

# Guide to Directory Releases

## Introduction

Releasing Apache Directory Projects can be involved. This release guide will walk you through the process first by preparing your maven and gpg configuration then by leading you through the release of various subprojects at Directory.

There may be a few things you'll need to setup before you can release. This release guide is geared to work off of a UNIX based system that has gpg installed. If you're using Windows then I feel for you 😊.

We use Maven version 3.0.2+ and JDK 1.6.0 to build all Directory subprojects even if they run on JDK 1.5.

## Maven Settings

You'll need a settings section for the Nexus and people.apache.org servers with a password or a path to the SSH key used. Here's what my settings.xml file in ~/.m2 looks like:

```
<settings>

  <servers>
    <!-- To publish a snapshot of some part of Maven -->
    <server>
      <id>apache.snapshots.https</id>
      <username>username</username>
      <password>*****</password>
    </server>

    <!-- To publish a website using Maven -->
    <server>
      <id>apache.directory</id>
      <username>username</username>
      <privateKey>/Users/username/.ssh/id_rsa</privateKey>
      <filePermissions>664</filePermissions>
      <directoryPermissions>775</directoryPermissions>
    </server>

    <!-- To stage a release of some part of Maven -->
    <server>
      <id>apache.releases.https</id>
      <username>username</username>
      <password>*****</password>
    </server>
  </servers>

  <profiles>
    <profile>
      <id>apache-release</id>
      <properties>
        <gpg.passphrase>*****</gpg.passphrase>
      </properties>
    </profile>
  </profiles>
</settings>
```

Just replace your username, passwords and paths. Note that the username and password is your Apache LDAP account.




You'll need to provide the passphrase in the settings.xml to access the gpg secret key installed on your host. This is due to a bug with the passphrase prompt in the maven-gpg-plugin. So unfortunately we must provide the passphrase in the settings.xml file in clear text. This should change in the future when this bug is fixed. Note that this passphrase is put into the release profile which we activate to properly sign and release the artifacts and poms via the release plugin.

## GPG Key

All subprojects are configured to deploy signatures for the artifacts uploaded to the repository. The gpg plugin will check use the default gpg key for the user deploying the project with the release:perform directive of the release plugin. This will prompt you for the passphrase for the default key. If you do not have one setup the build will fail.

You can generate and upload a PGP key to a PGP keyserver using the following commands:

```
gpg --gen-key
gpg --fingerprint
gpg --keyserver subkeys.pgp.net --send-keys <your key's id from last command>
```

 Make sure to have created the .pgpkey in your p.a.o/~ directory and to have added your public key to the [KEYS](#) file. See also <http://people.apache.org/~henkp/repo/faq.html#4>

## Releasing Directory Projects and Making Release Announcements

|                                     |   |
|-------------------------------------|---|
| Releasing Skins                     | <a href="#">Releasing Skins</a>                     |
| Releasing Checkstyle Configuration  | <a href="#">Releasing Checkstyle Configuration</a>  |
| Releasing JUnit Addons              | <a href="#">Releasing JUnit Addons</a>              |
| Releasing Apache Directory LDAP API | <a href="#">Releasing Apache Directory LDAP API</a> |
| Releasing ApacheDS                  | <a href="#">Releasing ApacheDS</a>                  |
| Releasing ApacheDS Manuals          | <a href="#">Releasing ApacheDS Manuals</a>          |
| Releasing Studio                    | <a href="#">Releasing Studio</a>                    |
| Releasing Studio Maven Plugin       | <a href="#">Releasing Studio Maven Plugin</a>       |
| Releases Announcements              | <a href="#">Release Announcements</a>               |

## Create a tar.gz with resource fork (Mac OS X only)

When creating a tar.gz which is used to upload generated documentation on the Apache CMS ([guide here](#)), if you're running on Mac OS X you will need to remove the resource fork ([a special construct of the Mac OS operating system used to store structured data in a file](#)). This can be done by exporting two properties before creating the tar.gz file via the command line. Here are those two exports:

```
export COPY_EXTENDED_ATTRIBUTES_DISABLE=true
export COPYFILE_DISABLE=true
```

Promoted by third party (update information may need to be sent)

| Project | Reference   | Contact  |
|---------|---|--|
| Studio  | <a href="#">Open Source Rich client platform (RCP) applications</a> | <a href="mailto:news@eclipse.org">news@eclipse.org</a> |