Deploying the Management Console

> ⊘ A more current version of this guide is now available. If you are deploying the Management Console for development, see this page: https://githu b.com/duracloud/deployment-docs/blob/master/dev-guide-getting-started.md. If you are deploying the Management Console in production, see this page: https://github.com/duracloud/deployment-docs/blob/master/management-console-setup.md. The following document should be considered deprecated and will be removed in the near future.

This article describes the necessary steps for configuring and running the DuraCloud Management Console. The Management Console is a web application that enables users to manage DuraCloud account and user information as well as global configuration settings for a DuraCloud installation. Mor e information about the capabilities of the DuraCloud Management Console is available here.

If you are not yet familiar with the DuraCloud Management Console please refer to the DuraCloud Architecture document, which describes how the Management Console fits into the DuraCloud ecosystem.

Requirements

1. Java: Version 8 required
    a. The Oracle JDK is recommended for building DuraCloud, as this is the JDK used for DuraCloud testing.
2. Maven 3.x
3. Tomcat 8.x
4. Git
5. MySQL database
6. AWS account and credentials (allowing use of SES and SNS for notifications)

Download, Build, Configure

1. Check out latest stable release from the GitHub repository. Determine the release number of the latest stable release by going to https://github. com/duracloud/management-console/releases and making a note of the tag name of the most current release.

```
git clone https://github.com/duracloud/management-console.git
cd management-console
git checkout <latest-release-tag>
```

2. Configure MySQL
    a. Create an empty MySQL database for the management console, along with a user that has read and write permissions to the database.
    b. Create the database schema using management-console/resources/schema.sql (found in the management-console code baseline.)
3. Create a configuration file.
    a. Set up a configuration file that can be used by the various components of the system. A template of this configuration file can be found in the code at account-management-app/src/main/webapp/WEB-INF/config/mc-config.properties
        i. Copy the example mc-config.properties file to a permanent location
        ii. Edit the file to configure the connections to the management console database and AWS (for notifications):

```
###################
# MC DATABASE
###################
# Config for the management console database - used to retrieve accounts and storage
provider credentials
db.host=[fill in]
db.port=[fill in]
db.name=[fill in]
# User must have read permission
db.user=[fill in]
db.pass=[fill in]

mc.host=<hostname>
mc.port=443
mc.context=<e.g. /ama> # leave blank if the mc is deployed in the root context.
notification.user=<aws access key id>
notification.pass=<aws secret key>
notification.from-address=<email address>
notification.admin-address=<admin email address>
```

    b. Add a system property (mc.config.file) to point to your management console configuration file.

i. This can be done by adding the property directly to your JAVA_OPTS or CATALINA_OPTS environment variables. You can also set these values in $CATALINA_HOME/bin/setenv.sh (linux,osx) or $CATALINA_HOME/bin/setenv.bat (windows), for example:

```
JAVA_OPTS="${JAVA_OPTS} -Dmc.config.file=file:c:/duracloud/mc-config.properties"
```

ii. The mc.config.file system property can also refer to an Amazon S3 address using the s3://<bucket>/<path to file> syntax provided your tomcat instance is running on an instance with the appropriate AWS credentials. More information on AWS credentials management.

4. Configure Tomcat and Maven to allow application deployment

    a. Follow the instructions here under "Configure Tomcat" and "Configure Maven"

5. On the command line, build the MC codebase

```
mvn clean install
```

6. Create an SNS topic

    a. Log into the AWS account associated with the Management Console application.
    b. In SNS add a topic.
    c. Ensure that the AWS credentials associated with the Management Console has permission to publish to that topic.

7. In a web browser, go to https://<yourhost>/<context>
8. Click on "Create New User" under the login form.
9. Create a new user (this will be your root user).
10. Verify that you can log in with your new user. You will not have access to any accounts yet.
11. Edit management-console/resources/sql/make-user-root.sql by replacing the text 'your-root-username-here' with the username you just created.
12. Login with the user you created
13. Click on Root Console link in the upper right corner.
14. Click on Global Properties
15. Click Edit, enter the ARN of the topic you created above as well as the CloudFront settings and click save.
    a. NB: the CloudFront key path can refer to an s3 url: ie s3://<bucket>/path/to/item.
    b. Also be sure that the AWS credentials have access to the specified bucket.
16. Click DuraCloud Mill tab and enter the appropriate settings.
17. Create an account via the Accounts tab.