How to Install MongoDB on Ubuntu 18.04

MongoDB is a free and open-source document database. It belongs to a family of databases called NoSQL, which is different from the traditional table-based SQL databases like MySQL and PostgreSQL.

In MongoDB, data is stored in flexible, JSON-like documents where fields can vary from document to document. It does not require a predefined schema, and data structure can be changed over time.

In this tutorial, we will cover the process of installing and configuring the latest version of MongoDB Community Edition on an Ubuntu 18.04 machine from the official MongoDB repositories.

1. Installing MongoDB

The following steps describe how to install MongoDB on your Ubuntu server:

1.1 Add the MongoDB GPG key to your system using the following command (make sure it is a single line):

sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 9DA31620334BD75D9DCB49F368818C72E52529D4

1.2 Once the key is imported, add the MongoDB repository with:

sudo add-apt-repository 'deb [arch=amd64] https://repo.mongodb.org/apt/ubuntu bionic/mongodb-org/4.0 multiverse'

If you get an error message saying add-apt-repository command not found, install the software-properties-common package.

1.3 Update the packages list and install the mongodb-org meta-package by typing:

sudo apt update

sudo apt install mongodb-org

The following packages will be installed on your system as a part of the mongodb-org package:

mongodb-org-server - The mongod daemon and corresponding init scripts and configurations.

mongodb-org-mongos - The mongos daemon.

mongodb-org-shell - The mongo shell is an interactive JavaScript interface to MongoDB. It is used to perform administrative tasks through the command line.

mongodb-org-tools - Contains several MongoDB tools for importing and exporting data, statistics, as well as other utilities.

2. Booting MongoDB

Once the installation is completed, start the MongoDB daemon and enable it to start on boot:

Before that, you need to create necessary folders:

2.1 create two folders

sudo mkdir /data

sudo mkdir /data/db

2.2 start mongoDB

sudo mongod --bind_ip=0.0.0.0

We need to bind address to “0.0.0.0”, thus remote systems will also be able to connect to the database.

Make sure there are no errors messages. If you see “waiting for connections on port 27017”, your mongoDB should be running all right.

Your current terminal should be trapped by the Mongod process; you can open another terminal to access your database.

Or you can start with “&” to run mongod as a background process:

sudo mongod --bind_ip=0.0.0.0 &

3. Validating MongoDB

3.1 To verify whether the installation has completed successfully, we will connect to the MongoDB database server using the **mongo** tool (i.e., a client tool) and print the connection status:

mongo --eval 'db.runCommand({ connectionStatus: 1 })'

```
MongoDB shell version v4.0.27
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
2021-10-21T17:08:54.809+0000 I NETWORK  [listener] connection accepted from 127
2021-10-21T17:08:54.810+0000 I NETWORK  [conn1] received client metadata from 1
oDB Internal Client", version: "4.0.27" }, os: { type: "Linux", name: "Ubuntu",
Implicit session: session { "id" : UUID("501f6f9f-38b7-4630-a166-5bfc11a1b757")
MongoDB server version: 4.0.27
{
        "authInfo" : {
                "authenticatedUsers" : [ ],
                "authenticatedUserRoles" : [ ]
        },
        "ok" : 1
}
2021-10-21T17:08:54.816+0000 I NETWORK  [conn1] end connection 127.0.0.1:45270
```

A value of 1 for the ok field indicates success.

3.2 Check mongodb service port number

Your mongodb is running and listening to a TCP port (e.g., 27017 by default). The live chat web server will connect to this port later.

To check this using:

sudo netstat -an | grep 27017

3.3 Play with your mongodb

You can further interact with mongodb using the interactive “mongo” tool: Simply tying down:

mongo

```
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---

>
```

Type down “help” for basic commands. For example,

3.3.1 you can use “show dbs” to display current databases in mongodb. There are three default databases.

3.3.2 you can use “use name_of_database” (e.g., use admin) to change to the view of one database

3.3.3 you can then use “show collections” to display all collections in the current database

3.3.4 you can then use “db.<name_of_collection>.find()” to show all objects in this collection

Please refer to these documents for commands details:

1. https://dzone.com/articles/top-10-most-common-commands-for-beginners

2. https://docs.mongodb.com/manual/reference/command/

Note that:

1) By default mongodb has no enabled access control, so there is no default user or password. But in a production environment, you need to enable proper access control (e.g., by creating usernames and passwords). For the simplicity of this project, we will not enable access control
2) While you connect the web server to the mongodb, please use the above commands to check what has been stored in the database for debugging or learning purposes.