A Quick Guide to Using the MySQL SLES Repository

Abstract

This is a quick guide for using the MySQL SLES repository to install and upgrade MySQL on SUSE Linux Enterprise Server (SLES).

The MySQL SLES repository supports SLES 12 (for MySQL 5.6 and later) and SLES 11 (from 11.3 onward, for MySQL 5.5 and later), and only for the x86-64 architecture.

> **Note**
>
> The MySQL SLES repository service is now in development release. We encourage you to try it and provide us with feedback. Please report any bugs or inconsistencies you observe to our Bugs Database.

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: 2016-06-02 (revision: 47898)

Table of Contents

- Steps for a Fresh Installation of MySQL
- Installing Additional MySQL Products and Components
- Upgrading MySQL with the MySQL SLES Repository
- Replacing MySQL Installed by an RPM from Other Sources

Steps for a Fresh Installation of MySQL

> **Note**
>
> The following instructions assume that no versions of MySQL (whether distributed by Oracle or other parties) have already been installed on your system; if that is not the case, follow instead the instructions given in Replacing MySQL Installed by an RPM from Other Sources.

1 Adding the MySQL SLES Repository

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

a. Go to the download page for MySQL SLES repository at http://dev.mysql.com/downloads/repo/suse/.

b. Select and download the release package for your SLES version.

The MySQL SLES repository supports SLES 12 (for MySQL 5.6 and later) and SLES 11 (from 11.3 onward, for MySQL 5.5 and later), and only for the x86-64 architecture.

c. Install the downloaded release package with the following command, replacing *`package-name`* with the name of the downloaded package:

```
shell> sudo rpm -Uvh package-name.rpm
```

For example, for version *n* of the package for SLES 11, the command is:

```
shell> sudo rpm -Uvh mysql-community-release-sles11-n.noarch.rpm
```

For version *n* of the package for SLES 12, the command is:

```
shell> sudo rpm -Uvh mysql57-community-release-sles12-n.noarch.rpm
```

2 Importing MySQL GnuPG Key

Import into the system the GnuPG key for MySQL products, which will be used for checking signatures of the downloaded packages from the MySQL SLES repository, with the following command:

```
shell> sudo rpm --import /etc/RPM-GPG-KEY-mysql
```

3 Selecting a Release Series

Within the MySQL SLES repository, different release series of the MySQL Community Server are hosted in different subrepositories. For SLES 12, the subrepository for the latest GA series (currently MySQL 5.7) is enabled by default, and the subrepositories for all other series are disabled. For SLES 11, the subrepository for the MySQL 5.6 series is enabled by default, and the subrepositories for all other series are disabled. Use this command to see all the subrepositories in the MySQL SLES repository, and see which of them are enabled or disabled:

```
shell> zypper repos | grep mysql.*community
```

To install the latest release from a specific series, before running the installation command, make sure that the subrepository for the series you want is enabled and the subrepositories for other series are disabled. For example, on SLES 12, to disable the subrepository for MySQL 5.7, which is enabled by default, use the following command:

```
shell> sudo zypper modifyrepo -d mysql57-community
```

Then, enable the subrepository for the release series you want. For example, to enable the subrepository for MySQL 5.6:

```
shell> sudo zypper modifyrepo -e mysql56-community
```

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

Verify that the correct subrepositories have been enabled by running the following command and checking its output:

```
shell> zypper repos -E | grep mysql.*community
4 | mysql56-community                | MySQL 5.6 Community Server
```

After that, use the following command to refresh the repository information for the enabled subrepository:

```
shell> sudo zypper refresh
```

4 Installing MySQL with Zypper

Install MySQL by the following command:

```
shell> sudo zypper install mysql-community-server
```

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

5 Starting the MySQL Server

Start the MySQL server with the following command:

```
shell> sudo service mysql start
```

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

```
shell> sudo service mysql status
```

You can stop the MySQL server with the following command:

```
shell> sudo service mysql stop
```

6 Securing the MySQL Installation

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 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.

Installing Additional MySQL Products and Components

You can install more components of MySQL. List subrepositories in the MySQL SLES repository with the following command:

```
shell> zypper repos | grep mysql.*community
```

Use the following command to list the packages for the MySQL components available for a certain subrepository, changing *`subrepo-name`* to the name of the subrepository you are interested in :

```
shell> zypper packages subrepo-name
```

Install any packages of your choice with the following command, replacing *`package-name`* with name of the package (you might need to enable first the subrepository for the package, using the same method for selecting a subrepository for a specific release series outlined in Selecting a Release Series):

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

For example, to install the MySQL benchmark suite from the subrepository for the release series you have already enabled:

```
shell> sudo zypper install mysql-community-bench
```

Upgrading MySQL with the MySQL SLES 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 SLES repository to perform an in-place update (that is, replacing the old version of the server 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 SLES repository; if that is not the case, following the instructions in Replacing MySQL Installed by an RPM from Other Sources instead):

Selecting a Target Series

During an update operation, by default, the MySQL SLES 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, follow the general instructions given in Selecting a Release Series.

As a general rule, to upgrade from one release series to another, go to the next series rather than skipping a series.

> **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.

- In-place downgrading of MySQL is not supported by the MySQL SLES repository. Follow the instructions in Downgrading MySQL.

Upgrading MySQL

Upgrade MySQL and its components by the following command:

```
shell> sudo zypper update mysql-community-server
```

Alternatively, you can update MySQL by telling Zypper to update everything on your system (this might take considerably more time):

```
shell> sudo zypper update
```

Upgrading the Data

The MySQL server always restarts after an update by Zypper. 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.

Note

After upgrading from MySQL 5.6 to 5.7, if you have problem connecting to the server as `root` (which will make it impossible to run `mysql_upgrade`), stop the server and then restart it with the `--skip-grant-tables` option before you run `mysql_upgrade`. See Upgrading from MySQL 5.6 to 5.7 for details.

You can also update a specific component only. Use the following command to list all the installed packages from the MySQL SLES repository:

```
shell> sudo zypper packages -i | grep mysql-.*community
```

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:

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

Replacing MySQL Installed by an RPM from Other Sources

RPMs for installing the MySQL Community Server and its components can be downloaded from MySQL either from the MySQL Developer Zone, from the native software repository of SLES, or from the MySQL SLES repository. The RPMs from the those sources might be different, and they might install and configure MySQL in different ways.

If you have installed MySQL with RPMs from the MySQL Developer Zone or the native software repository of SLES and want to replace the installation using the RPM from the MySQL SLES repository, follow these steps:

1. Back up your database to avoid data loss. See Backup and Recovery on how to do that.
2. Stop your MySQL server, if it is running. If the server is running as a service, you can stop it with the following command:

   ```
   shell> sudo service mysql stop
   ```

3. Follow the steps given for Adding the MySQL SLES Repository.

4. Follow the steps given for Selecting a Release Series.

5. Follow the steps given for Installing MySQL with Zypper. You will be asked if you want to replace the old packages with the new ones; for example:

```
Problem: mysql-community-server-5.6.22-2.sles11.x86_64 requires mysql-community-client = 5.6.22-2.sl
  but this requirement cannot be provided uninstallable providers:
  mysql-community-client-5.6.22-2.sles11.x86_64[mysql56-community]
 Solution 1: replacement of mysql-client-5.5.31-0.7.10.x86_64 with mysql-community-client-5.6.22-2.s
 Solution 2: do not install mysql-community-server-5.6.22-2.sles11.x86_64
 Solution 3: break mysql-community-server-5.6.22-2.sles11.x86_64 by ignoring some of its dependencie

Choose from above solutions by number or cancel [1/2/3/c] (c)
```

   Choose the “replacement” option (“Solution 1” in the example) to finish your installation from the MySQL SLES repository.

6. If you have replaced your MySQL installation with a higher version from the MySQL SLES repository, follow the instructions given in Upgrading the Data.

Legal Notices

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.