Installing MetaDB v3.2 on RedHat Enterprise Linux 6

(revision 1.0)

Eric Luhrs
Digital Initiatives Librarian
Lafayette College
April 12, 2011

These instructions will walk users through the installation of MetaDB on a new RHEL6 server, including the installation and configuration of all required software packages. This document is long, but only because it goes into a lot of detail. There are many, many ways to install and configure most Linux packages and services, but the following steps are known to work on an new RHEL6 install. All steps are to be performed as the root user, unless otherwise noted. All commands are printed in `Courier Font`, with multiple commands separated by line breaks. These instructions assume that MetaDB will be the primary webapp on its own server or vhost. Experienced systems administrators should feel free to change shell, tomcat, and postgres usernames, shell group names, asset paths, and other elements to match local environment and deployment preferences.

Make sure that your system is subscribed to the following optional RHN channels:

RHEL Server Supplementary
RHEL Server Optional

1. MetaDB requires the Tomcat application server, Postgres, Java JDK, Ant, and ImageMagick. Use RHEL6's yum package manager to install the following required packages, and allow it to install all additional dependencies:

   ```
   yum install httpd mod_ssl tomcat6 postgresql postgresql-server postgresql-jdbc java-1.6.0-sun java-1.6.0-sun-devel ImageMagick ant ant-nodeps svn
   ```

   NOTE: must use Sun Java, since JAI does not work with openjdk.

2. Configure the tomcat application server:

   a. Create symlink so tomcat can find postgresql.jar (exact source filename may vary):

      ```
      ln -s /usr/share/java/postgresql-jdbc-8.4.701.jar /usr/share/java/tomcat6/postgresql.jar
      ```

b. Modify iptables to permanently allow Tomcat traffic on port 8080, HTTP traffic on port 80, and HTTPS on port 443, then save and restart:

```
iptables -I INPUT 5 -p tcp –m state -–state NEW
–m tcp --dport 8080 -j ACCEPT

iptables -I INPUT 5 -p tcp –m state -–state NEW
–m tcp --dport 80 -j ACCEPT


iptables -I INPUT 5 -p tcp –m state -–state NEW
–m tcp --dport 443 -j ACCEPT

service iptables save

service iptables restart
```

c. To make sure that all webapp files are writable by users in the tomcat group, edit the startup file at /etc/init.d/tomcat6 and add two umask lines above, and one below the line (~190) that switches to the tomcat user (bold lines are new):

```
umask |read DEFAULTMASK
umask 007
[ "$RETVAL" -eq "0" ] && $SU - $TOMCAT_USER -c
"${TOMCAT_SCRIPT} start" >>  $TOMCAT_LOG 2>&1 ||
RETVAL="4"
umask $DEFAULTMASK
```

d. Give tomcat group write access to the webapps directory, and ownership of log and Catalina directories:

```
chmod 775 /var/lib/tomcat6/webapps

chown -R tomcat:tomcat /var/log/tomcat6

chown –R tomcat:tomcat /var/lib/tomcat6/webapps

chown -R tomcat:tomcat /etc/tomcat6/Catalina
```

e. Start the tomcat service:

```
service tomcat6 start
```

3. Configure the postgresql relational database manager:

    a. Setup initial database files:

    ```
    service postresql initdb
    ```

    b. Modify postgres access control, by commenting out example “host” and “local” config lines in ~postgres/data/pg_hba.conf, and adding the following:

    ```
    local all all ident
    host metadb metadb 127.0.0.1 255.255.255.255 md5
    host metadb metadb ::1/128 md5
    ```

    c. Start postgres:

    ```
    service postgresql start
    ```

    d. As the postgres user, create a database and user account (without create role privilege) for metadb:

    ```
    createuser -D -A -P metadb

    createdb -O metadb metadb
    ```

    Make note of the metadb username and password because you will need to provide them later it in the config file.

    e. Test that pg_hba.conf settings are correct by verifying that the metadb user can login to postgres over TCP/IP:

    ```
    psql -U metadb -W -h localhost

    psql -U metadb -W -h 127.0.0.1
    ```

4. Modify HTTPD configuration to proxy tomcat connections over port 80. This allows us to access MetaDB without appending URLs with tomcat port 8080.

    a. Create a new file called /etc/httpd/conf.d/tomcat6.conf with the following lines (replacing HOST.DOMAIN as appropriate):

    ```
    ProxyPass / http://HOST.DOMAIN.edu:8080/
    ProxyPassReverse / http://HOST.DOMAIN.edu:8080/
    ```

    b. The mod_ssl package installs an unsigned HTTPS certificate by default, which may be optionally replaced by specifying a signed certificate in /etc/httpd/conf.d/ssl.conf.

c. Apache can also be optionally instructed to forward all HTTP connections to HTTPS (important for logins, especially with LDAP enabled) by placing the following lines at the end of /etc/httpd/conf/httpd.conf:

```
# Forward all requests to HTTPS
Options +FollowSymLinks
RewriteEngine On

RewriteCond %{HTTPS} off
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
```

d. Start HTTPD:

```
service httpd start
```

5. Install and configure MetaDB:

a. Create directories where MetaDB will look for master image files, and where it will save generated derivatives, then set appropriate ownership and permissions.

i. As a safeguard, the master image directory should be set to read-only for the tomcat user:

```
mkdir –p /assets/metadb/master

chown root:tomcat /assets/metadb/master

chmod 755 /assets/metadb/master
```

ii. The tomcat user requires read/write access to the derivative directory:

```
mkdir –p /assets/metadb/access

chown tomcat:tomcat /assets/metadb/access

chmod 755 /assets/metadb/access
```

b. As the metadb user, unpack the source tar.gz file:

```
tar zxvf metadb-v3.2-release.tar.gz
```

c. Create a metadb shell account with membership in the tomcat group:

```
useradd -G tomcat metadb
```

d. As the metadb user, edit metadb/metadb.conf and set values for your environment.

   i. DB_USER and DB_PASSWORD
      Set to the same values supplied in step 4.d. above.

   ii. TOMCAT_PATH
       Set to the full path for CATALINA_HOME, as specified in your tomcat6.conf file, which for RHEL6 is /var/lib/tomcat6.

   iii. MASTERS_DIRECTORY
        Set to the full path for your master image directory. Should be the same value as the directory created in step 5. A. i, for example /assets/metadb/master. The tomcat user should have read only permission to this directory.

   iv. DERIV_DIRECTORY
       Set to the full path for your derivative image directory. Should be the same value as the directory created in step 5.A.ii, for example /assets/metadb/access. The tomcat user should have read and write permissions to this directory.

   v. SYMBOLIC_LINK_DIRECTORY
      Set to the full path of your DERIV_DIRECTORY if you don't want to store generated derivatives in your deployment directory. The ant “deploy” task will re-create this link whenever the webapp is updated.

6. The `utils/download-required-jars.sh` script will attempt to download all JARs required by MetaDB. After download it will extract just the needed files and delete everything else. The URLs were correct when the current version of MetaDB was released, but you may have to adjust if addresses change in the future.

   a. Change to the appropriate directory:

```
cd ~metadb/metadb/webapp-src/lib

../../utils/download-required-jars.sh
```

   b. When script is finished, verify that commons-fileupload, commons-io, commons-lang, jai_codec, jai_core, javacsv, and sanselan JARs are in the webapp-src/lib directory.

7. As the metadb user, run the configure script to check dependencies, create the Ant build file, and generate the SQL source file.

8. As the metadb user, run the SQL script to create the skeleton database and tables:

   ```
   psql -f metadb/metadb-install.sql
   ```

   This script first tries to drop the existing database and tables, so ignore these errors if this is the first time running this script.

9. As the metadb user, create an administrator account for the MetaDB web application by running the `utils/admin.sh` script.

10. As the metadb user, set the CLASSPATH to include servlet.jar (or add to .bashrc):

    ```
    export CLASSPATH=$CLASSPATH:/usr/share/java/servlet.jar
    ```

11. As the metadb user, use Ant to build MetaDB:

    ```
    ant deploy
    ```

12. Make metadb the default webapp (required for zoom/download URLs to work):

    ```
    ln –s /var/lib/tomcat6/webapps/metadb
    /var/lib/tomcat6/webapps/ROOT/
    ```

13. Start tomcat:

    ```
    service tomcat6 restart
    ```

14. If everything works correctly, you should now be able to load MetaDB in your browser and login as the admin user created in step 9. Your first steps will likely be to create a project, configure fields, parse master images, and create user accounts.