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:

- EL5-, EL6-, and EL7-based platforms (for example, the corresponding versions of Oracle Linux, Red Hat Enterprise Linux, and CentOS)
- Fedora 24 and 25

> **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 distrituion 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 either the MySQL Forums or MySQL Mailing Lists, where you can discuss your issues with other MySQL users.

For additional documentation on MySQL products, including translations of the documentation into other languages, and downloadable versions in variety of formats, including HTML and PDF formats, see the MySQL Documentation Library.

Document generated on: 2017-02-20 (revision: 50836)

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.

Adding the 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 at http://dev.mysql.com/downloads/repo/yum/.

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

c. Install the downloaded release package with the following command, replacing *platform-and-version-specific-package-name* with the name of the downloaded package:

```
shell> sudo rpm -Uvh platform-and-version-specific-package-name.rpm
```

For example, for version *n* of the package for EL6-based systems, the command is:

```
shell> sudo rpm -Uvh mysql57-community-release-el6-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 GA release of MySQL 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 (http://repo.mysql.com/yum/), different release series of the MySQL Community Server are hosted in different subrepositories. The subrepository for the latest GA series (currently MySQL 5.7) is enabled by default, and the subrepositories for all other series (for example, the MySQL 5.6 series) are disabled by default. Use this command to see all the subrepositories in the MySQL Yum repository, and see which of them are enabled or disabled (for dnf-enabled systems, replace `yum` in the command with `dnf`):

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

To install the latest release from the latest GA series, no configuration is needed. To install the latest release from a specific series other than the latest GA series, disable the subrepository for the latest GA series and enable the subrepository for the specific series before running the installation command. You can do that by editing manually the `/etc/yum.repos.d/mysql-community.repo` file. This is a typical entry for a release series' subrepository in the file:

```
[mysql57-community]
name=MySQL 5.7 Community Server
baseurl=http://repo.mysql.com/yum/mysql-5.7-community/el/6/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
```

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 5.6, make sure you have `enabled=0` for the above subrepository entry for MySQL 5.7, and have `enabled=1` for the entry for the 5.6 series:

```
# Enable to use MySQL 5.6
[mysql56-community]
name=MySQL 5.6 Community Server
baseurl=http://repo.mysql.com/yum/mysql-5.6-community/el/6/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
```

You should only enable subrepository for one release series at any time. When subrepositories for more than one release series are enabled, the latest series will be used by Yum.

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`):

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

Installing MySQL

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

```
shell> 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:

```
shell> sudo service mysqld start
```

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

```
shell> sudo service mysqld status
```

For MySQL 5.7 only: 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:

```
shell> 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:

```
shell> 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 upper case letter, one lower case letter, one digit, and one special character, and that the total password length is at least 8 characters.

Securing the MySQL Installation (for MySQL 5.6 only)

The program `mysql_secure_installation` allows you to perform important operations like setting the root password, removing anonymous users, and so on. Always run it to secure your MySQL 5.6 installation:

```
shell> mysql_secure_installation
```

It is important to remember the root password you set. See `mysql_secure_installation` — Improve MySQL Installation Security for details.

Do not run `mysql_secure_installation` after an installation of MySQL 5.7, as the function of the program has already been performed by the Yum repository installation.

> **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`):

```
shell> 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`):

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

For example, to install MySQL Workbench on Fedora 24:

```
shell> 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 off 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

1

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, a 5.6.x installation will NOT be updated to a 5.7.x release automatically. 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 5.6 to 5.7, perform the *reverse* of the steps illustrated in Selecting a Release Series, disabling the subrepository for the MySQL 5.6 series and enabling that for the MySQL 5.7 series.

As a general rule, to upgrade from one release series to another, go to the next series rather than skipping a series. For example, if you are currently running MySQL 5.5 and wish to upgrade to 5.7, upgrade to MySQL 5.6 first before upgrading to 5.7.

> **Important**
>
> - For important information about upgrading from MySQL 5.5 to 5.6, see Upgrading from MySQL 5.5 to 5.6.
> - For important information about upgrading from MySQL 5.6 to 5.7, see Upgrading from MySQL 5.6 to 5.7.
> - For important information about upgrading from MySQL 5.7 to 8.0, see Upgrading from MySQL 5.7 to 8.0.
> - In-place downgrading of MySQL is not supported by the MySQL Yum repository. Follow the instructions in Downgrading MySQL.

Upgrading MySQL

2

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

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

For dnf-enabled systems:

```
shell> 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:

```
shell> sudo yum update
```

For dnf-enabled systems:

```
shell> sudo dnf upgrade
```

3. Restarting MySQL

The MySQL server always restarts after an update by Yum. Once the server restarts, run `mysql_upgrade` 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.

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`):

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

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

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

For dnf-enabled systems:

```
shell> 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 GA release (from the MySQL 5.7 series currently) from the MySQL Yum repository, follow these steps:

1. 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.

2. 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.

3. 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 GA release (from the MySQL 5.7 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.

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

International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its 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.

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

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.

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.