Installing MongoDB

For the purposes of this work we focus on the *MongoDB Community Edition* for the simple reason that it's free of charge. This version is also an excellent way to get your feet wet so to speak, allowing you to learn about and experiment with MongoDB risk free. Before beginning installation be sure to check the minimum requirements for your operating system in the MongoDB installation manual: https://docs.mongodb.com/manual/installation/.

On a live server, in a commercial enterprise, it is recommended you use the *MongoDB Enterprise Advanced* version. You might also consider two cloud-based offerings, *MongoDB Atlas* or *MongoDB Stitch*. The former provides a cloud-hosted MongoDB database service. The latter builds upon the former, opening the MongoDB API so that your apps can make calls and receive responses.

Installing MongoDB on Windows

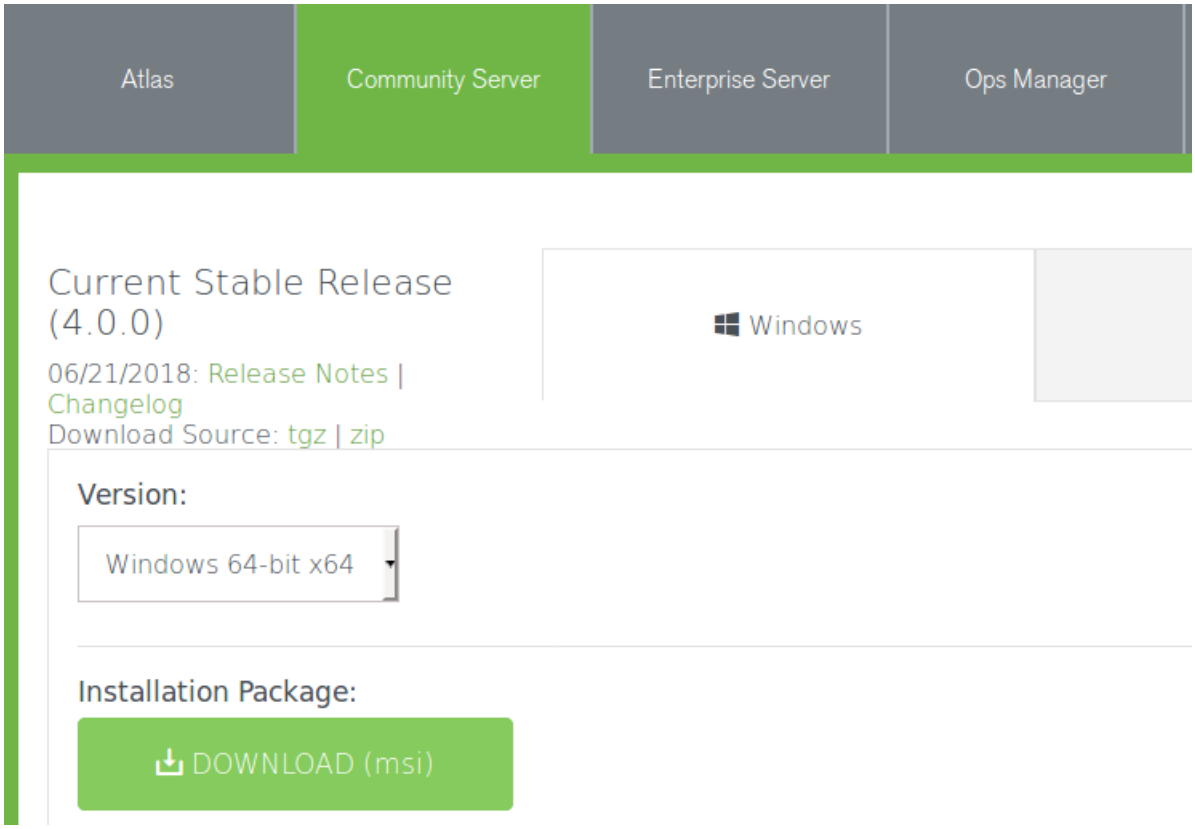
The version featured in this book is MongoDB 4.0. The minimum requirement for a Windows installation is *Windows Server 2008 R2*, *Windows 7*, or later.

The MongoDB reference manual warns that *Windows Server 2008 R2* or *Windows 7* require a hotfix patch be installed to prevent errors from occurring under certain conditions. For more information see: https://support.microsoft.com/en-gb/help/2731284/33-dos-error-code-when-memory-memory-mapped-files-are-cleaned-by-using.

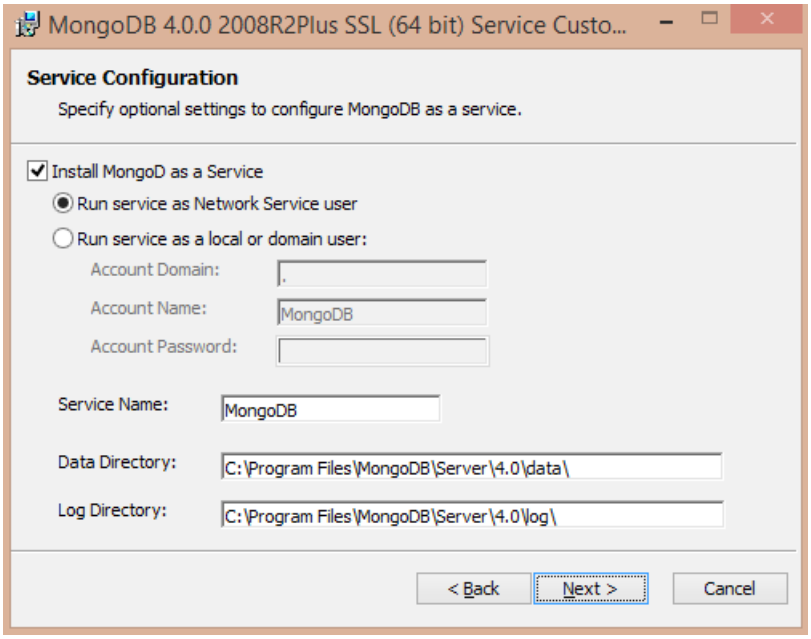
Download and Install

To download and install MongoDB on Windows, proceed as follows:

1. Go to the MongoDB Download center at: https://www.mongodb.com/download-center#community
2. Select the appropriate operating system where it says *Version*
3. Click on Download (msi)



4. When prompted, choose Save File
5. Click on the saved MSI file to start the installer
6. Click OK when the security prompt appears asking to Open Executable File*?*
7. Click Run when the security warning appears
8. Click Next to start the MongoDB Setup Wizard
9. Read the license agreement and click on the checkbox and Next. Note that if you do not accept the license agreement the installation will terminate.
10. When asked Choose Setup Type, for the purposes of this illustration, select Complete. *MongoDB Compass*, which is a handy utility which greatly facilitates database management, is automatically installed.
11. Now that all choices have been made, click on Install and click Yes when the User Account Control security warning pops up
12. As of MongoDB v4.0, the installation wizard lets you configure startup options. If you want to have MongoDB start automatically and run in the background, choose *Run server as Network Server user*. You can also configure the directory where MongoDB stores its data files (*Data Directory*), and where log files are stored (*Log Directory*).

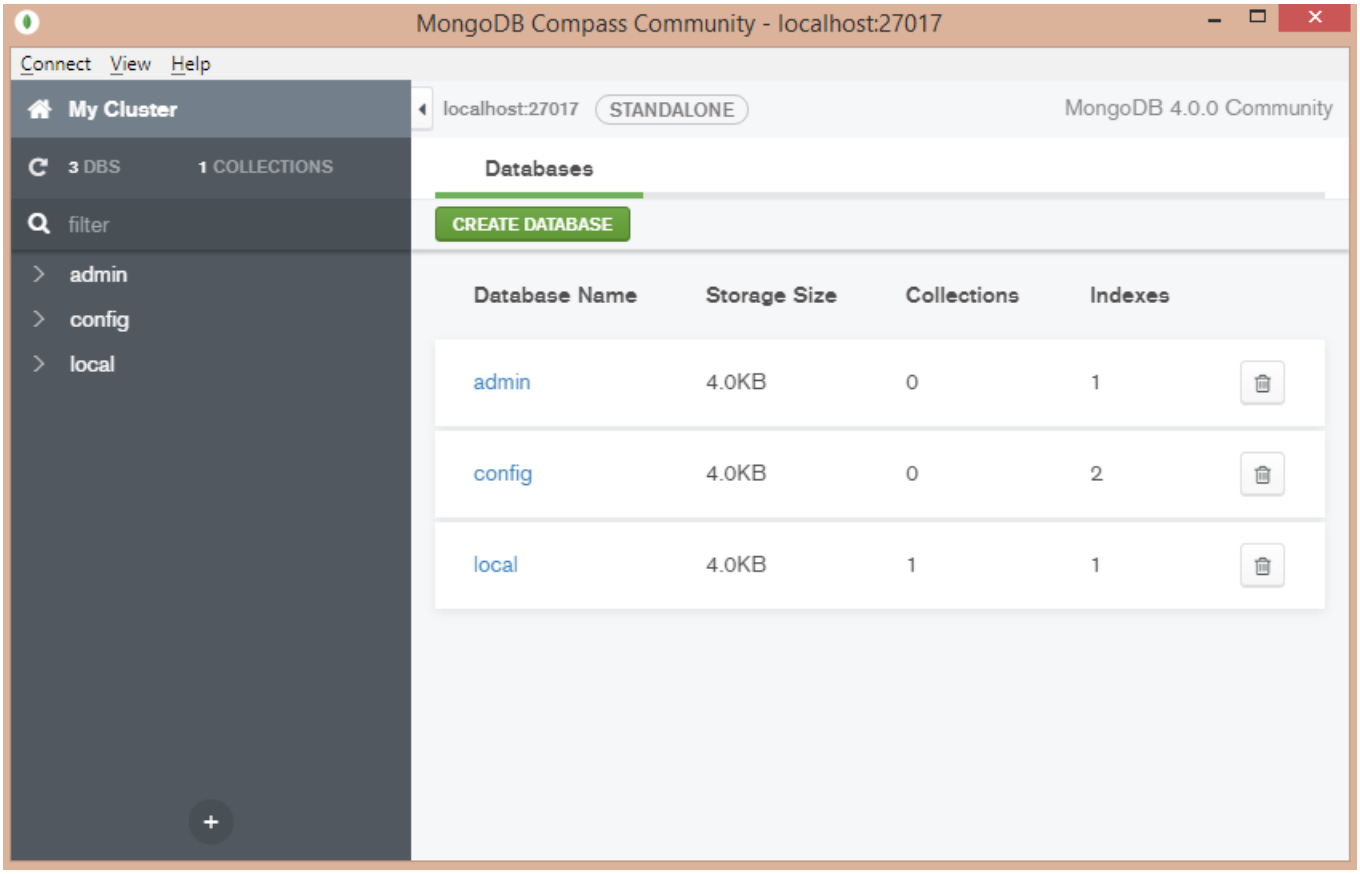


13. Click Next to continue and Finish when the installation completes

MongoDB Compass

Assuming you elected to install the complete package, *MongoDB Compass* will auto-launch once the installation completes. You will need to scroll down through its license agreement (separate from the license agreement for MongoDB itself), and click Agree. You can follow and then close the initial help tutorial, and also set Privacy Settings which control whether or not you will be sending crash reports, usage statistics, and requesting automatic updates to/from MongoDB Inc.

This utility is described in more details in the next chapter *Understanding MongoDB Data Structures*. We also use this utility to create our first database and collection (see below). Here is the *Compass* screen as seen on Windows.

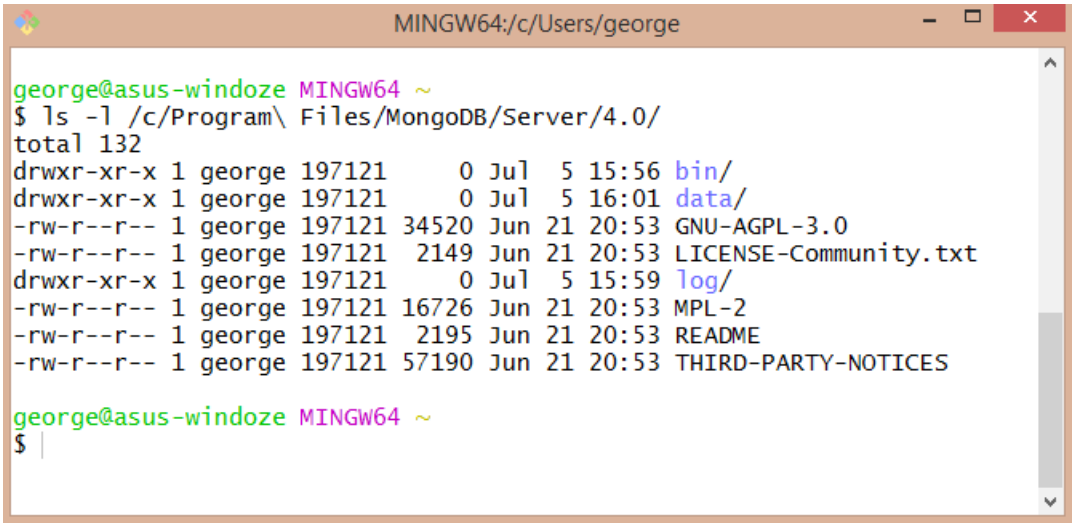


MongoDB Windows File Locations

If using the Windows MSI installer (recommended), the MongoDB program files will be stored here:

C:\Program Files\MongoDB\Server\<version>

You have the option, during the installation process, of specifying the location where the database and log files are stored. Once finished, here is a look at the new directory structure:

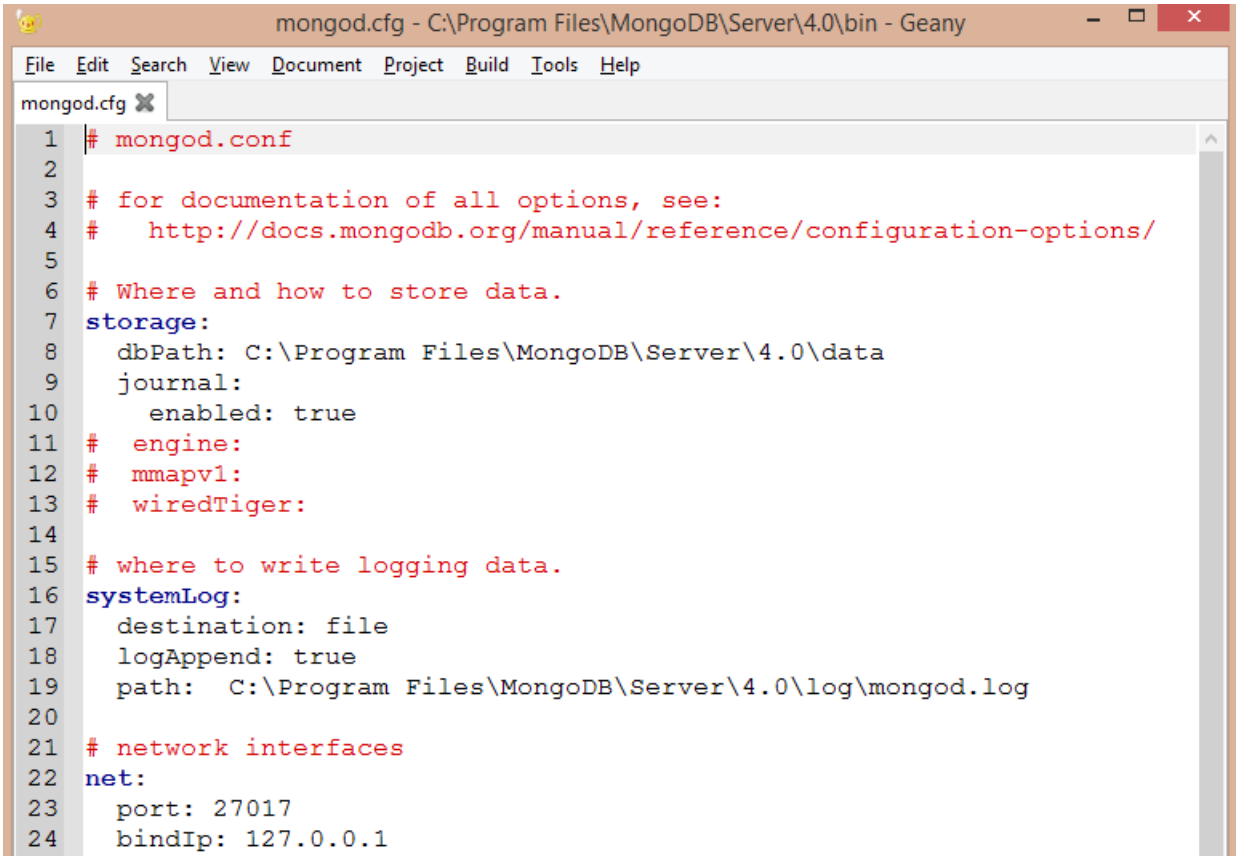


If you elected to install MongoDB as a service it starts automatically, and can be administered just as any Windows service.

The configuration file which contains the locations of the database and log files defaults to:

C:\Program Files\MongoDB\Server\4.0\bin\mongod.cfg

This file is automatically generated by the installer. By default here are its contents:



Installing MongoDB on Linux

It is important to understand the MongoDB installation process on Linux even if you are a developer or IT professional, and are not using Linux personally, it's extremely likely that the Internet-facing server you or your customer use is running Linux. W3Techs, a company which does web technology surveys, estimates that in 2018, the running on Linux was at 68.1% compared with 32% for Windows.

There are three primary considerations when installing MongoDB on Linux, each of which we will address in turn:

- Linux based upon Debian and Ubuntu
- Linux based upon RedHat, Fedora and CentOS
- Installing directly from source code

With the bewildering array of Linux distributions currently available, it is difficult to decide which version to feature for the purposes of demonstrating MongoDB on Linux. A significant number of Linux distributions are based on either *Debian* or *Red Hat* Linux. Accordingly, this section covers

installing MongoDB on both. A website which gives good insight on all reported Linux distributions is DistroWatch. Linux Mint although now extremely popular, was included here as it's Debian-based and not as commercially available as Ubuntu.

Installing on Debian or Ubuntu Linux

Debian Linux, self-described as the universal operating system, is a free open-source project which uses a fork of the Linux kernel, and draws heavily upon GNU (i.e. GNU Not Unix) software. Ubuntu Linux is produced by the Canonical Group Ltd based in South Africa, and is based upon Debian. For the purposes of this book, we will focus on *Ubuntu* version 18.04, code-named Bionic Beaver, released in April 2018, a designated LTS (Long Term Support) version.

The preferred way to install any given software on Ubuntu is to use a Debian *package*. Such packages have the extension `*.deb` and include a script which tells the package management program where to place the pre-compiled binary files as they are extracted. Popular package management programs include: *synaptic* (graphical interface, resolves dependencies, and does a lot of "housekeeping"), *aptitude* (like *synaptic* but has a textual, command-line menu), and *apt-** (i.e. *apt-get*, *apt-key* etc.: very fast, command-line only). For the purposes of this section we will use *apt-get*.

Ubuntu provides its own MongoDB package which is what gets installed if you simply run sudo apt-get install mongodb. To get the latest "official" version directly from MongoDB, you should follow the procedure outlined below. If you already have installed the Ubuntu "mongodb" package, you will need to first uninstall it before proceeding.

The MongoDB packages available for Ubuntu/Debian include the following:

`mongodb-org-server`	Primary MongoDB system daemon
`mongodb-org-mongos`	MongoDB "shard" routing service
`mongodb-org-shell`	MongoDB shell
`mongodb-org-tools`	Provides various mongo* tools for import, export, restore, etc.

In addition, a composite package mongodb-org which contains all four packages mentioned above is provided.

Package Installation

To install MongoDB on an Ubuntu/Debian server, you will need *root* access. A unique feature of Debian-based Linux distributions is that direct login as root is not allowed for security reasons. Accordingly, you can promote yourself to root using *su*, or you can precede the various commands with *sudo*, which instructs the OS to process this command as root. Please proceed as follows:

1. Import the public key from the MongoDB key server. This is needed so that the package manager can authenticate the MongoDB package:

2. Add the MongoDB repository to the Linux server's *sources* list:

IMPORTANT: the commands listed should be on one line. We use a backslash ("\") to indicate a line of text which is too long to fit the printed page. When typing the command, omit the backslash ("\") and do not hit enter until the command has been fully entered.

3. Refresh the package database from the sources list by running: sudo apt-get update.

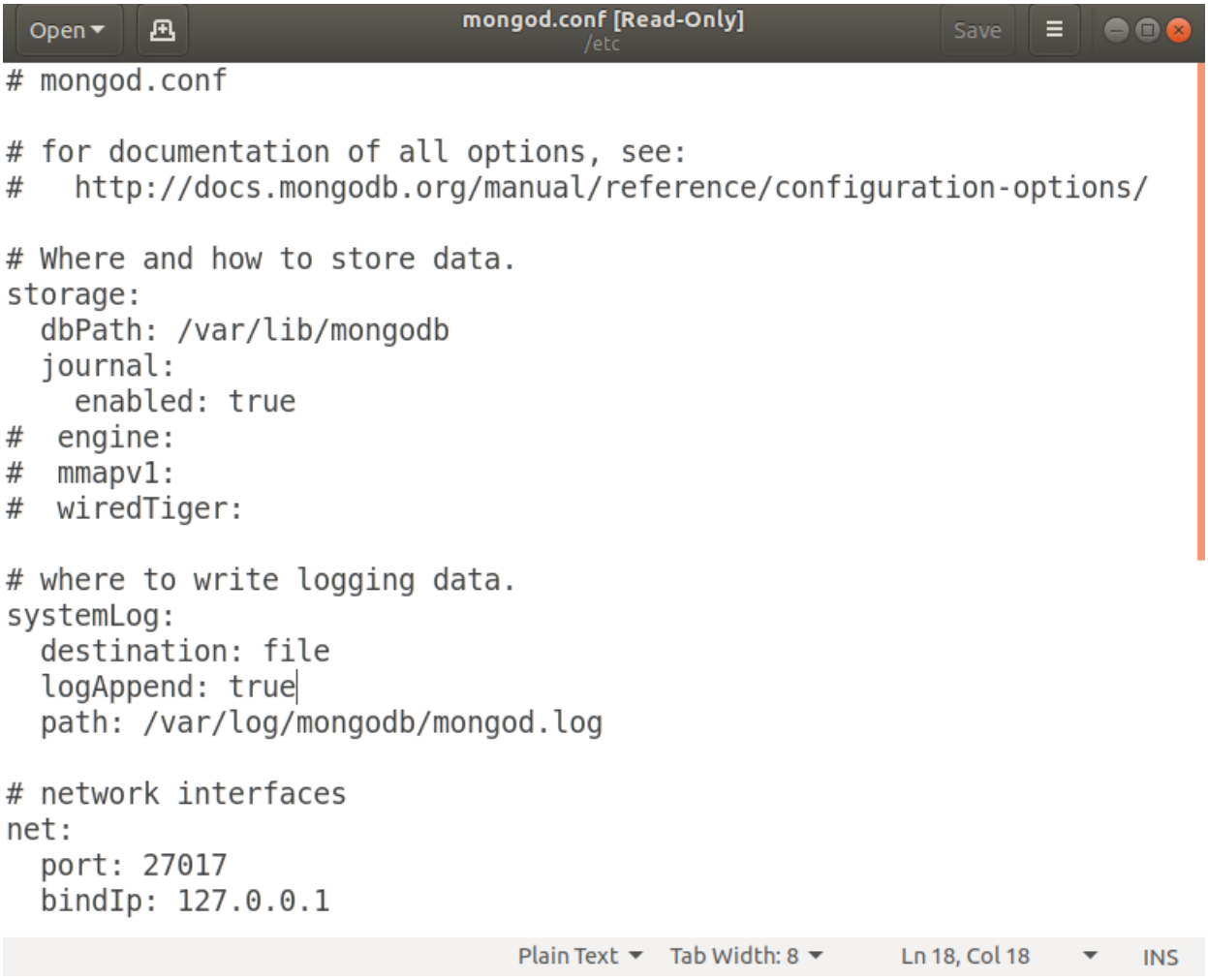
Ubuntu version 18.04 is code-named *bionic*. You will note this name is used in step #2 above where the MongoDB repository is added to the *sources* list. If this source is not found, you will receive an error message:
The repository ... bionic/mongodb-org/4.0 ... does not have a Release file
In this situation, substitute the code name `xenial` (Ubuntu 16.04) in place of bionic (Ubuntu 18.04).

4. Install the latest (stable) version of MongoDB. Here we install only the composite package which alleviates the need to separately install the four primary packages listed above:
   `sudo apt-get install -y mongodb-org`

You will note, at the end of the installation, that the installer creates a user mongodb who belongs to a group mongodb, which is also newly created. This is the system user MongoDB uses when it runs.

Configure and Run MongoDB on Ubuntu/Debian

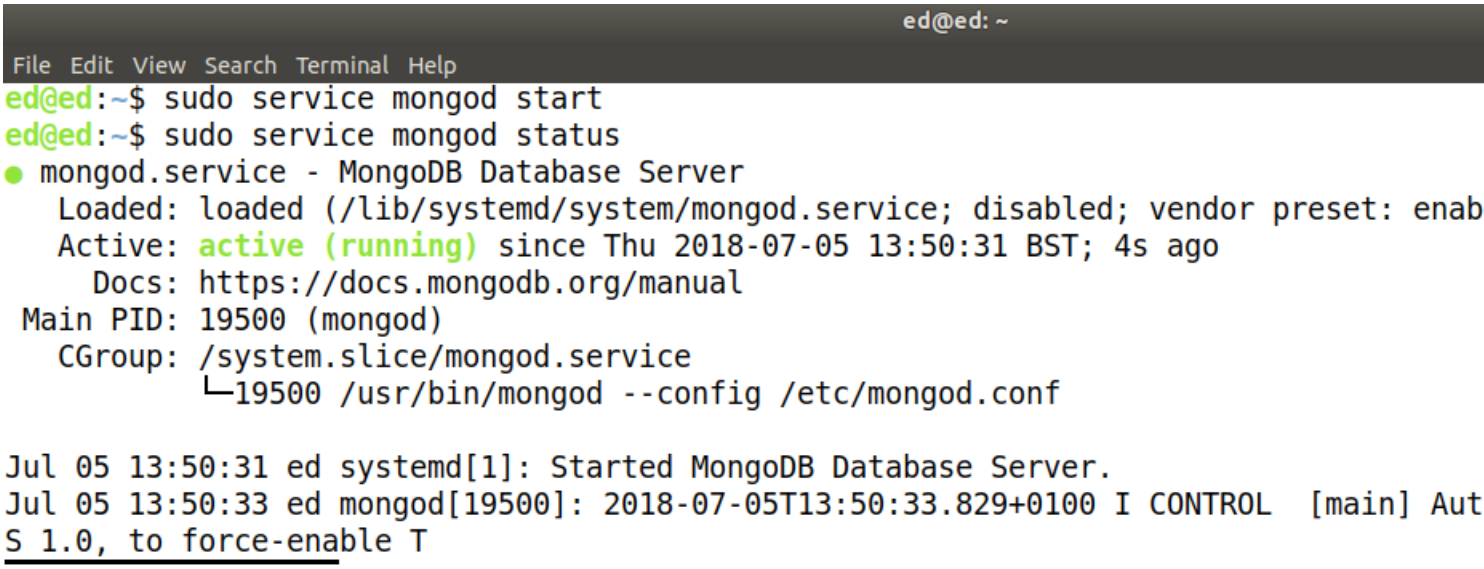
If you followed the procedure outlined above, a configuration file `/etc/mongod.conf` was auto-generated by the installation script. By default, data files will be placed in `/var/lib/mongodb` and log files in `/var/log/mongodb/mongod.log`.



You are now able to perform these operations:

Operation	Command
Start | stop | restart the server	**`sudo service mongod start|stop|restart`**
Get the server status	**sudo service mongod status**
Access MongoDB via the shell (covered later)	`mongo --host 127.0.0.1:27017`

Here you can see the server started along with its status.



MongoDB installation scripts now automatically bind MongoDB to localhost (IP address 127.0.0.1) for security reasons.

Installing on RedHat, Fedora or CentOS Linux

The Red Hat, Fedora and CentOS have a relationship similar to that of Debian and Ubuntu. Red Hat is the original company behind this distribution, producing its first release in 1995. In addition to making improvements in the graphical interface and overall management of Linux, Red Hat is known for its *rpm* (Red Hat Package Management) technology. In this corner of the Linux world, packages are bundled into files with the extension *.rpm, and contain installation instructions which make installation, updating and management of Linux software much easier.

Fedora is a free open source version of what is now *RHEL* (Red Hat Enterprise Linux). Fedora and the Fedora Project are sponsored by Red Hat, and serve as a test bed for innovation, which, when stable, is ported to RHEL. Fedora Linux releases tend to have rapid development cycles and short lifespans. CentOS is also affiliated with Red Hat, and is allowed direct use of RHEL source code. The main difference is that CentOS is free, but support is only available via the community (which is to say you are on your own!). For the purposes of this book we will use CentOS version 7.

Package Installation

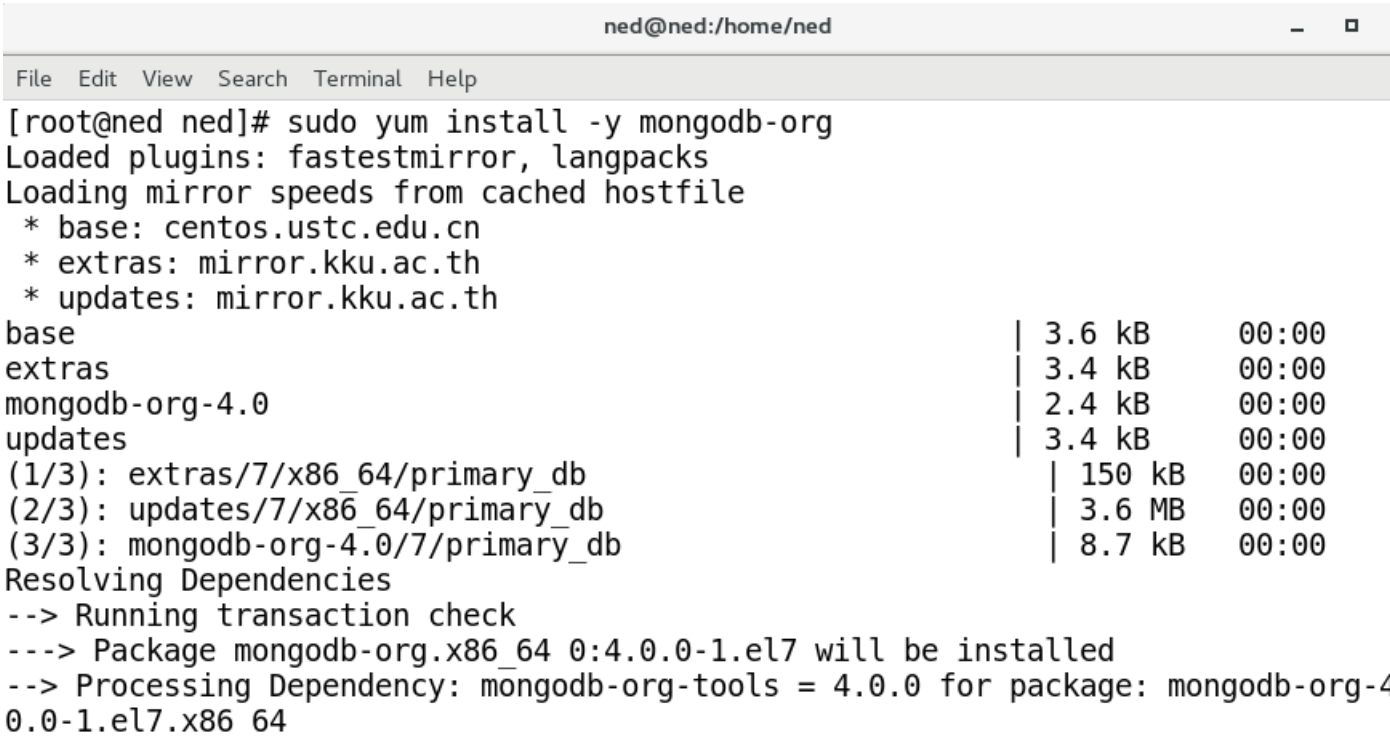
The MongoDB packages available for RHEL/Fedora/CentOS are exactly the same as those described above for Debian/Ubuntu. Also, as described earlier, a composite package mongodb-org which contains all four packages is available. Because RHEL/Fedora/CentOS packages use *rpm* for packaging, the tool of choice for installation, updating and management of packages is *yum* (Yellowdog Updater, Modified).

To install MongoDB on RHEL/Fedora/CentOS Linux distributions, proceed as follows:

1. Create a repository file for *yum* in the `/etc/yum.repos.d` directory. The filename should be like this:

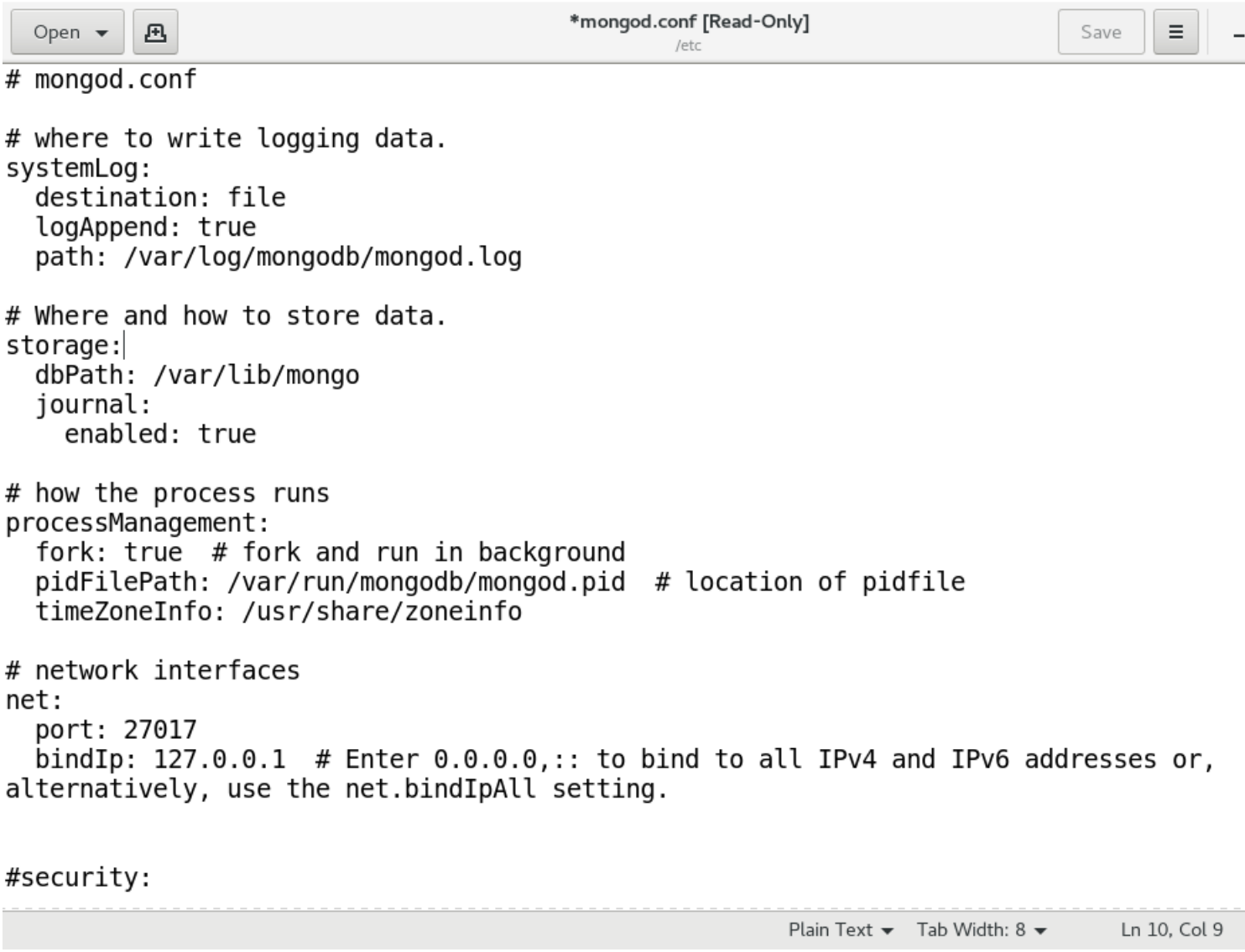
`mongodb-org-X.Y.repo` where **X** is the major version number for MongoDB, and **Y** is the minor release. As an example, for MongoDB version 4.0, the current version as of this writing, the filename would be:
`/etc/yum.repos.d/mongodb-org-4.0.repo`

2. Install the composite package using: sudo yum install -y mongodb-org.



Configure and Run MongoDB on RHEL/Fedora/CentOS

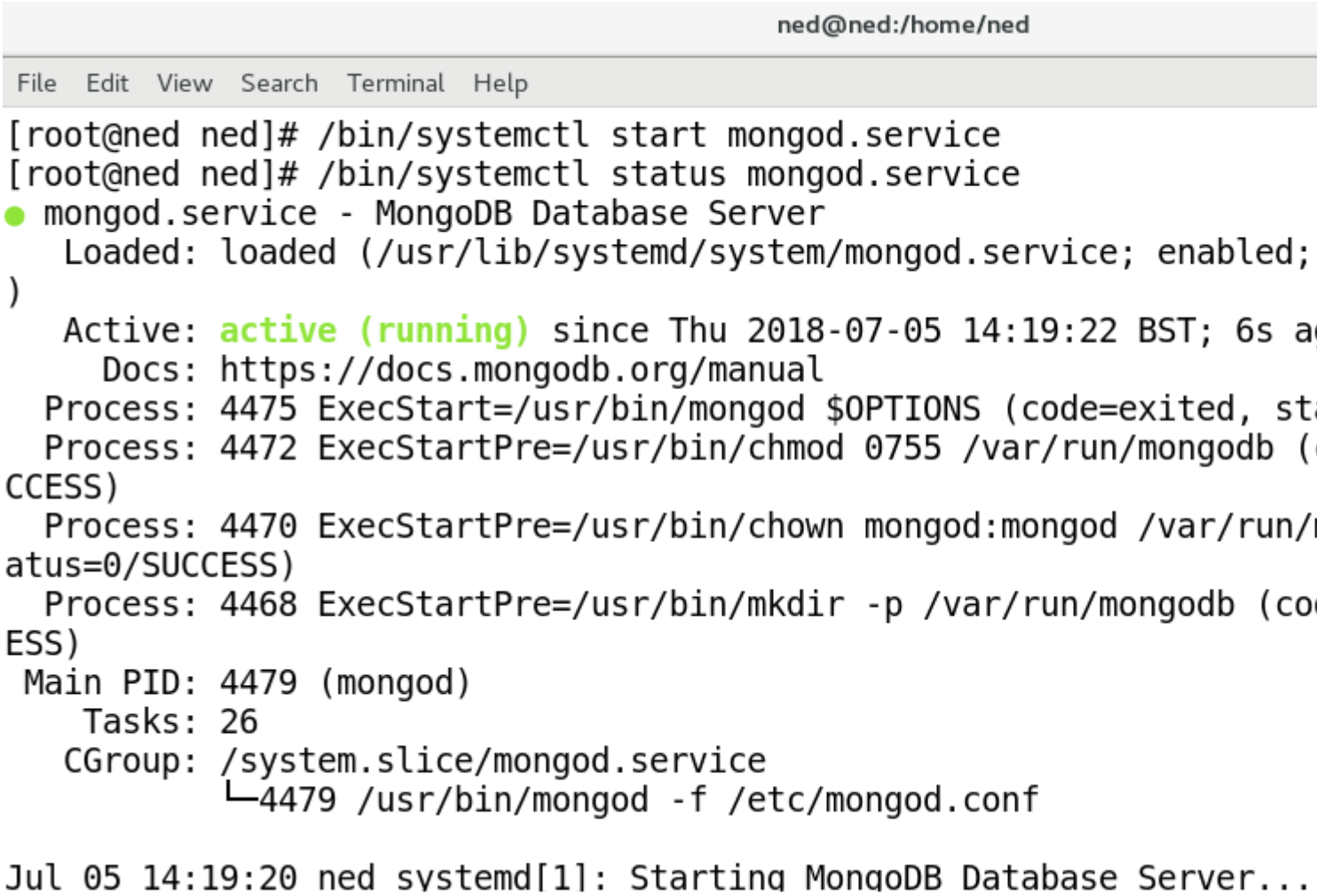
If you followed the procedure outlined above, a configuration file `/etc/mongod.conf` was auto-generated by the installation script. By default, database files will be placed in `/var/lib/mongodb` and log files in `/var/log/mongodb/mongod.log`. Here is an example of the auto-generated file for MongoDB v4.0 on CentOS 7:



You are now able to perform these operations:

Operation	Command
Start | stop | restart the server	`/bin/systemctl start|stop|restart mongod.service`
Access MongoDB via the shell (covered later)	`mongo --host 127.0.0.1:27017`

After starting the service, use the command `/bin/systemctl status mongod.service` to confirm the status of MongoDB:



Installing from Source

The beauty of installing MongoDB directly from its source code is that it ensures that you can run MongoDB on *any* server, and that you have the absolute latest version. Minimum requirements for source installation include:

- A *modern* C++ 11 compiler
- Python 2.7 or above
- pip (tool for installing python packages)
- git (recommended)

In addition, there are OS-specific requirements which are detailed in this table:

Linux	Compiler: GCC 4.8.2 or later
Red Hat, etc.	Libraries Needed: glibc-devel libcurl-devel openssl-devel epel-release

	python-devel
Ubuntu, etc.	Libraries Needed: build-essential libffi-dev libssl-dev python-dev
Mac OSX	Compiler: Clang 3.4 of XCode 5
	Libraries: XCode (especially command line tools)
Windows	Compiler: Visual Studio 2013 Update 4 or later

It is highly recommended you carefully read through the source installation process documentation, which can be found on github.com at this URL: https://github.com/mongodb/mongo/wiki/Build-Mongodb-From-Source.

The source build process does not follow the traditional sequence of configure, make and make install. Installation is performed using SCons (Software Construction Tool), which, in turn, uses the programming language Python. Accordingly, after you clone or download the MongoDB source, you will notice many Python scripts and configuration files.

For the purposes of this illustration we use CentOS 7. To install MongoDB from source, assuming all prerequisites listed above are met, proceed as follows:

1. Download the source code from github.com. There are two ways to download the MongoDB source code from github.com:
    1. Download directly from this URL:
    https://github.com/mongodb/mongo/archive/master.zip
    You would then need to unzip it into a folder such as `/home/user/mongo`.
    2. If you have installed *git*, you can clone the repository from a command line terminal as follows:

```
[root@ned ned]# git clone https://github.com/mongodb/mongo.git
Cloning into 'mongo'...
remote: Counting objects: 622827, done.
remote: Compressing objects: 100% (466/466), done.
remote: Total 622827 (delta 254), reused 245 (delta 134), pack-reused 622225
Receiving objects: 100% (622827/622827), 398.97 MiB | 10.54 MiB/s, done.
Resolving deltas: 100% (460128/460128), done.
Checking out files: 100% (19634/19634), done.
```

2. Change to the newly created (or cloned) **mongo** directory

3. Install pip requirements:

```
[root@ned mongo]# python -mpip install --user -r buildscripts/requirements.txt
Ignoring mypy: markers u'python_version > "3"' don't match your environment
Ignoring pypiwin32: markers u'sys_platform == "win32" and python_version < "3"' don't match your
Ignoring pypiwin32: markers u'sys_platform == "win32" and python_version > "3"' don't match your
Collecting cryptography==1.7.2 (from -r buildscripts/requirements.txt (line 2))
  Downloading https://files.pythonhosted.org/packages/99/df/71c7260003f5c469cec3db4c547115df39e9
0e78fd8a/cryptography-1.7.2.tar.gz (420kB)
    100% |████████████████████████████████| 430kB 1.7MB/s
Collecting jira==1.0.10 (from -r buildscripts/requirements.txt (line 3))
  Downloading https://files.pythonhosted.org/packages/5e/6e/96a299deee19be84d1f6317f71dd86e736b4
```

   On Windows you would need to run this command:
   `pip.exe install -r buildscripts\requirements.txt`
4. Build the source code using SCons:

```
[root@ned mongo]# buildscripts/scons.py all
scons: Reading SConscript files ...
scons version: 2.5.0
python version: 2 7 5 'final' 0
Checking whether the C compiler works... yes
Checking whether the C++ compiler works... yes
Checking that the C++ compiler can link a C++ program... yes
Checking if C++ compiler "g++" is GCC... yes
Checking if C compiler "gcc" is GCC... yes
Detected a x86_64 processor
Checking if target OS linux is supported by the toolchain... yes
```

At this point you can then follow the same steps listed above to run MongoDB:

- Create a directory for the database (e.g. `/var/lib/mongo`)
- Create a directory for the log (e.g. `/var/log/mongo`)
- Create a config file which indicates the locations of the database and log (e.g. `/etc/mongod.conf`)
- Start MongoDB