

Securing MongoDB:

1. Start MongoDB without access control
2. Connect to the instance, you can use Robo 3T GUI or mongo shell
3. Connect to localhost instance and use below command to create user with userAdminAnyDatabase role

```
use admin
db.createUser(
{
  user: "myUserAdmin",
  pwd: "abc123",
  roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]
}
```

4. Re-start MongoDB instance with security.authorization enabled in config file^[d] or can use below command line option

```
mongod --auth --port 27017 --dbpath /data/db1
```

5. To connect as admin user from command line:

```
mongo --port 27017 -u "myUserAdmin" -p "abc123" --authenticationDatabase "admin"
```

Code Changes in Node js

You can provide user name and password in mongoose connection URI ^[e].

```
mongoose.connect('mongodb://username:password@localhost:port/database');
```

Reference:

- a. Enabling Authentication in MongoDB <https://docs.mongodb.com/manual/tutorial/enable-authentication/>
- b. About how to provide connections using Mongoose.js <http://mongoosejs.com/docs/connections.html>
- c. Creating User in MongoDB: <https://docs.mongodb.com/manual/core/security-users/#user-authentication-database>
- d. Security Authorization in config file: <https://docs.mongodb.com/manual/reference/configuration-options/#security.authorization>
- e. More details regarding URI options here <https://docs.mongodb.com/manual/reference/connection-string/>