Chapter 2: Authorization and Encryption

Homework 2.2 : Create application specific users

Back to the Question

If you were able to complete the first homework exercise for this chapter successfully then you should have no problem with this exercise. The goal of this exercise is to make sure you understand the difference between the database that a role applies to and the database that a user must authenticate to.

Like last time go ahead and set up your directory structure and generate your keyfile:

COPY

```
$ mkdir -p ~/M310-HW-2.2/{r0,r1,r2}
$ openssl rand -base64 755 > ~/M310-HW-2.2/keyfile
$ chmod 400 ~/M310-HW-2.2/keyfile
```

We can then start each member in our replica set:

COPY

```
$ mongod --dbpath ~/M310-HW-2.2/r0 --logpath ~/M310-HW-
2.2/r0/mongo.log --port 31220 \
         --fork --auth --keyFile ~/M310-HW-2.2/keyfile --
replSet HW-2.2
$ mongod --dbpath ~/M310-HW-2.2/r1 --logpath ~/M310-HW-
2.2/r1/mongo.log --port 31221 \
         --fork --auth --keyFile ~/M310-HW-2.2/keyfile --
replSet HW-2.2
$ mongod --dbpath ~/M310-HW-2.2/r2 --logpath ~/M310-HW-
2.2/r2/mongo.log --port 31222 \
```

```
        --fork --auth --keyFile ~/M310-HW-2.2/keyfile --
replSet HW-2.2
```

From here we can connect to the **mongod** running on port **32220**, initiate the replica set, and begin creating our users:

COPY

```
$ mongo --port 31220
use admin
rs.initiate()
db.createUser({user: "admin", pwd: "webscale", roles: [{
role:"root", db: "admin" }]})
db.auth('admin', 'webscale')
db.createUser({user: "reader", pwd: "books", roles: [{
role:"read", db: "acme" }]})
db.createUser({user: "writer", pwd: "typewriter", roles: [{
role:"readWrite", db: "acme" }]})
```

After you've created the **admin** user you can authenticate against them and add the other members.

COPY

```
$ mongo --port 31220
use admin
db.auth('admin', 'webscale')
rs.add('database.m310.mongodb.university:31221')
rs.add('database.m310.mongodb.university:31222')
```

Proceed to next section