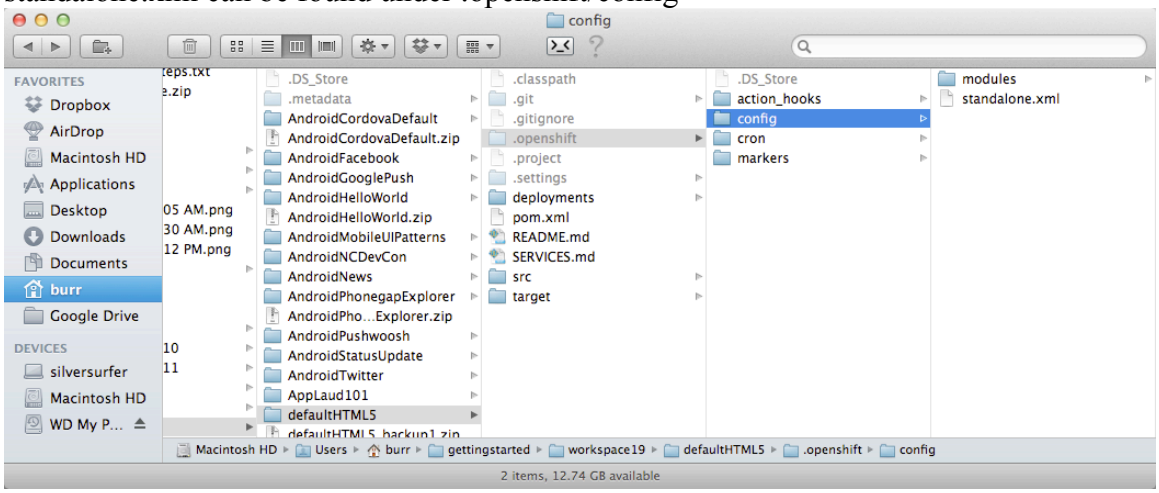
Setting up Basic Auth on a JBoss on Openshift application

1) Turn on the security module in standalone.xml
standalone.xml can be found under .openshift/config



https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html-single/Administration_and_Configuration_Guide/index.html#chap-Securing_JBoss_Enterprise_Application_Platform

Sample Code:
https://github.com/burrsutter/cordovaendpoint

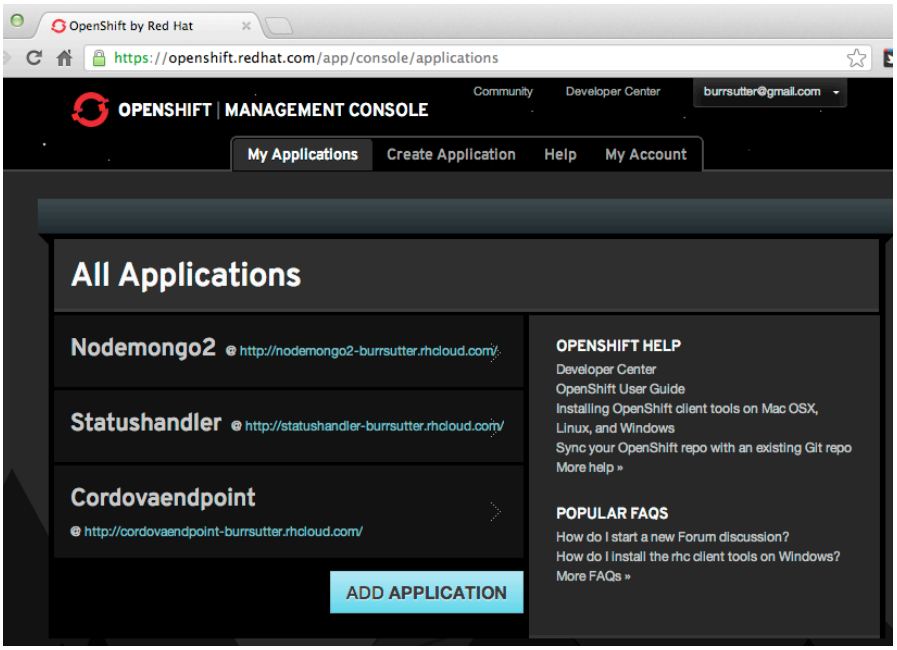
Add the login-module for the "ApplicationRealm"

```
<login-module code="UsersRoles" flag="required">
   <module-option name="usersProperties"
     value="${env.OPENSHIFT_DATA_DIR}/application-users.properties"/>
   <module-option name="rolesProperties"
     value="${env.OPENSHIFT_DATA_DIR}/application-roles.properties"/>
</login-module>
```

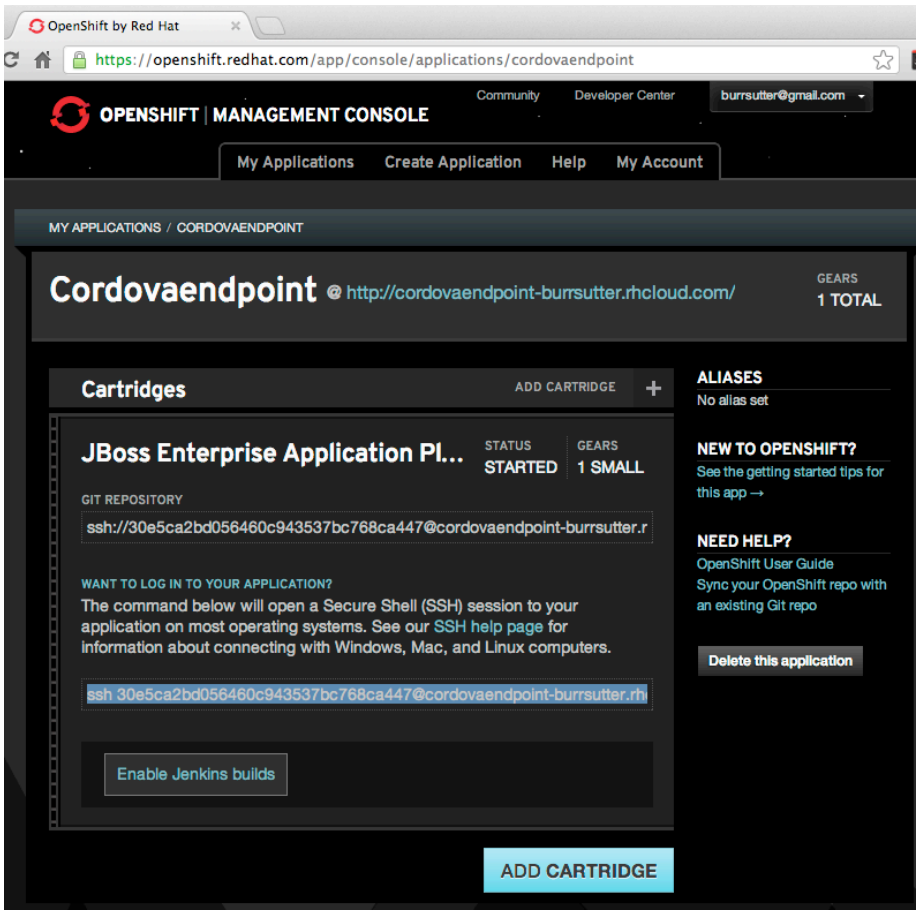
Note: OPENSHIFT_DATA_DIR is an env variable on the openshift host. It points to a persistent storage directory that will be there between restarts of the JBoss server and your cloud instance.

From the command line, use "git commit standalone.xml" and "git push" to make this change happen on your Openshift application.

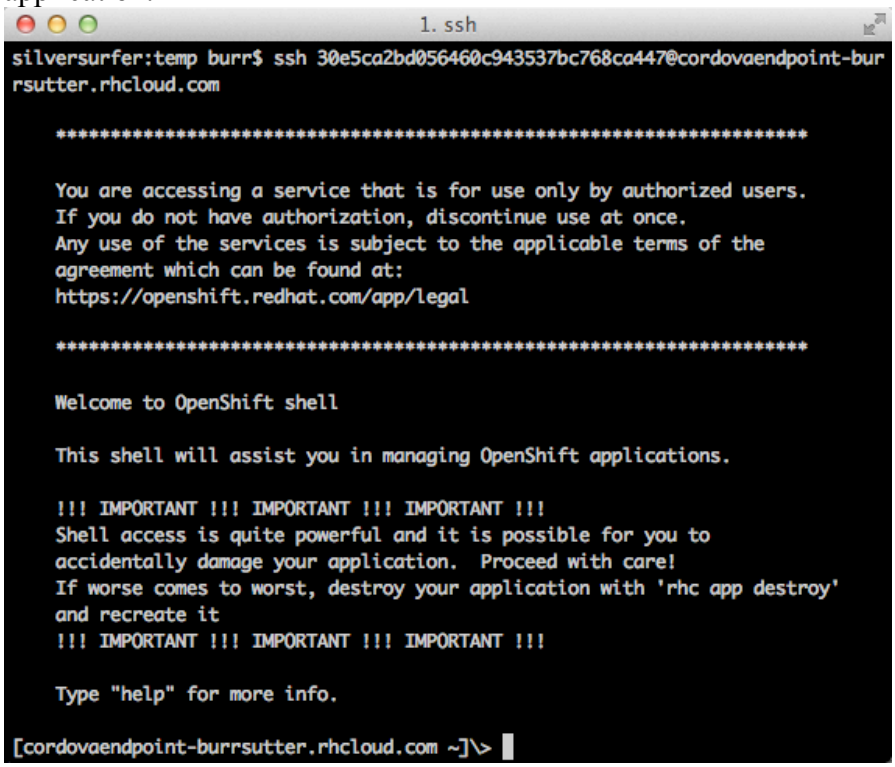
2) SSH into your application
2.1) First login to the Openshift Management Console - and select the application needing security.



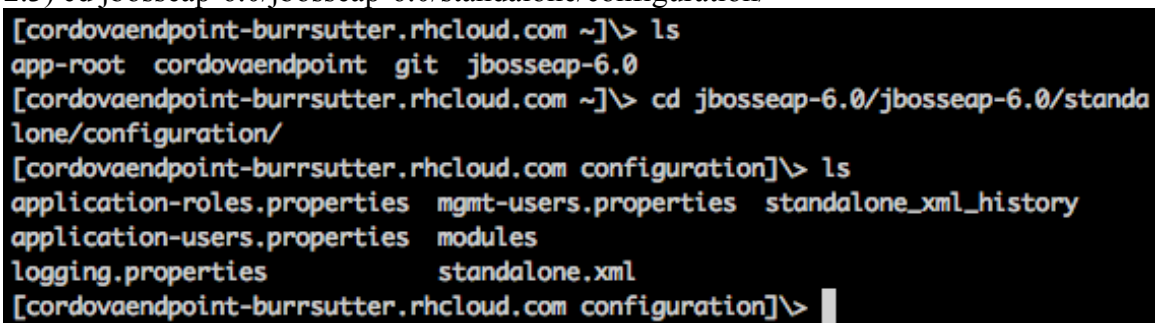
Select the link **"**Want to log in to your application?**"** - it will expand to provide you the SSH details.



2.2) Copy the ssh line and pasted it into your SSH client - on Linux and Mac, that is the normal terminal. On Windows, you will need to install & configure a SSH client application.



2.3) cd jbosseap-6.0/jbosseap-6.0/standalone/configuration/



2.4) cp application-roles.properties $OPENSHIFT_DATA_DIR
2.5) cp application-users.properties $OPENSHIFT_DATA_DIR
2.6) cd $OPENSHIFT_DATA_DIR
2.7) Edit the roles file for your needs
guest=guest
remove the # which comments this line out
2.8) Edit the users file to add your user
guest=password

3) Add a jboss-web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Configure usage of the security domain "other" -->
<jboss-web>
   <security-domain>other</security-domain>
   <disable-audit>true</disable-audit>
</jboss-web>
```

4) Add or update your web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd" version="3.0">
  <context-param>
    <param-name>resteasy.role.based.security</param-name>
    <param-value>true</param-value>
  </context-param>

  <security-constraint>
    <web-resource-collection>
      <web-resource-name>Resteasy</web-resource-name>
      <url-pattern>/rest/secured/*</url-pattern>
    </web-resource-collection>
    <auth-constraint>
      <role-name>admin</role-name>
      <role-name>guest</role-name>
    </auth-constraint>
  </security-constraint>

  <login-config>
    <auth-method>BASIC</auth-method>
    <realm-name>UsersRoles</realm-name>
  </login-config>

  <security-role>
    <role-name>admin</role-name>
  </security-role>
  <security-role>
    <role-name>guest</role-name>
  </security-role>
</web-app>
```

5) Add RolesAllowed to your REST endpoint

```
    @POST
    @Consumes("application/json")
    @RolesAllowed({"admin","guest"})
    public StatusUpdate create(StatusUpdate entity)
    {
       em.joinTransaction();
       entity.setCreate_time(new java.util.Date());

      entity.setVersion(1);
       em.persist(entity);
       return entity;
    }
```

6) Client-side JavaScript (Phonegap or mobile web)

```
var endpointURL =
"https://cordovaendpoint-
burrsutter.rhcloud.com/rest/secured/statusupdate";

function make_base_auth(user, password) {
        var tok = user + ':' + password;
        var hash = btoa(tok);
        return "Basic " + hash;
}

$.ajax({
    url: endpointURL,
    contentType: "application/json",
    dataType: "json",
    type: "POST",
    data: statusDataAsJSON,
    beforeSend: function (xhr){
       xhr.setRequestHeader('Authorization',
        make_base_auth("guest", "password"));
    },
    success: function(data) {
      refreshList();
     $("#newStatusForm")[0].reset();
    },
    error: function(error) {
      console.log(error);
    }
}); // ajax
```