Install MongoDB Community Edition on Ubuntu

On this page

Overview

Use this tutorial to install MongoDB 4.2 Community Edition on LTS (long-term support) releases of Ubuntu Linux using the `apt` package manager.

MongoDB Version

This tutorial installs MongoDB 4.2 Community Edition. To install a different version of MongoDB Community, use the version drop-down menu in the upper-left corner of this page to select the documentation for that version.

Considerations

Platform Support

EOL Notice

- MongoDB 4.2 Community Edition removes support for Ubuntu 14.04 (“Trusty”) on x86_64

MongoDB 4.2 Community Edition supports the following **64-bit** Ubuntu LTS (long-term support) releases on x86_64 architecture:

- 18.04 LTS (Long Term Support) (“Bionic”)
- 16.04 LTS (Long Term Support) (“Xenial”)

MongoDB only supports the 64-bit versions of these platforms.

MongoDB 4.2 Community Edition on Ubuntu also supports the ARM64 and s390x architectures on select platforms.

See Supported Platforms for more information.

> **WINDOWS SUBSYSTEM FOR LINUX (WSL) - UNSUPPORTED:**
> MongoDB does not support the Windows Subsystem for Linux (WSL).

Production Notes

Before deploying MongoDB in a production environment, consider the Production Notes document which offers performance considerations and configuration recommendations for production MongoDB deployments.

Official MongoDB Packages

To install MongoDB Community on your Ubuntu system, these instructions will use the official `mongodb-org` package, which is maintained and supported by MongoDB Inc. The official `mongodb-org` package always contains the latest version of MongoDB, and is available from its own dedicated repo.

> **IMPORTANT:**
> The `mongodb` package provided by Ubuntu is **not** maintained by MongoDB Inc. and conflicts with the official `mongodb-org` package. If you have already installed the `mongodb` package on your Ubuntu system, you **must** first uninstall the `mongodb` package before proceeding with these instructions.

Install MongoDB Community Edition

Follow these steps to install MongoDB Community Edition using the `apt` package manager.

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

From a terminal, issue the following command to import the MongoDB public GPG Key from https://www.mongodb.org/static/pgp/server-4.2.asc:

```
wget -qO - https://www.mongodb.org/static/pgp/server-4.2.asc | sudo apt-key add -
```

The operation should respond with an `OK`.

However, if you receive an error indicating that `gnupg` is not installed, you can:

1. Install `gnupg` and its required libraries using the following command:

   ```
   sudo apt-get install gnupg
   ```

2. Once installed, retry importing the key:

   ```
   wget -qO - https://www.mongodb.org/static/pgp/server-4.2.asc | sudo apt-key add
   ```

2 Create a list file for MongoDB.

Create the list file `/etc/apt/sources.list.d/mongodb-org-4.2.list` for your version of Ubuntu.

Ubuntu 18.04 (Bionic) | Ubuntu 16.04 (Xenial)

The following instruction is for **Ubuntu 18.04 (Bionic)**. For Ubuntu 16.04 (Xenial), click on the appropriate tab.

Create the `/etc/apt/sources.list.d/mongodb-org-4.2.list` file for Ubuntu 18.04 (Bionic):

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu bionic/mongodb-or
```

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 version of MongoDB. | Install a specific release of MongoDB.

To install the latest stable version, issue the following

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

package at the currently installed version:

```
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
```

For help with troubleshooting errors encountered while installing MongoDB on Ubuntu, see our troubleshooting guide.

Run MongoDB Community Edition

ulimit Considerations

Most Unix-like operating systems limit the system resources that a session may use. These limits may negatively impact MongoDB operation. See UNIX ulimit Settings for more information.

Directories

If you installed via the package manager, the data directory `/var/lib/mongodb` and the log directory `/var/log/mongodb` are created during the installation.

By default, MongoDB runs using the `mongodb` user account. If you change the user that runs the MongoDB process, you **must** also modify the permission to the data and log directories to give this user access to these directories.

Configuration File

The official MongoDB package includes a configuration file (`/etc/mongod.conf`). These settings (such as the data directory and log directory specifications) take effect upon startup. That is, if you change the configuration file while the MongoDB instance is running, you must restart the instance for the changes to take effect.

Procedure

using the default settings.

Init System

To run and manage your `mongod` process, you will be using your operating system's built-in init system. Recent versions of Linux tend to use **systemd** (which uses the `systemctl` command), while older versions of Linux tend to use **System V init** (which uses the `service` command).

If you are unsure which init system your platform uses, run the following command:

```
ps --no-headers -o comm 1
```

Then select the appropriate tab below based on the result:

- `systemd` - select the **systemd (systemctl)** tab below.
- `init` - select the **System V Init (service)** tab below.

systemd (systemctl) | **System V Init (service)**

1. **Start MongoDB.**

   You can start the `mongod` process by issuing the following command:

   ```
   sudo systemctl start mongod
   ```

   If you receive an error similar to the following when starting `mongod`:

   ```
   Failed to start mongod.service: Unit mongod.service not found.
   ```

   Run the following command first:

Then run the start command above again.

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

```
sudo systemctl status mongod
```

You can optionally ensure that MongoDB will start following a system reboot by issuing the following command:

```
sudo systemctl enable mongod
```

3 **Stop MongoDB.**

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

```
sudo systemctl stop mongod
```

4 **Restart MongoDB.**

You can restart the `mongod` process by issuing the following command:

```
sudo systemctl restart mongod
```

You can follow the state of the process for errors or important messages by watching the output in the `/var/log/mongodb/mongod.log` file.

5 **Begin using MongoDB.**

Start a `mongo` shell on the same host machine as the `mongod`. You can run the `mongo` shell without any command-line options to connect to a `mongod` that is running on your localhost with default port 27017:

For more information on connecting using the `mongo` shell, such as to connect to a `mongod` instance running on a different host and/or port, see The mongo Shell.

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

Uninstall MongoDB Community Edition

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*
```

Remove MongoDB databases and log files.

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

Additional Information

Localhost Binding by Default

By default, MongoDB launches with `bindIp` set to `127.0.0.1`, which binds to the localhost network interface. This means that the `mongod` can only accept connections from clients that are running on the same machine. Remote clients will not be able to connect to the `mongod`, and the `mongod` will not be able to initialize a replica set unless this value is set to a valid network interface.

This value can be configured either:

- in the MongoDB configuration file with `bindIp`, or
- via the command-line argument `--bind_ip`

WARNING:
Before binding to a non-localhost (e.g. publicly accessible) IP address, ensure you have secured your cluster from unauthorized access. For a complete list of security recommendations, see Security Checklist. At minimum, consider enabling authentication and hardening network infrastructure.

For more information on configuring `bindIp`, see IP Binding.

MongoDB Community Edition Packages

MongoDB Community Edition is available from its own dedicated repository, and contains the following officially-supported packages:

`mongodb-org`	A `metapackage` that will automatically install the four component packages listed below.
`mongodb-org-server`	Contains the `mongod` daemon, associated init script, and a configuration file (`/etc/mongod.conf`). You can use the initialization script to start `mongod` with the configuration file. For details, see Run MongoDB Community Edition.
`mongodb-org-mongos`	Contains the `mongos` daemon.
`mongodb-org-shell`	Contains the `mongo` shell.
`mongodb-org-tools`	Contains the following MongoDB tools: `mongoimport` `bsondump`, `mongodump`, `mongoexport`, `mongofiles`, `mongorestore`, `mongostat`, and `mongotop`.