C2SIM/BML Version 4 Server Installation

Overview

This is a description how to build a C2SIM/BML server. It describes the process used for the public server available on the C2SIM sandbox supported by the George Mason University C4I Cyber Center.

This is obviously not the only way to do this but is one that works.

Required Software

Create VMWare Image for Centos version 7. It is also possible to use a native mode Linux server or a different Linux distribution. The following describes a Centos 7 implementation.

Links to public software required for operation and development

Centos 7

Apache Apollo 1.7.1

Apache Tomcat

Java Runtime

Java JDK (Required only for development)

Create Accounts

Root Account
    Password xxxxx

Create bmluser account
    Password: yyyyy

Set bmluser for automatic logon (Optional)

Copy bmldist.zip to bmluser home and unzip

Create ~/bin directory and make sure it is in the path for bmluser

Locate the following line in /etc/sudoers

```
root    ALL=(ALL)       ALL
```

Add the following to give bmluser full sudo rights:

```
bmluser ALL=(ALL)           ALL
```

Copy start-all and stop-all from the distribution folder to ~/bin and make them executable.

Copy bmlFiles from the distribution to /home/bmluser

Add the following to .bash_profile:

```
BML_HOME="/home/bmluser/bmlFiles/"
export BML_HOME
```

Install Java

Remove Open Java (Non Sun/Oracle)
sudo yum -y remove java*

Download latest jdk from Oracle

Install java
rpm –ivh jdk-8u65-linux-x64.rpm (Actual release may vary)

Java is installed in /usr/java with various soft links

Set JAVA_HOME in .bash_profile

```
JAVA_HOME="/usr/java/jdk1.8.0_65"
export JAVA_HOME
```

Install Tomcat

Download and install tomcat in /opt/tomcat
Download latest tomcat e.g. apache-tomcat-8.0.30.tar.gz
Create /opt/tomcat
sudo mkdir /opt/tomcat
cd /opt/tomcat
sudo chmod /opt/tomcat 755

Copy apache-tomcat-xxx.tar.gz to /opt/tomcat

Install by unzipping and un-tarring the .tag.gz file

Create symbolic links in ~/bin to start and stop tomcat

ln -s /opt/tomcat/apache-tomcat-8.0.30/bin/startup.sh tomcat_start
ln -s /opt/tomcat/apache-tomcat-8.0.30/bin/shutdown.sh tomcat_stop

Set $CATALINA_HOME in ~/.bash_profile

```
   CATALINA_HOME="/opt/tomcat/apache-tomcat-8.0.30"
export CATALINA_HOME
```

Set up tomcat admin user.
Adding the following to /conf/tomcat-users.xml

```
<xml version="1.0" encoding="UTF-8">
   <tomcat-users>
      <role rolename="tomcat"/>
      <user username="tomcat" password="tomcat" roles="manager-
         script,manager-gui"/>
   </tomcat-users>
```

Copy the BML Server code BMLServer.war to:

/opt/tomcat/apache-tomcat-8.0.30/webapps

Install Apache Apollo (Messaging Broker)

Download apache-apollo-1.7.1-unix-distro.tar.gz from:
http://activemq.apache.org/apollo/download.html

Create /opt/lib/apollo and unpack the distribution there.

Be sure that bmluser is the owner of /opt and /opt/lib/apollo and has RWX permissions.

Create /opt/apollo

Navigate to /opt/apollo and create a broker instance:

/opt/lib/apollo/apache-apollo-1.7.1/bin/apollo create /opt/apollo/bmlStomp

The main configuration file for this instance is apollo.xml.

Copy apollo.xml from the distribution folder to /opt/apollo/bmlStomp/etc

Test Installation

Start two server components

~/bin/start-all will start tomcat and apollo (STOMP).

Ports Used:
Tomcat 8080
Apollo 61613
The Linux firewall set up to pass these ports

To shutdown both services gracefully:

~/bin/stop-all

There are web based management applications for both services. These are on the Firefox toolbar.

Logins:
Tomcat: URL: http://localhost:8080/manager/html
User: tomcat Password: tomcat

Apache Apollo: URL: http://localhost:61680/console/index.html#
User: admin Password: password

Client Applications

Open a terminal window

cd to the bmlClient folder in bmldist

Start STOMP client to Receive publications

java –jar BML_STOMP localhost

Expected Response: CONNECTED

Open a second terminal window and submit a test transaction

java –jar BML_WS_All.jar localhost IBML09_GSR.xml userID (Use your initials)

Expected Response in second window:

<?xml version="1.0" encoding="UTF-8"?><result><status>OK</status><msgNumber>1</msgNumber><time>0.212</time></result>

Elapsed time = 2.248

Expected Response in First Window:

<?xml version="1.0" encoding="UTF-8"?><BMLReport xmlns:jc3iedm="urn:int:nato:standard:mip:jc3iedm:3.1a:oo:2.0" xmlns="http://netlab.gmu.edu/IBML" xmlns:msdl="http://netlab.gmu.edu/JBML/MSDL"> <Report> <CategoryOfReport>StatusReport</CategoryOfReport> <TypeOfReport>GeneralStatusReport</TypeOfReport> <StatusReport>

… Some lines omitted

</BMLReport>