**Step 1: Download the Tarball**

$ curl -O http://downloads.mongodb.org/linux/mongodb-linux-x86\_64-amazon-3.2.11.tgz

**Step 2: Extract the files**

Replace the name of the file that you downloaded in the command below (if you downloaded another version).

$ tar -zxvf mongodb-linux-x86\_64-amazon-3.2.11.tgz

**Step 3: Change the folder name**

A long folder name is inconvenient. Let's rename it...

$ sudo mv mongodb-linux-x86\_64-amazon-3.2.11/ mongodb

**Step 4: Add mongod to Path**

To ensure that you are able to access mongod from your shell, you must add the following to your ~./bashrc file.

sudo vi ~/.bashrc

Add the following at the end of the file and save it.

export PATH=/home/ec2-user/mongodb/bin:$PATH

Reload the rc file by typing:

$ source ~/.bashrc

Test if all is well by typing mongo. If you get an output like the following, you are good. The connect simply failed because the mongod daemon is still not running.:

MongoDB shell version: 3.2.11

connecting to: test

2017-01-10T11:48:59.610+0000 W NETWORK [thread1] Failed to connect to 127.0.0.1:27017, reason: errno:111 Connection refused

2017-01-10T11:48:59.610+0000 E QUERY [thread1] Error: couldn't connect to server 127.0.0.1:27017, connection attempt failed :

connect@src/mongo/shell/mongo.js:229:14

@(connect):1:6

exception: connect failed

**Step 5: Specify the Database & Log Path**

Create a path for your data to reside. It is recommended that you use WiredTiger storage engine. [Starting with MongoDB 3.2](https://docs.mongodb.com/v3.2/core/wiredtiger/), the WiredTiger storage engine is the default engine in 64-bit builds. It provides document-level concurrency control for write operations. In simple words, it allows multiple clients to modify different documents of a collection at the same time. [Read More](https://docs.mongodb.com/v3.2/core/wiredtiger/).

$ mkdir -p <data-path>

$ mongod --storageEngine wiredTiger --dbpath <data-path> --fork --logpath </var/log/mongod.log>

You can also use --replSet switch to start a [default replica set](https://docs.mongodb.com/manual/replication/).

$ mongod --storageEngine wiredTiger --dbpath <data-path> --fork --logpath </var/log/mongod.log> --replSet <replicaset-name>

**Step 6: Create the Init Script**

I tweaked [this script](https://github.com/syokenz/mongodb-init-script) for Amazon Linux. (Many thanks [Shoken](https://github.com/syokenz)!)

**6.1: Create initialization script for mongod**

Create the script file by using the command below.

$ sudo vi /etc/init.d/mongod

Now, copy the following code and use a text editor to modify the DBPATH and OPT variables as appropriate:

#!/bin/sh

# chkconfig: 35 85 15

# description: Mongo is a scalable, document-oriented database.

# processname: mongod

# config: /etc/mongod.conf

# pidfile: /var/run/mongo/mongo.pid

. /etc/rc.d/init.d/functions

MONGOHOME="/home/ec2-user/mongodb"

CONFIGFILE="/etc/mongod.conf"

DBPATH="/home/ec2-user/<data-path>"

COMMAND="$MONGOHOME/bin/mongod"

OPT="--config $CONFIGFILE --replSet <replicaset-name>"

mongod=${MONGOD-$COMMAND}

usage() {

echo "Usage: $0 {start|stop|restart|status}"

exit 0

}

if [ $# != 1 ]; then

usage

fi

start()

{

echo -n $"Starting mongod: "

daemon $COMMAND $OPT

RETVAL=$?

echo

[ $RETVAL -eq 0 ] && sudo touch /var/lock/subsys/mongod

}

stop()

{

echo -n $"Stopping mongod: "

killproc -p "$DBPATH"/mongod.lock -d 300 "$COMMAND"

RETVAL=$?

echo

[ $RETVAL -eq 0 ] && sudo rm -f /var/lock/subsys/mongod

}

case "$1" in

start)

start

;;

stop)

stop

;;

restart)

stop

start

;;

status)

status $mongod

RETVAL=$?

;;

\* )

usage

;;

esac

**6.2: Execute Permission**

Give execute permission to the script.

$ chmod +x /etc/init.d/mongod

**6.3: Create Config file**

Now, create the config file using the following command:

$ sudo vi /etc/mongod.conf

Paste the following code in the configuration file (change the data-path, log-path and log-file.

# mongo.conf

dbpath = /home/ec2-user/<data-path>

#port = 27017

#

#where to log

logpath = /home/ec2-user/<log-path>/<log-file>.log

logappend = true

#rest = true

verbose = true

## for log , more verbose

##vvvvv = true

#

##profile = 2

##slowms = 10

# fork and run in background

fork = true

# Disables write-ahead journaling

# nojournal = true

# Enables periodic logging of CPU utilization and I/O wait

#cpu = true

# Turn on/off security. Off is currently the default

#noauth = true

#auth = true

# Verbose logging output.

#verbose = true

# Inspect all client data for validity on receipt (useful for

# developing drivers)

#objcheck = true

# Enable db quota management

#quota = true

# Set oplogging level where n is

# 0=off (default)

# 1=W

# 2=R

# 3=both

# 7=W+some reads

#oplog = 0

# Ignore query hints

#nohints = true

# Disable the HTTP interface (Defaults to localhost:27018).

#nohttpinterface = true

# Turns off server-side scripting. This will result in greatly limited

# functionality

#noscripting = true

# Turns off table scans. Any query that would do a table scan fails.

#notablescan = true

# Disable data file preallocation.

#noprealloc = true

# Specify .ns file size for new databases.

# nssize = <size>

# Accout token for Mongo monitoring server.

#mms-token = <token>

# Server name for Mongo monitoring server.

#mms-name = <server-name>

# Ping interval for Mongo monitoring server.

#mms-interval = <seconds>

# Replication Options

# in replicated mongo databases, specify here whether this is a slave or master

#slave = true

#source = master.example.com

# Slave only: specify a single database to replicate

#only = master.example.com

# or

#master = true

#source = slave.example.com

**6.4 Check Config**

Chkconfig command is used to setup, view, or change services that are configured to start automatically during the system startup. Let's add information for mongod now so that the service starts when your server reboots.

$ chkconfig --add mongod

**6.5 Test commands**

All the hard work is over! You can now use service commands to control mongod:

$ service mongod {start|stop|restart|status}

Start the service using:

$ service mongod start

Connect using the Mongo shell by simply typing mongo. Check your version by typing:

> version()