

Geronimo 3.0.x Release Process

Geronimo 3.0.x Release Process

Reference <http://www.apache.org/dev/publishing-maven-artifacts.html>

Have 3.0.1 release as an example.

Release Checklist

1. Better to use a non-Windows system to create the release candidate

- Dos line endings makes all unix shell scripts unexecutable
 - <https://issues.apache.org/jira/browse/GERONIMO-5440>

2. mvn verify

- Refer to the same section in [Geronimo 2.1.x Release Process](#)

3. Manually update some files:

- Updates all pom.xml files to search "SNAPSHOT" to ensure there is no snapshot dependencies
- Updates plugin-list url in \$SRC\framework\configs\plugin\pom.xml
- Updates ##VERSION## in README.txt and RELEASE_NOTES.txt in source code root folder and \$SRC\framework\configs\karaf-framework
- Updates JIRAs in RELEASE_NOTES.txt (bugs, improvement, new features, known issues, and limitations)
- Updates the copyright year number in NOTICE files
- Updates some un-released modules' versions to 3.0.1. Refer to this ant scripts to update the versions in batch.

```

<target name="manualupdate-versions">
  <echo>Updating un-released modules' versions manually</echo>
  <replace dir="${user.dir}" token="version>3.0.1-SNAPSHOT" value="version>3.0.1" summary="yes">
    <include name="assemblies/geronimo-jetty8-javaee6/pom.xml" />
    <include name="assemblies/geronimo-jetty8-javaee6-web/pom.xml" />
    <include name="assemblies/geronimo-jetty8-minimal/pom.xml" />
    <include name="framework/modules/geronimo-config-groovy-transformer/pom.xml" />
    <include name="pluginingroups/clustering-jetty/pom.xml" />
    <include name="pluginingroups/clustering-tomcat/pom.xml" />
    <include name="pluginingroups/webservices-cxf/pom.xml" />
    <include name="plugins/activemq/activemq-jetty-server/pom.xml" />
    <include name="plugins/activemq/activemq-webconsole/pom.xml" />
    <include name="plugins/activemq/activemq-webconsole-jetty/pom.xml" />
    <include name="plugins/activemq/activemq-webconsole-tomcat/pom.xml" />
    <include name="plugins/axis/axis-jetty-server/pom.xml" />
    <include name="plugins/ca-helper/pom.xml" />
    <include name="plugins/ca-helper/ca-helper-jetty/pom.xml" />
    <include name="plugins/ca-helper/ca-helper-tomcat/pom.xml" />
    <include name="plugins/ca-helper/geronimo-ca-helper/pom.xml" />
    <include name="plugins/clustering/clustering-it/pom.xml" />
    <include name="plugins/clustering/clustering-it/clustering-test/pom.xml" />
    <include name="plugins/clustering/clustering-it/customer-jetty/pom.xml" />
    <include name="plugins/clustering/clustering-it/plugin-farm-it/pom.xml" />
    <include name="plugins/clustering/clustering-it/sample-datasource/pom.xml" />
    <include name="plugins/connector-1_6/geronimo-connector-server/pom.xml" />
    <include name="plugins/console/console-jetty-server/pom.xml" />
    <include name="plugins/console/console-tomcat-server/pom.xml" />
    <include name="plugins/corba/corba-server/pom.xml" />
    <include name="plugins/cxf/cxf-jaxws-tools/pom.xml" />
    <include name="plugins/cxf/cxf-tools/pom.xml" />
    <include name="plugins/cxf/geronimo-cxf-tools/pom.xml" />
    <include name="plugins/jetty8/jetty-server/pom.xml" />
    <include name="plugins/monitoring/mconsole-jetty-server/pom.xml" />
    <include name="plugins/monitoring/mconsole-jetty-server/mconsole-itest-simple/pom.xml" />
    <include name="plugins/openejb/geronimo-openejb-server/pom.xml" />
    <include name="plugins/uddi/uddi-jetty-server/pom.xml" />
    <include name="plugins/wab/web-jetty-server/pom.xml" />
    <include name="plugins/wab/web-tomcat-server/pom.xml" />
    <include name="plugins/welcome/welcome-jetty-server/pom.xml" />
    <include name="plugins/welcome/welcome-tomcat-server/pom.xml" />
    <include name="plugins/wink/wink-tomcat-server/pom.xml" />
  </replace>
</target>

```

- Commit them

4. mvn release:prepare -DdryRun=true -Pall-subprojects

5. Release Prepare

- Before doing release prepare, clean up you local repository to avoid the bad staging release artifacts to be included in the geronimo release. see [reference](#).
- This will update the versions in branch 3.0 and create the release tag
 - Manually remove all *.log files in the source code root folder, otherwise those might be packed into the source code zip files.
 - mvn release:clean -Pall-subprojects
 - mvn release:prepare -Pall-subprojects
 - you need "mvn clean install -Dstage=bootstrap" in midway

6. Release Perform

- This will stage the release artifacts,
 - mvn release:perform -Pall-subprojects
- In Apache nexus repository, click "close" <https://repository.apache.org/index.html#welcome>

7. Vote

- Vote in mailing list, meanwhile wait TCK results. Sample release vote email as following:

[VOTE] Release Geronimo 3.0.0

Hi Devs,

With correction of legal files, here we have a new release candidate for vote. Please help vote at your earliest convenient time.

The server code up for vote is:

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/geronimo/3.0.0/geronimo-3.0.0-source-release.tar.gz>

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/geronimo/3.0.0/geronimo-3.0.0-source-release.zip>

The binary code up for vote is:

Java EE 6 Full Profile Tomcat Assemblies:

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-javaee6/3.0.0/geronimo-tomcat7-javaee6-3.0.0-bin.tar.gz>

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-javaee6/3.0.0/geronimo-tomcat7-javaee6-3.0.0-bin.zip>

Java EE 6 Web Profile Tomcat Assemblies:

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-javaee6-web/3.0.0/geronimo-tomcat7-javaee6-web-3.0.0-bin.tar.gz>

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-javaee6-web/3.0.0/geronimo-tomcat7-javaee6-web-3.0.0-bin.zip>

Little-G Tomcat Assemblies:

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-minimal/3.0.0/geronimo-tomcat7-minimal-3.0.0-bin.tar.gz>

<https://repository.apache.org/content/repositories/orgapachegeronimo-013/org/apache/geronimo/assemblies/geronimo-tomcat7-minimal/3.0.0/geronimo-tomcat7-minimal-3.0.0-bin.zip>

Staging repo is:

<https://repository.apache.org/content/repositories/orgapachegeronimo-013>

The tag has created at:

<http://svn.apache.org/repos/asf/geronimo/server/tags/geronimo-3.0.0>

Java EE 6 TCKs all passed!

Vote will be at least open for 72 hours.

[] +1 approve
[] +0 no opinion
[] -1 disapprove (and reason why)

- Post "**VOTE PASS**xxx" in the subject, and summarize the vote status in the body when vote close.

8. Release artifacts

- In Apache nexus, click "release"
 - the artifacts will be synchronized to maven central repository in some time.

9. Update geronimo-plugins.xml

- delete your local `~/m2/repository/geronimo-plugins.xml`
- build tag 3.0.0, which will generate a new `geronimo-plugins.xml` in `~/m2/repository/`
- do the actions as described here – <https://cwiki.apache.org/GMOxPMGT/geronimo-server-release-process.html> – in step 12.

10. Check-in artifacts into dist svnpubsub

- Check-in the artifacts into <https://dist.apache.org/repos/dist/release/geronimo/>
 - Update <https://svn.apache.org/repos/asf/geronimo/KEYS> and <https://dist.apache.org/repos/dist/release/geronimo/KEYS> with your public key (if it is not there).
 - Check-in the artifacts and their checksums (*.tar.gz.md5, *.zip.md5, *.tar.gz.sha1, *.zip.sha1, *.asc) into <https://dist.apache.org/repos/dist/release/geronimo/3.0.0>

11. Announce in Mailing list and Post news in homepage

- Modify <http://geronimo.apache.org/downloads.html>.
- Add a new page list the artifacts that can be downloaded.
- Modify frontpage and add a news.

12. Update the security advisory page

- Add a section to the new release at <https://cwiki.apache.org/confluence/display/GMOxSITE/3.0.x+Security+Report>
- If there are outstanding advisories for vulnerabilities fixed by this release, move the vulnerability descriptions to the new release section.

13. Manually update files in the 3.0 branch after release

- update 3.0.1-SNAPSHOT to ##VERSION## in README.txt and RELEASE_NOTES.txt
- remove the JIRA list in RELEASE_NOTES.txt (bugs, improvement, new features, limitations)
- search "3.0.0" and change them to "3.0.1-SNAPSHOT"
- Update artifact-alias, add version 3.0.0 in artifact-alias after 3.0.0 release
 - /framework/configs/pom.xml
 - /plugins/client/pom.xml
 - /plugins/corba/client-corba-yoko/pom.xml
 - /plugins/pom.xml
- commit them

Prerequisite

1. Use Genesis 2.0 as a parent pom

```
<parent>
  <groupId>org.apache.geronimo.genesis</groupId>
  <artifactId>genesis-java5-flava</artifactId>
  <version>2.0</version>
</parent>
```

- genesis-java5-flava-2.0.pom
 - genesis-default-flava-2.0.pom
 - genesis-2.0.pom
 - apache-6.pom

2. Use Maven 3.0.3

- Enable Apache Servers (refer: <http://maven.apache.org/developers/committer-settings.html>)

```

<settings>
...
<servers>
  <!-- To publish a snapshot of some part of Maven -->
  <server>
    <id>apache.snapshots.https</id>
    <username> <!-- YOUR APACHE LDAP USERNAME --> </username>
    <password> <!-- YOUR APACHE LDAP PASSWORD --> </password>
  </server>
  <!-- To publish a website of some part of Maven -->
  <server>
    <id>apache.website</id>
    <username> <!-- YOUR APACHE LDAP USERNAME --> </username>
    <filePermissions>664</filePermissions>
    <directoryPermissions>775</directoryPermissions>
  </server>
  <!-- To stage a release of some part of Maven -->
  <server>
    <id>apache.releases.https</id>
    <username> <!-- YOUR APACHE LDAP USERNAME --> </username>
    <password> <!-- YOUR APACHE LDAP PASSWORD --> </password>
  </server>
  <!-- To stage a website of some part of Maven -->
  <server>
    <id>stagingSite</id> <!-- must match hard-coded repository identifier in site:stage-deploy -->
    <username> <!-- YOUR APACHE LDAP USERNAME --> </username>
    <filePermissions>664</filePermissions>
    <directoryPermissions>775</directoryPermissions>
  </server>
  ...
</servers>
</settings>

```

reference:

It is highly recommended to use Maven's password encryption capabilities for your passwords. <http://maven.apache.org/guides/mini/guide-encryption.html>

3. Setup PGP Keys (for the ones who be the release manager the first time)

- Download gnupg2
- Generate your PGP Key (refer: <http://www.apache.org/dev/openpgp.html>) so that maven-release-plugin can sign your built artifacts when do release:perform
 - How To Avoid SHA-1
 - How To Generate a Strong Key
- Update Maven's settings.xml with following:

```

<settings>
...
<profiles>
  <profile>
    <id>apache-release</id>
    <properties>
      <gpg.passphrase> <!-- YOUR KEY PASSPHRASE --> </gpg.passphrase>
    </properties>
  </profile>
</profiles>
...
</settings>

```

- Meanwhile, append your public key to <https://svn.apache.org/repos/asf/geronimo/KEYS> and <https://dist.apache.org/repos/dist/release/geronimo/KEYS> so that user can verify the artifacts you released.
 - gpg --gen-key
 - RSA and RSA (default), 4096
 - gpg --list-sigs "xxxxxx" && gpg --armor --export "xxxxxx" > xxxxxx.key
 - "cat" your public key to above KEYS file

reference:

<http://maven.apache.org/developers/release/pmc-gpg-keys.html>