

# Artifactory Pro - Artifact Repository

- [About](#)
- [How to access](#)
  - [Login with GÉANT AAI](#)
  - [Get access to specific repository](#)
- [Short guides](#)
  - [How to use Artifactory as Maven repository](#)
    - [GÉANT Artifactory Repositories](#)
    - [Working with Repository in Maven](#)
    - [Using Artifactory as maven repositories mirror](#)
  - [How to use Artifactory as docker registry](#)
    - [How to use with Docker client](#)
- [Related articles](#)

## About

The GÉANT Software Repository is based on Artifactory (<http://www.jfrog.com/products.php>) with Pro Power Pack ( <http://www.jfrog.com/pro-features.php>). Artifactory provide base Maven/Gradle repository functionality: you can download artefacts and deploy own snapshot and release artefacts like in any repository. It can be used as mirror for any external repository. Also it provide nice GUI to manage your artefacts, deploy files manually and integration Bamboo. Integration with Bamboo (<https://ci.geant.org>) allow eg. performing License Check process after artefacts deploy, deployment from Bamboo with global defined credentials etc.

Artifactory Pro Power Pack allow us to support different types of repository wider adopted in software development, see available predefined repositories types below:

- [Build Technologies](#)
  - [Maven](#)
  - [Gradle](#)
  - [Ivy](#)
- [Infrastructure Package Managers](#)
  - [Docker](#)
  - [Yum](#)
  - [Puppet](#)
  - [Chef](#)
  - [Debian](#)
  - [Vagrant](#)
- [Development Language Package Managers](#)
  - [Npm](#)
  - [Nuget](#)
  - [Conan](#)
  - [Bower](#)
  - [Opkg](#)
  - [CocoaPods](#)
  - [PHP](#)
  - [P2](#)
  - [Git](#)
  - [gitLFS](#)
  - [Ruby](#)
  - [Python](#)
  - [Go/vGo](#)
  - [Helm Chart](#)
  - [CRAN](#)

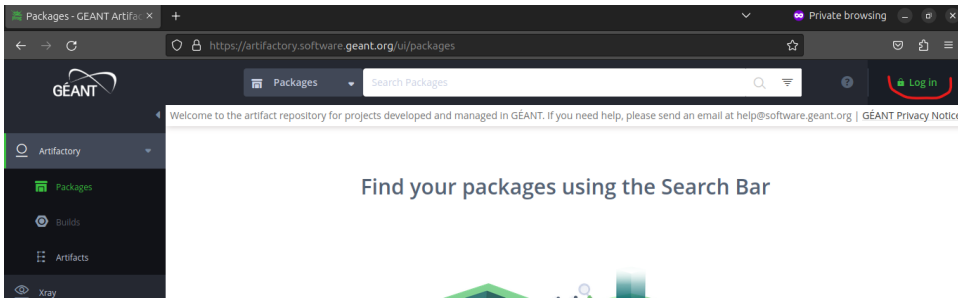
## How to access

URL: <https://artifactory.software.geant.org>

Artifactory uses federated authentication. During login process you must choose your Identity Provider. After successful authentication you will be redirect back to Artifactory home page.

## Login with GÉANT AAI

1. Go to <https://artifactory.software.geant.org/>
2. Click the **Log in** link in the top right corner



3. Click on the **GEANT AAI**

## WELCOME TO JFROG

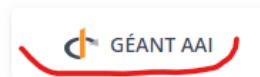
\* Username

\* Password

☐ Remember me

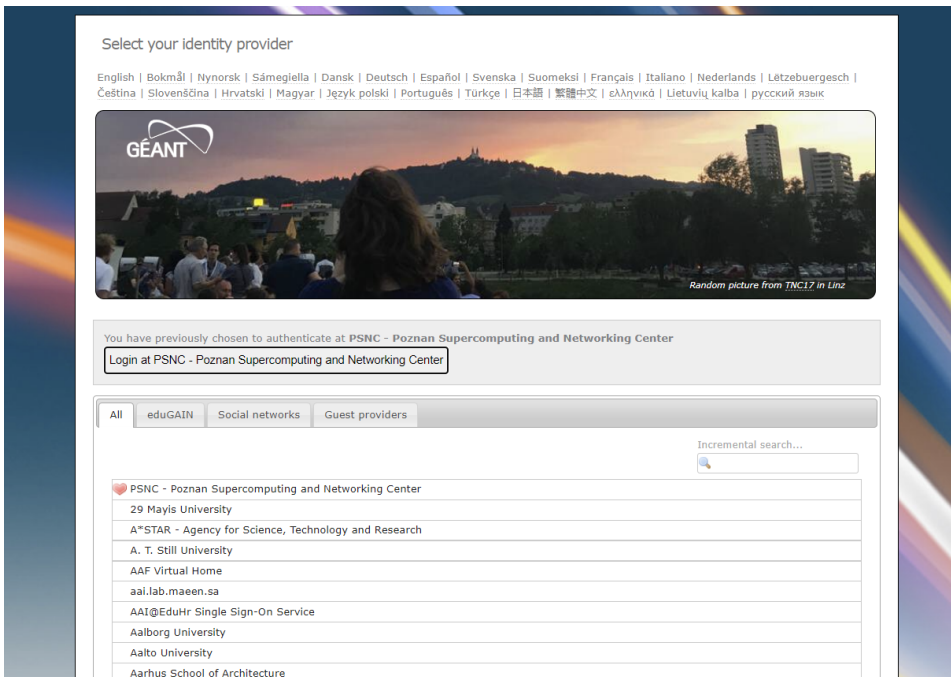
Login

OR SIGN IN WITH



SAML SSO

4. You will be redirected to [login.geant.org](https://login.geant.org) where you can choose your Identity Provider and use it to log in.



#### GÉANT AAI Registration

If you are logging in for the first time using GÉANT AAI Service, you will see the message "You tried to access (a service via) GÉANT AAI Service, but we couldn't find a registration Let's fix that!" and you have to register first. To do so, please follow the instructions from [Registration in GÉANT AAI Service](#)

5. After successful login you will be redirected to Artifactory main page

## Get access to specific repository

To get access to specific repository:

1. Log in to <https://artifactory.software.geant.org>
2. Send access request - create issue on [Help Desk Portal](#) or send e-mail at [help@software.geant.org](mailto:help@software.geant.org) You have to provide us:
  - your user name in Artifactory (federated-user-XXX)
  - repository name

For "non GUI" access to repository for automated deployment tools (Jenkins, Bitbucket), you can:

- generate API key for your account (federated-user-XXX) - see [How to use API Key](#)
- request for local account with password - this is the simplest way for CI purposes

## Short guides

### How to use Artifactory as Maven repository

#### GÉANT Artifactory Repositories

Artifactory allow to create three repository types: local, remote and virtual. In GÉANT we use all of them, to understand how it works read <http://wiki.jfrog.org/confluence/display/RTF/Understanding+Repositories> page on Artifactory documentation page.

Each GÉANT development projects have their own local repositories:

- project-release-local - This repository contains all project release artefacts.
- project-snapshot-local - This repository contains last 3\* project snapshot artefacts.
- project-ext-lib (optional) This repository contains dependency unavailable in public Maven repositories)

### Working with Repository in Maven

1. Request for project internal repositories in Help Desk Portal (<https://jira.software.geant.org/servicedesk/customer/portal/2>)
2. Setting Up a Maven Deployment Descriptor You can follow: <http://wiki.jfrog.org/confluence/display/RTF/Configuring+Deployment> or:
  - 2.1. Provide configuration in main POM file eg.:

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://maven.apache.org/POM/4.0.
0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>net.geant.ishare</groupId>
  <artifactId>ishare</artifactId>
  <packaging>pom</packaging>
  <version>2.0.1-SNAPSHOT</version>
  <name>${project.artifactId}</name>
  <description>I-SHARe Project</description>
  <distributionManagement>
    <repository>
      <id>GEANT Artifactory</id>
      <name>GEANT Artifactory ishare-release</name>
      <url>https://artifactory.software.geant.org/artifactory/ishare-release-local</url>
    </repository>
    <snapshotRepository>
      <id>GEANT Artifactory</id>
      <name>GEANT Artifactory ishare-snapshots</name>
      <url>https://artifactory.software.geant.org/artifactory/ishare-snapshot-local</url>
    </snapshotRepository>
  </distributionManagement>
</project>
```

## 2.2. Provide your credentials in settings.xml file.

- Get into your profile page (click on your login name on the upper-right corner) and type-in your current password.
  - Once you enter a correct password you will see your password in the Encrypted Password field.
  - Copy this value (including the { ... } prefix) or use the sample server xml snippet in your settings.xml (you'd have to change the server name to match the id of your repository).
- Eg.:

```
<server>
  <id>GEANT Artifactory</id>
  <username>psnc.kupinski</username>
  <password>\{DESede\}00000000</password>
</server>
```

## 3. Try to deploy snapshot artefacts.

Reference:

- Artifactory User Guide: <https://www.jfrog.com/confluence/display/RTF/Welcome+to+Artifactory#WelcometoArtifactory-HowisthisGuideOrganized?>

## Using Artifactory as maven repositories mirror

Easiest way to use GN Artifactory as global maven repositories mirror is to define profile in your POM file. If you want always use GN Artifactory you can activate profile by default. See below example how to configure it:

```
...
    <profile>
      <activation>
        <activeByDefault>true</activeByDefault>
      </activation>
      <id>artifactory</id>
      <repositories>
        <repository>
          <id>central</id>
          <name>libs-release</name>
          <url>https://artifactory.software.geant.org/artifactory/repo</url>
        </repository>
      </repositories>
      <pluginRepositories>
        <pluginRepository>
          <id>central</id>
          <name>plugins-release</name>
          <url>https://artifactory.software.geant.net/artifactory/repo</url>
        </pluginRepository>
      </pluginRepositories>
    </profile>
  ...
```

## How to use Artifactory as docker registry

Artifactory can be used to serve Docker images like Docker Registry. It is just another type of repository and could be browsed and managed in Artifactory web dashboard.

Address: [artifactory.software.geant.org/<REPO\\_NAME>/<IMAGE\\_NAME>:<TAG>](https://artifactory.software.geant.org/<REPO_NAME>/<IMAGE_NAME>:<TAG>)

### How to use with Docker client

#### 1. Login

##### Docker login

```
docker login artifactory.software.geant.org
```

And provide your Artifactory username and password

To manually set your credentials copy the following snippet to your `~/docker/config.json` file.

##### Credentials

```
{
  "auths": {
    "https://artifactory.software.geant.org": {
      "auth": "<USERNAME>:<PASSWORD>",
      "email": "youremail@email.com"
    }
  }
}
```

#### 2. Push image to registry

To push an image tag an image using the docker tag and then docker push command.

##### Docker push

```
docker tag <MY_IMAGE_ID> artifactory.software.geant.org/<REPO_NAME>/<IMAGE_NAME>:<TAG>
docker push artifactory.software.geant.org/<REPO_NAME>/<IMAGE_NAME>:<TAG>
```

#### 3. Pull image

#### Docker pull

```
docker pull artifactory.software.geant.org/<REPO_NAME>/<IMAGE_NAME>:<TAG>
```

## Related articles



The JFrog Academy provides free, unlimited access to On-Demand Online Training, some of them are dedicated to DevOps and could help you to get deep knowledge about useful Artifactory features and also build automation and DevOps best practices:

- Artifactory Basics (60m)
- DevOps Pipelines with Artifactory (120m)

See more at: <https://academy.jfrog.com>

- [Artifactory Pro - Artifact Repository](#)