OpenAM Server Setup

Table of Contents

Pre-Installation Procedure

JVM Tuning

Using the guide JVM Tuning document at the following link, set up Tomcat 8:

> https://backstage.forgerock.com/#!/docs/openam/12.0.0/admin-guide/chap-tuning

To set the JVM options:

1. Check if the script ***setenv.sh*** exists on the Tomcat8 install in **/usr/share/tomcat8/bin** directory.
    a. If it doesn't create the ***setenv.sh*** file, be sure to change the owner to tomcat, and give it execution permission:

Tomcat setenv.sh Creation

```
> cd /usr/share/tomcat8/bin
> touch setenv.sh

# NOTE: the user:group you specify here should match the other files in the
Tomcat directory. Depending on your installation, you may
#        have a different tomcat group

> chown tomcat8:TOMCAT setenv.sh
> chmod u+x setenv.sh
```

    b. If it already does, ensure it has execution permission and is owned by tomcat, and continue to the next step.
2. Using a command line editor, open ***setenv.sh*** and edit it to match the following. If you already have a ***setenv.sh*** script, then use your discretion for options not specified below that you may have previously set.

setenv.sh contents

```
JAVA_OPTS="-server -Xms2048m -Xmx2048m -XX:NewSize=256m -XX:MaxNewSize=256m
-XX:PermSize=256m -XX:MaxPermSize=256m -Dsun.net.client.defaultReadTimeout=60000
-XX:+UseConcMarkSweepGC -XX:+UseCMSCompactAtFullCollection
-XX:+CMSClassUnloadingEnabled"
```

    a. Note that we're using the recommended 2048m due to the use of OpenDJ. IF you're NOT using OpenDJ, then 1024m should be sufficient. Although they do recommend even higher for a production environment.
3. Once you've saved ***setenv.sh*** to disk, restart Tomcat, and verify it's using your specified settings

```
> service tomcat8 stop # if it was already running
 * Stopping Tomcat servlet engine tomcat8
[ OK ]
> service tomcat8 start
 * Starting Tomcat servlet engine tomcat8
[ OK ]
> psaux | grep tomcat
tomcat8  18364  0.1 29.1 3842824 1179324 ?     Sl   Jan29   4:18
/usr/lib/jvm/default-java/bin/java
-Djava.util.logging.config.file=/var/lib/tomcat8/conf/logging.properties
-Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -server
-Xms2048m -Xmx2048m -XX:NewSize=256m -XX:MaxNewSize=256m -XX:PermSize=256m
-XX:MaxPermSize=256m -Djava.endorsed.dirs=/usr/share/tomcat8/endorsed -classpath
/usr/share/tomcat8/bin/bootstrap.jar:/usr/share/tomcat8/bin/tomcat-juli.jar
-Dcatalina.base=/var/lib/tomcat8 -Dcatalina.home=/usr/share/tomcat8
-Djava.io.tmpdir=/tmp/tomcat8-tomcat8-tmp org.apache.catalina.startup.Bootstrap
start
```

a. Note that the "psaux" command displays the specified JVM parameters

   If Tomcat failed to start, check your JVM parameters. Also check your server's actual memory capacity. Tomcat won't start if the Xms or Xmx are set higher than the actual available memory.

4. Next, make sure that desired JDK is being used. In this case we can see that **/usr/lib/jvm/default-java/bin/java** is being used. Make sure this link is pointing to a JDK 1.7 Java installation:

```
> ls -l /usr/lib/jvm/default-java
lrwxrwxrwx 1 root root 11 Jun  2  2015 /usr/lib/jvm/default-java -> jdk1.7.0_79/
```

   a. If it's not pointing to an Oracle JDK installation, update the default-java link to point to one if you have one, and install one if you don't.

File Descriptors

Since we plan to use OpenDJ, the guide also recommends we increase the file descriptors available to the openam user.

Edit **/etc/security/limits.conf** and add the following lines (if they're not already there):

/etc/security/limits.conf

```
openam soft nofile 65536
openam hard nofile 131072
```

Apache

Configure Apache to forward to **/openam**

1. Add the following to your HTTP site config, e.g., **/etc/apache2/sites-available/default-ssl.conf**

```
Redirect permanent /openam https://[public FQDN]/openam
 Redirect permanent / https://[public FQDN]/openam
```

   a. Note that this assumes solely using https, not http; which is the recommended setup.
   b. [public FQDN] should be your full public URL to the server that matches the SSL certificates, e.g., identity.dcds.ll.mit.edu

2. Add the following to your SSL site config, e.g., **/etc/apache2/sites-available/default-ssl.conf**:

```
ProxyPass /openam http://[host/IP]:8080/openam
ProxyPassReverse /openam http://[host/IP]:8080/openam
ProxyPass / http://[host/IP]:8080/openam
ProxyPassReverse / http://[host/IP]:8080/openam
```

   a. Where **[host/IP]** is the internally accessible hostname or IP address of the server

If it's not already enabled, you'll need to enable the Proxy module for Apache to use the ProxyPass and ProxyPassReverse lines

Setting your FQDN (Fully Qualified Domain Name)

Check to see if your FQDN is already set correctly:

```
> hostname -f
identity.yourhost.com # Example result
```

If your result shows the FQDN, such as ***identity.yourhost.com*** , the FULL URL used to access your server publicly, then your FQDN is already set properly for OpenAM. If not, follow the steps below.

With sudo privileges, edit your **/etc/hosts** file. Edit the line containing "127.0.1.1 localhost" to match the following:

```
<the IP of this server> <your FQDN> <the hostname of the server>
```

For example:

```
123.45.6.7    identity.somehost.com    dcds-identity
```

Now when you run "hostname -f", with the example above, you should see "identity.somehost.com"

This line doesn't have to remain once you've configured OpenAM, but it's required during initial setup of OpenAM to properly populate fields and configuration files.

Next Section
Getting Started - Prepare etc hosts