1. **To make mongodb listen to another port instead of using by default port:**

Now, you will read solutions like the following:

**mongo --port 22222**Where, 22222 is the desired port number.

After that, do the following:

**sudo service mongod stop**

Or,

**sudo systemctl stop mongod**

Now, this solution won’t work. It will give the following error:

**MongoDB shell version v3.4.3**

**connecting to: mongodb://127.0.0.1:22222/**

**2017-04-15T12:37:46.947+0530 W NETWORK [thread1] Failed to connect to 127.0.0.1:22222, in(checking socket for error after poll), reason: Connection refused**

**2017-04-15T12:37:46.948+0530 E QUERY [thread1] Error: couldn't connect to server 127.0.0.1:22222, connection attempt failed :**

**[connect@src/mongo/shell/mongo.js:237:13](mailto:connect@src/mongo/shell/mongo.js:237:13)**

Initial guess was port 22222 was not free. So, it is checked using

**netstat -nlp|grep 22222**

It was free.

One of the possible solutions (This one is correct. However, there could be other solutions).

**sudo systemctl stop mongod (centos 7)**

**Or,**

**sudo service mongod stop(centos 6)**

After that, edit the file **/etc/mongod.conf** using any editor

You will find lines like the following:

**# network interfaces**

**net:**

**port: 27017**

**bindIp: 127.0.0.1 # Listen to local interface only, comment to listen on all interfaces.**

Change the port number here.

**To any port number you like.**

After that start the mongod service again.

**sudo systemctl start mongod**

Or,

**sudo service mongod start**

It will work.

1. **Delete Mongodb**

First stop the mongod service if it is running

**i)sudo service mongod stop**

Remove any MongoDB packages that you had previously installed.

**ii)sudo yum erase $(rpm -qa | grep mongodb-org)**

Remove MongoDB databases and log files.

**iii)sudo rm -r /var/log/mongodb**

**sudo rm -r /var/lib/mongo**

1. **Make MongoDB listen to remote connection request:**

According to the default settings, mongodb will only listen to connections which are coming from 127.0.0.1 or localhost. So, until we enable mongodb to listen from remote connection request,

This is a trial function explaining that unless mongodb is configured to listen to remote connection request, remote connection requests will cause error:

**package trialdbauthentication;**

**import com.mongodb.DB;**

**import com.mongodb.MongoClient;**

**import java.net.UnknownHostException;**

**public class TrialDBAuthentication**

**{**

**/\*\***

**\* @param args the command line arguments**

**\*/**

**public static void main(String[] args)**

**{**

**MongoClient mongoClientInstance = null;**

**try**

**{**

**mongoClientInstance = new MongoClient("192.168.226.130", 27017);**

**DB databaseInstance = mongoClientInstance.getDB("serverApplicationDB");**

**boolean auth = databaseInstance.authenticate("dbadmin", "dbadmin123".toCharArray());**

**if (auth == false)**

**{**

**System.out.println("The database authentication is not successful");**

**}**

**else**

**{**

**System.out.println("The database authentication is successful");**

**}**

**}**

**catch (UnknownHostException ex)**

**{**

**System.out.println("The mongoClientInstance cannot be created successfully");**

**}**

**}**

**}**

This wont work. Even if the program is actually running in 192.168.226.130.

Unless, we change this line:

**mongoClientInstance = new MongoClient("192.168.226.130", 27017);**

To,

**mongoClientInstance = new MongoClient("127.0.0.1", 27017);**

Now, how to enable mongodb to listen to remote connection requests:

Open /etc/mongod.conf using vim editor.

**# Listen to local interface only. Comment out to listen on all interfaces.**

**bind\_ip=127.0.0.1**

Comment out bind\_ip line.

To, make it look like this:

**# Listen to local interface only. Comment out to listen on all interfaces.**

**#bind\_ip=127.0.0.1**