Testing running MongoDB

1. Create new DB dir

$ mkdir d:\data\db

1. Run DB process (service side)

$ "C:\Program Files\MongoDB\Server\3.4\bin\mongod.exe" --dbpath d:\data\db

1. Connect to DB (client side)

$ "C:\Program Files\MongoDB\Server\3.4\bin\mongo.exe"

Config windows services

Create database dir and log

$ mkdir c:\data\db

$ mkdir c:\data\log

Install windows service

$ "C:\Program Files\MongoDB\Server\3.4\bin\mongod.exe" --config "C:\Program Files\MongoDB\Server\3.4\mongod.cfg" --install --serviceName "MongoDB"

Or

sc.exe create MongoDB binPath= "\"C:\Program Files\MongoDB\Server\3.4\bin\mongod.exe\" --service --config=\"C:\Program Files\MongoDB\Server\3.4\mongod.cfg\"" DisplayName= "MongoDB" start= "auto"

Start windows service

$ net start MongoDB

Stop windows service

net stop MongoDB

Delete windows service

$ "C:\Program Files\MongoDB\Server\3.4\bin\mongod.exe" --remove

Auth

$ mongo

Add an account to admin DB

use admin

db.createUser(

{

user: "myUserAdmin",

pwd: "abc123",

roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]

}

)

Exit shell

exit

Restart DB process(Service side)

If the windows service is not started, we can use mongod instruction

mongod --auth --port 27017 --dbpath c:/data/db

If we can use windows service, we can simplify the above instruction, we need to first configure the config file in mongodb , that is C:\Program Files\MongoDB\Server\4.0\mongod.cfg

systemLog:

destination: file

path: c:\data\log\mongod.log log

Append: true

storage:

dbPath: c:\data\db

journal:

enabled: true

net:

bindIp: 127.0.0.1

port: 27017

security:

authorization: enabled

Start or stop windows service

$ net stop MongoDB

$ net start MongoDB

Use user admin account to log in

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

You haven't configure the security features in Mongodb like authorization and authentication. You can ignore this if you are going to learn Mongodb. But when the product is going to production level. you should concern them. You can enable access control by using mongod --auth.

For example you can run mongod --auth --port 27017 --dbpath /data/db1. After that, you can secure your database with username and password.

You can add user in database using following command:

use admin

db.auth("myUserAdmin", "abc123" )

After that you can use

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

to connect to the database.

You can add bind\_ip in mongod.conf as follows,

bind\_ip = 127.0.0.1,192.168.161.100

You can define many if you need. This bind\_ip option tells MongoDB to accept connections from which local network interfaces, not which “remote IP address”. And run mongod --config <file path to your mongod.conf> Altogether you can run mongod --auth --port 27017 --dbpath /data/db1 --config <file path to your mongod.conf>

https://docs.mongodb.com/manual/reference/method/db.auth/

db.createUser(

... {

... user: "lukeli",

... pwd: "555",

... roles: [ {role:"userAdminAnyDatabase", db:"admin"}]

... }

... )

2018-08-31T21:10:20.380-0700 E QUERY [js] Error: couldn't add user: not authorized on admin to execute command { createUser: "lukeli", pwd: "xxx", roles: [ { role: "userAdminAnyDatabase", db: "admin" } ], digestPassword: true, writeConcern: { w: "majority", wtimeout: 600000.0 }, $db: "admin" } :

\_getErrorWithCode@src/mongo/shell/utils.js:25:13

[DB.prototype.createUser@src/mongo/shell/db.js:1491:15](mailto:DB.prototype.createUser@src/mongo/shell/db.js:1491:15)

https://www.jianshu.com/p/c3fa4be0679b

@(shell):1:1