently) from the MySQL Yum repository, follow these steps:

Backing Up Your Database

To avoid loss of data, always back up your database before trying to replace your MySQL installation using the MySQL Yum repository. See Backup and Recovery on how to back up your database.

Adding the MySQL Yum Repository

Add the MySQL Yum repository to your system's repository list by following the instructions given in Adding the MySQL Yum Repository.

Replacing the Native Third-Party Distribution by a Yum Update or a DNF Upgrade

By design, the MySQL Yum repository will replace your native, third-party MySQL with the latest bugfix release (from the MySQL 8.4 series currently) from the MySQL Yum repository when you perform a `yum update` command (or `dnf upgrade` for dnf-enabled systems) on the system, or a `yum update mysql-server` (or `dnf upgrade mysql-server` for dnf-enabled systems).

After updating MySQL using the Yum repository, applications compiled with older versions of the shared client libraries should continue to work. However, *if you want to recompile applications and dynamically link them with the updated libraries*, see Upgrading the Shared Client Libraries for some special considerations.

> **Notes**
>
> - *For EL7-based platforms:* See Compatibility Information for EL7-based platforms.
>
> - If you have a third-party distribution of MySQL that you have downloaded and installed from a nonnative repository (for example, from MariaDB or Percona), it is important to follow the instructions for replacing them given in the MySQL server's reference manual.

Installing MySQL NDB Cluster Using the Yum Repository

> **Notes**
>
> - The following instructions assume that neither the MySQL server nor MySQL NDB Cluster has already been installed on your system; if that is not the case, remove the MySQL server or MySQL NDB Cluster, including all its executables, libraries, configuration files, and data directories, before you continue. However there is no need to remove the release package you might have used to enable the MySQL Yum repository on your system.
>
> - The NDB Cluster SQL node package has a dependency on the Perl `Class::MethodMaker` module. Yum can take care of this dependency if the

> EPEL (Extra Packages for Enterprise Linux) repository has been enabled on your
> system; see the instructions here for how to enable the EPEL repository.
>
> • The MySQL Yum repository supports installation of MySQL NDB Cluster
>   for release 7.5.6 and later. For other methods of installing NDB Cluster, see
>   Installation of NDB Cluster on Linux, Installation of NDB Cluster on Linux, or
>   Installation of NDB Cluster on Linux, depending on the release series that you are
>   using.

Adding¹the MySQL Yum Repository for MySQL NDB Cluster

Follow the steps in Adding the MySQL Yum Repository to add the MySQL Yum repository to your
system's repository list. If you have already performed the step before, make sure you have the most
up-to-date version of the release package by running the following command, for platforms that are not
dnf-enabled:

```
$> sudo yum update mysql-8.4-lts-community-release
```

For dnf-enabled systems:

```
$> sudo dnf --refresh upgrade mysql-8.4-lts-community-release
```

Selecting the MySQL NDB Cluster Subrepository

Within the MySQL Yum repository (https://repo.mysql.com/yum/), the MySQL Community Server
and MySQL NDB Cluster are hosted in different subrepositories. By default, the subrepository for the
latest bugfix series of the MySQL server is enabled and the subrepository for MySQL NDB Cluster
is disabled. To install NDB Cluster, disable the subrepository for the MySQL server and enable the
subrepository for NDB Cluster. If your platform supports the yum-config-manager or dnf config-
manager command, you can do that by issuing, for example, the following commands, which disable
the subrepository for the MySQL 8.4 series and enable the one for the MySQL NDB Cluster 8.4 for
platforms that are not dnf-enabled:

```
$> sudo yum-config-manager --disable mysql-8.4-lts-community
$> sudo yum-config-manager --enable mysql-cluster-8.4-community
```

For dnf-enabled platforms:

```
$> sudo dnf config-manager --disable mysql-8.4-lts-community
$> sudo dnf config-manager --enable mysql-cluster-8.4-community
```

To install MySQL NDB Cluster 8.0 instead of MySQL NDB Cluster 8.4, enable mysql-cluster-8.0-
community instead of mysql-cluster-8.4-community.

To install MySQL NDB Cluster innovation series instead of MySQL NDB Cluster 8.4, enable mysql-
cluster-innovation-community instead of mysql-cluster-8.4-community.

Besides using yum-config-manager or the dnf config-manager command, you can also select
a subrepository by editing manually the /etc/yum.repos.d/mysql-community.repo file. This is a
sample entry for the MySQL 8.4 subrepository in the file:

> **Note**
>
> The baseurl value in the /etc/yum.repos.d/mysql-community.repo
> file may differ slightly by platform.

```
[mysql-8.4-lts-community]
```

```
name=MySQL 8.4 Community Server
baseurl=http://repo.mysql.com/yum/mysql-8.4-community/el/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql-2023
```

To install NDB Cluster 8.4, you must disable the MySQL 8.4 subrepository by making `enabled=0` for the above subrepository entry and enable the NDB Cluster 8.4 subrepository by making `enabled=1` for its entry:

```
[mysql-cluster-8.4-community]
name=MySQL Cluster 8.4 Community
baseurl=http://repo.mysql.com/yum/mysql-cluster-8.4-community/el/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql-2023
```

After saving the changes to the file, verify that the correct subrepositories have been enabled by running the following command (for dnf-enabled systems, replace `yum` in the command with `dnf`):

```
$> yum repolist enabled | grep mysql
!mysql-cluster-8.4-community/x86_64 MySQL Cluster 8.4 Community          18
!mysql-connectors-community/x86_64  MySQL Connectors Community           31
!mysql-tools-8.4-lts-community/x86_64    MySQL Tools Community                33
```

The subrepository for NDB Cluster 8.4 (Community edition) has now been enabled. Also in the list are a number of other subrepositories of the MySQL Yum repository that have been enabled by default.

Installing MySQL NDB Cluster

For a minimal installation of MySQL NDB Cluster, follow these steps (for dnf-enabled systems, replace `yum` in the commands with `dnf`):

• Install the components for SQL nodes:

```
$> sudo yum install mysql-cluster-community-server
```

After the installation is completed, start and initialize the SQL node by following the steps given in Starting the MySQL Server.

If you choose to initialize the data directory manually using the `mysqld --initialize` command (see Initializing the Data Directory for details), a `root` password is going to be generated and stored in the SQL node's error log; see MySQL Server Initialization [3] for how to find the password, and for a few things you need to know about it.

• Install the executables for management nodes:

```
$> sudo yum install mysql-cluster-community-management-server
```

• Install the executables for data nodes:

```
$> sudo yum install mysql-cluster-community-data-node
```

Configuring and Starting MySQL NDB Cluster

See Initial Configuration of NDB Cluster on how to configure MySQL NDB Cluster and Initial Startup of NDB Cluster on how to start it for the first time.

> **Note**
>
> *For EL7-based platforms:* See Compatibility Information for EL7-based platforms.

Installing Additional MySQL NDB Cluster Products and Components

You can use Yum to install individual components and additional products of MySQL NDB Cluster from the MySQL Yum repository. To do that, assuming you already have the MySQL Yum repository on your system's repository list (if not, follow Step 1 and 2 of Installing MySQL NDB Cluster Using the Yum Repository), follow the same steps given in Installing Additional MySQL Products and Components.

> **Note**
>
> *Known issue:* Currently, not all components required for running the MySQL NDB Cluster test suite are installed automatically when you install the test suite package (`mysql-cluster-community-test`). Install the following packages with `yum install` (or `dnf install` for dnf-enabled systems) before you run the test suite:
>
> - `mysql-cluster-community-auto-installer`
>
> - `mysql-cluster-community-management-server`
>
> - `mysql-cluster-community-data-node`
>
> - `mysql-cluster-community-memcached`
>
> - `mysql-cluster-community-java`
>
> - `mysql-cluster-community-ndbclient-devel`

Platform Specific Notes

ARM Support

ARM 64-bit (aarch64) is supported on Oracle Linux 7 and requires the Oracle Linux 7 Software Collections Repository (ol7_software_collections). For example, to install the server:

```
$> yum-config-manager --enable ol7_software_collections
$> yum install mysql-community-server
```

Further Readings

More information on the MySQL Yum repository can be found at the following sections in the MySQL server's reference manual:

- Installing MySQL on Linux Using the MySQL Yum Repository

- Replacing a Third-Party Distribution of MySQL Using the MySQL Yum Repository

- Upgrading MySQL with the MySQL Yum Repository

Legal Notices

Copyright © 1997, 2025, Oracle and/or its affiliates.

License Restrictions

affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Use of This Documentation

This documentation is NOT distributed under a GPL license. Use of this documentation is subject to the following terms:

You may create a printed copy of this documentation solely for your own personal use. Conversion to other formats is allowed as long as the actual content is not altered or edited in any way. You shall not publish or distribute this documentation in any form or on any media, except if you distribute the documentation in a manner similar to how Oracle disseminates it (that is, electronically for download on a Web site with the software) or on a CD-ROM or similar medium, provided however that the documentation is disseminated together with the software on the same medium. Any other use, such as any dissemination of printed copies or use of this documentation, in whole or in part, in another publication, requires the prior written consent from an authorized representative of Oracle. Oracle and/or its affiliates reserve any and all rights to this documentation not expressly granted above.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.