Chapter 2: Authorization and Encryption

Homework 2.3 : Create custom role

Back to the Question

This is a great exercise to bring together all of the core concepts that define the role-based access control model that MongoDB uses for authorization. In order to be successful on this homework you need to understand actions, resources, privileges, and be familiar with the built-in roles.

Here's the break down of each one of these privileges:

role name	db	collection	action	inherited role
HRDEPARTMENT	HR		find	
HRDEPARTMENT	HR	employees	insert	
HRDEPARTMENT	HR		dropUser	
MANAGEMENT	HR			dbOwner
EMPLOYEEPORTAL	HR	employees	find	
EMPLOYEEPORTAL	HR	employees	update	

Now that we know the important parts of our custom roles we can go ahead and start up our standalone server.

COPY

```
$ mkdir -p ~/M310-HW-2.3
$ mongod --dbpath ~/M310-HW-2.3 --logpath ~/M310-HW-2.3/mongo.log
--port 31230 --fork
```

We can now transform our role data into the appropriate role documents and execute them on our server.

COPY

```
$ mongo --port 31230
use admin
db.createRole({
  role: "HRDEPARTMENT",
  privileges: [
    {
      resource: { db: "HR", collection: "" },
      actions: [ "find", "dropUser" ]
    }, {
      resource: { db: "HR", collection: "employees" },
      actions: [ "insert" ]
    }
  ],
  roles:[]
})

db.createRole({
  role: "MANAGEMENT",
  privileges: [],
  roles:[{
    role: "dbOwner", db: "HR"
  }]
})

db.createRole({
  role: "EMPLOYEEPORTAL",
  privileges: [{
    resource: { db: "HR", collection: "employees" },
    actions: [ "find", "update" ]
  }],
  roles:[]
})
```

Proceed to next section