Seaports machines

# Machines Description

## seaports-pacific-sql:

This machine is meant to be a MySQL server, containing the seaports database, so that the Seaports application can access data.

Requirements:

MySQL server installed, running and launching at start.

Port 3306 opened.

MySQL configuration:

MySQL databases: “seaports” & “seaports-test”

MySQL user “seaports-app” created with access to the “seaports” database

MySQL user “seaports-test” created with access to the “seaports-test” database.

## seaports-pacific-dev:

This machine is meant to be a Tomcat web server to host the seaports web application.

Requirements:

Tomcat server latest version installed, running and launching at start.

Seaports application deployed on Tomcat server.

Port 8080 opened.

Tomcat configuration:

Configured to access the MySQL database on the “seaports-pacific-dev” machine.

# Download Seaports packages:

Download or checkout the packaged files for setting up the Seaports app from: “http://……”

# Installation

## seaports-pacific-sql:

Make sure the port 3306 (MySQL default port) is opened under the cloud platform hosting it, such as NeCTAR (see “*Security and Groups*” in the NeCTAR dashboard).

Install MySQL server:

*$> sudo yum install mysql mysql-server*

Start MySQL server:

*$> sudo /etc/init.d/mysqld restart*

Enable MySQL service to run at system start:

*$> chkconfig mysqld on*

Configure MySQL server:

*$> sudo /usr/bin/mysql\_secure\_installation*

Go through the steps of the secure installation with the following selection: replace MySQL server root password, remove anonymous users, allow root login remotely, remove test DB, reload privilege tables.

Open port 3306 (to all since it is not production environment):

$> *sudo iptables -A INPUT -i eth0 -p tcp -m tcp --dport 3306 -j ACCEPT*

Connect to mysql console:

*$> sudo mysql -u root -p*

Create a new user called “**seaport-app**” with password (eg. “**P@ssw0rd**”)

*mysql> CREATE USER 'seaport-app'@'%' IDENTIFIED BY 'P@ssw0rd';*

Grant all privileges (again, because this is not production):

*mysql> GRANT ALL PRIVILEGES ON \*.\* TO 'seaport-app'@'%' WITH GRANT OPTION;*

Reload the privileges after modifications:

*mysql> FLUSH PRIVILEGES;*

*mysql> CREATE DATABASE seaports;*

*mysql> CREATE DATABASE seaports\_test;*

**FROM LOCAL TERMINAL, NOT VM**

Copy the SQL scripts to load the databases on the machine:

Copy the SQL files from the packaged files to the MySQL machine:

Example:

*local> scp ./seaports\_dump.sql username@XXX.XXX.XXX.XXX:/target/path*

*local> scp ./seaports\_test\_dump.sql \ username@XXX.XXX.XXX.XXX:/target/path*

**BACK ON THE VM**

In the mysql command line:

*mysql>USE seaports;*

*mysql>source /target/path/seaports\_dump.sql;*

*mysql>USE seaports\_test;*

*mysql>source /target/path/seaports\_test\_dump.sql;*

## seaports-pacific-dev:

Make sure the port 8080 (Tomcat default port) is opened under the cloud platform hosting it, such as NeCTAR (see “*Security and Groups*” in the NeCTAR dashboard).

Find latest Java version and install it (minimum 1.7.0):

*$> sudo yum search openjdk*

*$> sudo yum install java-1.7.0-openjdk*

Download and uncompress Tomcat 8.0.0 into */opt/*:

This version of Tomcat is available in the Tomcat website archives:

[*http://archive.apache.org/dist/tomcat/tomcat-8/v8.0.0-RC10/bin/*](http://archive.apache.org/dist/tomcat/tomcat-8/v8.0.0-RC10/bin/)

Example:

*$> cd /opt/*

*$> sudo wget http://archive.apache.org/dist/tomcat/tomcat-8/v8.0.0-RC10/bin/apache-tomcat-8.0.0-RC10.tar.gz*

*$> sudo tar -zxvf apache-tomcat-8.0.0-RC10.tar.gz*

*$> sudo rm apache-tomcat-8.0.0-RC10.tar.gz*

Startup Tomcat server:

Before starting Tomcat, create a file “*/bin/setenv.sh*” in order to set the startup option SKIP\_IDENTIFIER\_CHECK to *true*. This is the content of the *setenv.sh* file:

*#!/bin/sh*

*# Identifiers will not be checked to ensure that they conform to the Java Language Specification for Java identifiers.*

*CATALINA\_OPTS="-Dorg.apache.el.parser.SKIP\_IDENTIFIER\_CHECK=true"*

*$> cd /opt/apache-tomcat-8.0.0-RC10/bin*

*$> sudo sh startup.sh*

Check that the Tomcat server is up and running by connecting to it via a web browser (http://XXX.XXX.XXX.XXX:8080).

Create tomcat users by editing the file “*/conf/tomcat-users.xml*”. Add inside <tomcat-users> section:

<!-- user manager can access only manager section.-->

<role rolename="manager-gui" />

<user username="**manager**" password="**tomcat**" roles="manager-gui" />

<!-- user admin can access manager and admin section both.-->

<role rolename="admin-gui" />

<user username="**admin**" password="**tomcat**" roles="manager-gui,admin-gui" />

It is now possible to log in the Tomcat “manager” app (http://XXX.XXX.XXX.XXX:8080/manager/) using **admin/tomcat** credentials.

Deploy the web application on Tomcat server:

On the development machine, in Eclipse. select the Seaports project (CSS).

File -> Export…

Open “Web” option and select “WAR file”.

Browse to choose where the WAR file gets created. Name the file **ROOT.war** if you want it to be deployed under “/” on the server.

On the Tomcat server, increase the maximum upload size in order to upload the War file. Edit the file /opt/apache-tomcat-8.0.0-RC10/webapps/manager/WEB-INF/web.xml.

Increase the numbers at lines 55 and 56:

<multipart-config>

<!-- 50MB max -->

<max-file-size>52428800**0**</max-file-size>

<max-request-size>52428800**0**</max-request-size>

<file-size-threshold>0</file-size-threshold>

</multipart-config>

Go to http://XXX.XXX.XXX.XXX:8080/manager/ and upload the generated WAR file.

Check that the War file has been properly deployed: http://XXX.XXX.XXX.XXX:8080/

Set up the connection to the SQL database:

Edit the file **/opt/apache-tomcat-8.0.0-RC10/webapps/ROOT/WEB-INF/classes/db.properties** in order to specify the connection credentials to the SQL database.

db.password=**P@ssw0rd**

db.username=**seaport-app**

db.url=jdbc:mysql://**XXX.XXX.XXX.XXX:3306/seaports**

db.dialect=org.hibernate.dialect.MySQL5Dialect

db.driver=com.mysql.jdbc.Driver

Reload the web application by clicking on the “Reload” button in the Tomcat manager interface. Alternatively, stop the web application using the Tomcat manager interface (“Stop” button) and start it again (“Start” button).

Try logging in at http://XXX.XXX.XXX.XXX:8080/login using the credentials **user/password**.

If the login succeeds, the Climate Smart Seaports application is fully deployed.

Enable Tomcat service to run at system start:

**FROM LOCAL TERMINAL, NOT VM**

Copy the script called “Tomcat” from the packaged files to the Tomcat server machine, into the folder ‘/etc/init.d’:

*local> scp ./seaports\_dump.sql username@XXX.XXX.XXX.XXX:/target/path*

**BACK ON THE VM**

Edit this script to replace the ‘JAVA\_HOME’ and ‘TOMCAT\_BIN’ path as per your machine installation.

Set the ‘execute’ permissions onto this file:

*$> cd /etc/init.d*

*$> chmod +x Tomcat*

Enable this script to run at startup:

*$> chkconfig --add Tomcat*

*$> chkconfig --level 2345 Tomcat on*

Sources:

<http://www.java-allandsundry.com/2012/11/jsp-expression-language-and-class.html>

<http://tomcat.apache.org/tomcat-7.0-doc/config/systemprops.html>

<http://stackoverflow.com/questions/15900044/expression-language-skip-identifier-check-in-tomcat-7>