Install MongoDB on Ubuntu

On this page

Overview

Use this tutorial to install MongoDB on LTS Ubuntu Linux systems from `.deb` packages. While Ubuntu includes its own MongoDB packages, the official MongoDB packages are generally more up-to-date.

> **PLATFORM SUPPORT:**
> MongoDB only provides packages for 64-bit long-term support Ubuntu releases. Currently, this means 12.04 LTS (Precise Pangolin) and 14.04 LTS (Trusty Tahr). While the packages may work with other Ubuntu releases, this is not a supported configuration.

Packages

MongoDB provides officially supported packages in their own repository. This repository contains the following packages:

- `mongodb-org`
  This package is a `metapackage` that will automatically install the four component packages listed below.
- `mongodb-org-server`

This package contains the `mongod` daemon and associated configuration and init scripts.

- `mongodb-org-mongos`
  This package contains the `mongos` daemon.
- `mongodb-org-shell`
  This package contains the `mongo` shell.
- `mongodb-org-tools`
  This package contains the following MongoDB tools: `mongoimport` `bsondump`, `mongodump`, `mongoexport`, `mongofiles`, `mongooplog`, `mongoperf`, `mongorestore`, `mongostat`, and `mongotop`.

Init Scripts

The `mongodb-org` package includes various init scripts, including the init script `/etc/init.d/mongod`. These scripts are used to stop, start, and restart daemon processes.

The package configures MongoDB using the `/etc/mongod.conf` file in conjunction with the init scripts. See the Configuration File reference for documentation of settings available in the configuration file.

As of version 3.0.13, there are no init scripts for `mongos`. The `mongos` process is used only in sharding. You can use the `mongod` init script to derive your own `mongos` init script for use in such environments. See the `mongos` reference for configuration details.

Considerations

MongoDB only provides packages for 64-bit long-term support Ubuntu releases. Currently, this means 12.04 LTS (Precise Pangolin) and 14.04 LTS (Trusty Tahr). While the packages may work with other Ubuntu releases, this is not a supported configuration.

You cannot install these packages concurrently with the `mongodb`, `mongodb-server`, or `mongodb-clients` packages provided by Ubuntu.

The default `/etc/mongod.conf` configuration file supplied by the 3.0 series packages has `bind_ip` set to `127.0.0.1` by default. Modify this setting as needed for your environment before initializing a replica set.

Changed in version 2.6: The package structure and names have changed as of version 2.6. For instructions on installation of an older release, please refer to the documentation for the appropriate version.

Install MongoDB

1 Import the public key used by the package management system.

The Ubuntu package management tools (i.e. `dpkg` and `apt`) ensure package consistency and authenticity by requiring that distributors sign packages with GPG keys. Issue the following command to import the MongoDB public GPG Key:

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 7F0CEB10
```

2 Create a list file for MongoDB.

Create the `/etc/apt/sources.list.d/mongodb-org-3.0.list` list file using the command appropriate for your version of Ubuntu:

Ubuntu 12.04

```
/mongodb-org/3.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.0.list
```

Ubuntu 14.04

```
/mongodb-org/3.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.0.list
```

3 Reload local package database.

Issue the following command to reload the local package database:

```
sudo apt-get update
```

4

Install the MongoDB packages.

You can install either the latest stable version of MongoDB or a specific version of MongoDB.

Install the latest stable version of MongoDB.

Issue the following command:

```
sudo apt-get install -y mongodb-org
```

Install a specific release of MongoDB.

To install a specific release, you must specify each component package individually along with the version number, as in the following example:

```
sudo apt-get install -y mongodb-org=3.0.13 mongodb-org-server=3.0.13 mongodb-org-she
```

If you only install `mongodb-org=3.0.13` and do not include the component packages, the latest version of each MongoDB package will be installed regardless of what version you specified.

Pin a specific version of MongoDB.

Although you can specify any available version of MongoDB, `apt-get` will upgrade the packages when a newer version becomes available. To prevent unintended upgrades, pin the package. To pin the version of MongoDB at the currently installed version, issue the following command sequence:

```
echo "mongodb-org hold" | sudo dpkg --set-selections
echo "mongodb-org-server hold" | sudo dpkg --set-selections
echo "mongodb-org-shell hold" | sudo dpkg --set-selections
echo "mongodb-org-mongos hold" | sudo dpkg --set-selections
echo "mongodb-org-tools hold" | sudo dpkg --set-selections
```

Versions of the MongoDB packages before 2.6 use a different repository location. Refer to the version of the documentation appropriate for your MongoDB version.

Run MongoDB

The MongoDB instance stores its data files in `/var/lib/mongodb` and its log files in `/var/log/mongodb` by default, and runs using the `mongodb` user account. You can specify alternate log and data file directories in `/etc/mongod.conf`. See `systemLog.path` and `storage.dbPath` for additional information.

If you change the user that runs the MongoDB process, you **must** modify the access control rights to the `/var/lib/mongodb` and `/var/log/mongodb` directories to give this user access to these directories.

1. **Start MongoDB.**

   Issue the following command to start `mongod`:

   ```
   sudo service mongod start
   ```

2. **Verify that MongoDB has started successfully**

   Verify that the `mongod` process has started successfully by checking the contents of the log file at `/var/log/mongodb/mongod.log` for a line reading

```
[initandlisten] waiting for connections on port <port>
```

where `<port>` is the port configured in `/etc/mongod.conf`, `27017` by default.

3 Stop MongoDB.

As needed, you can stop the `mongod` process by issuing the following command:

```
sudo service mongod stop
```

4 Restart MongoDB.

Issue the following command to restart `mongod`:

```
sudo service mongod restart
```

5 Begin using MongoDB.

To help you start using MongoDB, MongoDB provides Getting Started Guides in various driver editions. See Getting Started for the available editions.

Before deploying MongoDB in a production environment, consider the Production Notes document.

Later, to stop MongoDB, press `Control+C` in the terminal where the `mongod` instance is running.

Uninstall MongoDB

To completely remove MongoDB from a system, you must remove the MongoDB applications themselves, the configuration files, and any directories containing data and logs. The following section guides you through the necessary steps.

> **WARNING:**
> This process will *completely* remove MongoDB, its configuration, and *all* databases. This process is not reversible, so ensure that all of your configuration and data is backed up before proceeding.

1 Stop MongoDB.

Stop the `mongod` process by issuing the following command:

```
sudo service mongod stop
```

2 Remove Packages.

Remove any MongoDB packages that you had previously installed.

```
sudo apt-get purge mongodb-org*
```

3 Remove Data Directories.

Remove MongoDB databases and log files.

```
sudo rm -r /var/log/mongodb
sudo rm -r /var/lib/mongodb
```