



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Setting up Genesys Multicloud CX Private Edition

[Downloading your Genesys Multicloud CX containers](#)

Contents

- [1 Overview](#)
- [2 Accessing repositories on JFrog](#)
- [3 Signing up for update notifications](#)
- [4 Setting up automated downloads](#)
 - [4.1 Downloading using Docker CLI](#)
 - [4.2 Downloading using the Helm CLI](#)
- [5 Downloading using cURL](#)
- [6 Downloading manually](#)
- [7 Additional reading material](#)

Genesys Multicloud CX containers are accessible through JFrog. You can also automate downloads to set up a Continuous Delivery (CD) pipeline.

Related documentation:

-
-

RSS:

- [For private edition](#)

Overview

Important

Use the information on this page to set up and manage your own repository. You can replicate the Genesys repository at your end. Do not pull images directly to your production environment. Refer to [Setting up a CD pipeline](#) for more information on managing your own repository.

Genesys uses JFrog to deliver and distribute its release containers. JFrog is a hybrid, universal, end-to-end devops platform. It is a fully automated platform for distributing software releases from code to production. You can pull new releases from the JFrog Artifactory Edge repository.

Warning

Genesys Engage will deprecate the JFrog Edge on June 15, 2024. Please contact your Genesys Account Representative if your organization requires access to the new utility.

Use the information in this topic to set up your Continuous Delivery (CD) pipeline.

Important

Your CD pipeline must accommodate any necessary steps to meet your corporate requirements such as, performing security scans and validation testing.

The Engage private edition release containers can be accessed through either of the following:

- Artifactory Edge Portal
- Artifactory Edge API
- Command Line Interfaces (CLI) for Docker, Helm, or cURL.

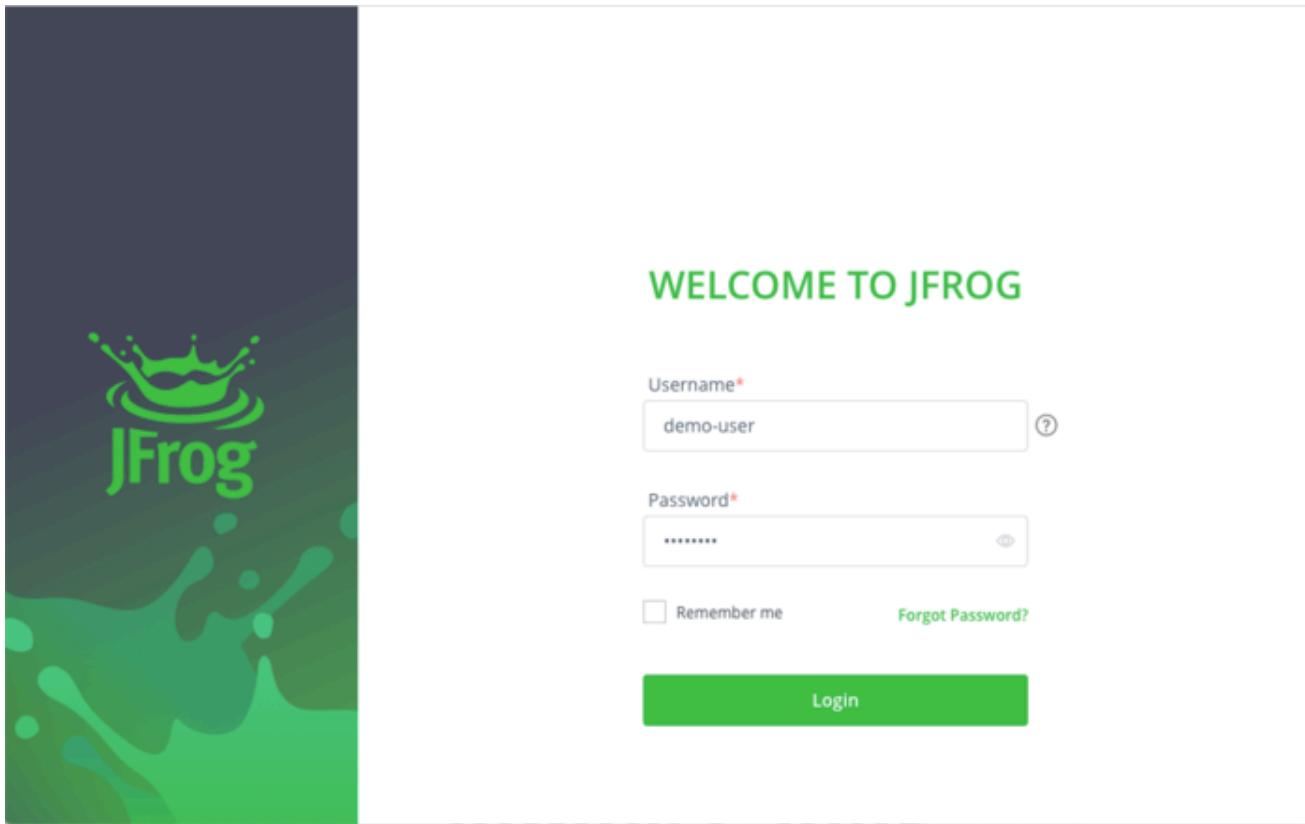
You can set up automated downloads for any of the available repositories using the **Set Me Up** screen for the corresponding repository.

Accessing repositories on JFrog

1. Navigate to the following URL using your browser:
<https://pureengageuse1.jfrog.io/ui/login/>
2. Use your JFrog credentials from Genesys to log in.

Important

Credentials to access the Genesys repository on JFrog are automatically emailed to new accounts. Please contact your Genesys Account Representative if you have not received your credentials.



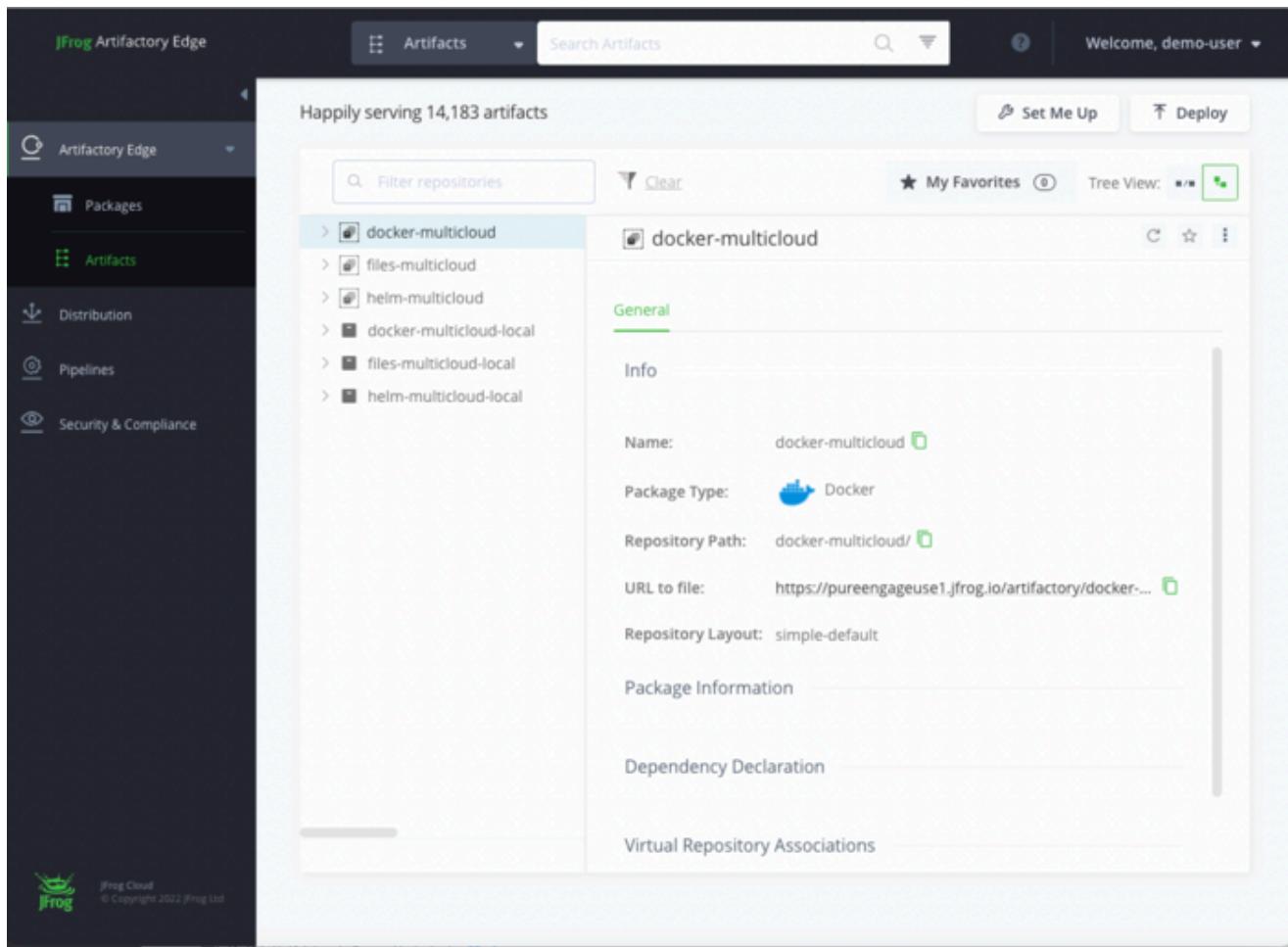
Artifactory Edge contains the following six Genesys repositories:

Important

A virtual repository aggregates several repositories with the same package type under a common URL.

Name of repository	Type of artifacts
helm-multicloud-local	Helm charts local repository
files-multicloud-local	Configuration files local repository
docker-multicloud-local	Docker local repository
helm-multicloud	Helm virtual repository
files-multicloud	Files virtual repository
docker-multicloud	Docker virtual repository

Once you log in, select **Artifacts** from the **Artifactory Edge** menu from the left pane. All repositories available for download from Genesys are listed.

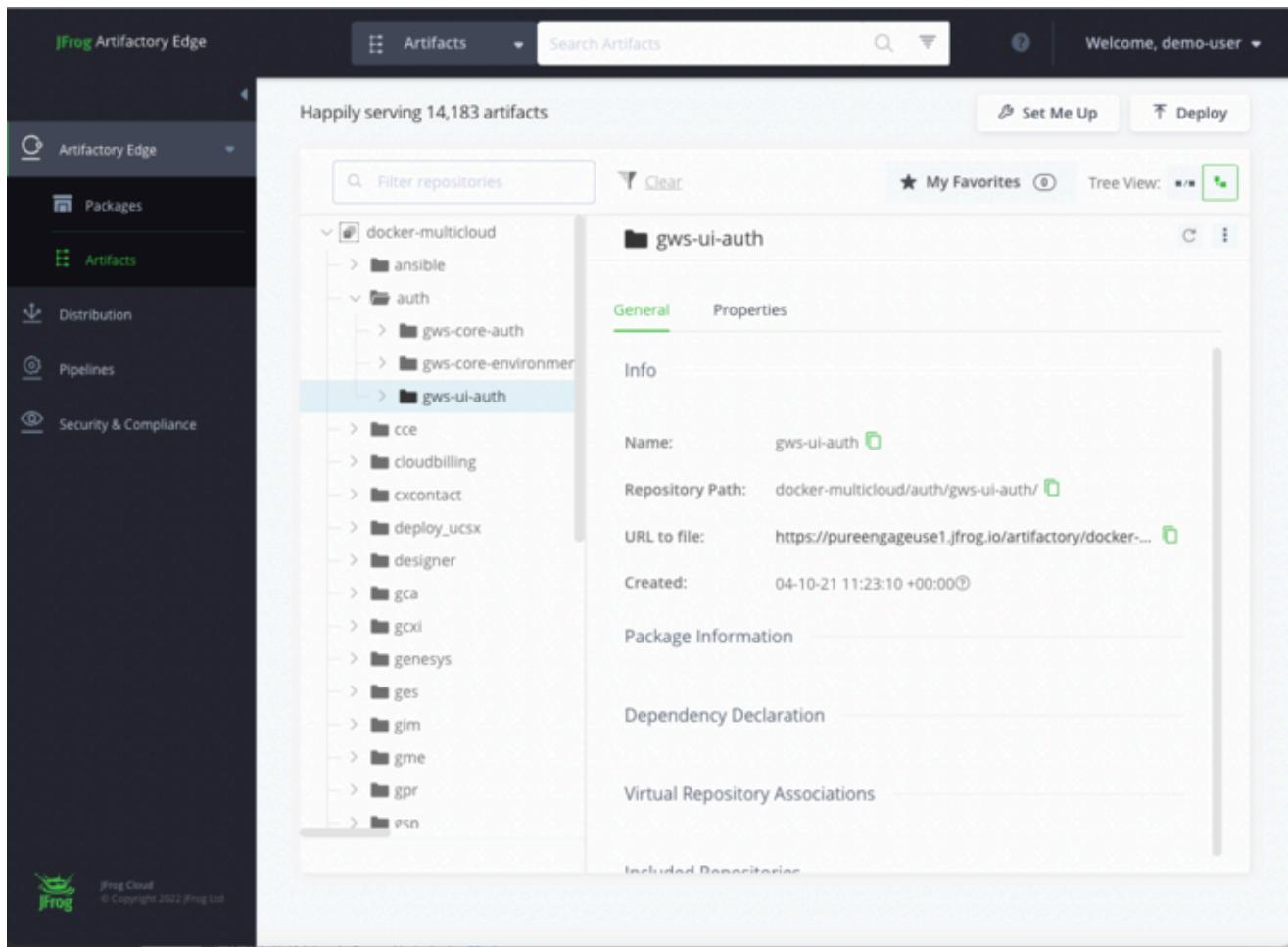


The screenshot shows the JFrog Artifactory Edge interface. The left sidebar has 'Artifactory Edge' selected under 'Artifacts'. The main area displays a list of repositories: docker-multicloud, files-multicloud, helm-multicloud, docker-multicloud-local, files-multicloud-local, and helm-multicloud-local. The 'docker-multicloud' repository is selected and expanded. The right pane shows the 'Info' tab for the 'docker-multicloud' repository. The repository details are as follows:

Name:	docker-multicloud
Package Type:	Docker
Repository Path:	docker-multicloud/
URL to file:	https://pureengageuse1.jfrog.io/artifactory/docker-...
Repository Layout:	simple-default

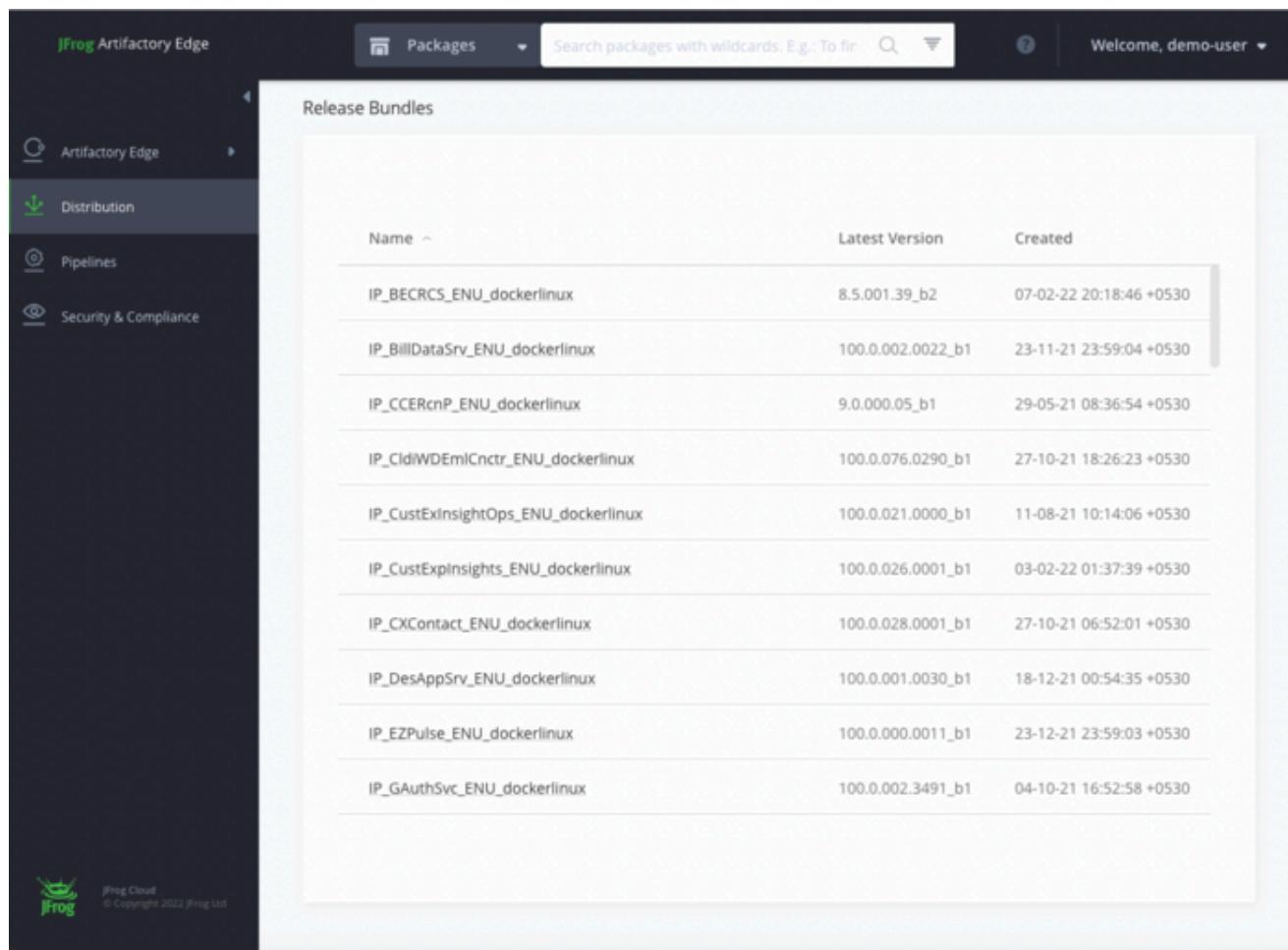
Below the 'Info' tab, there are sections for 'Package Information', 'Dependency Declaration', and 'Virtual Repository Associations'.

You can expand each repository and navigate to any of the files within a folder to view its properties on the left pane. Note that as all available files are listed, you must navigate to the one you require based on the date and version number.



The screenshot shows the JFrog Artifactory Edge interface. The left sidebar has 'Artifactory Edge' selected under 'Artifacts'. The main area displays a tree view of artifacts. A folder named 'gws-ui-auth' is selected. The right panel shows the 'General' tab for this artifact, with fields for Name (gws-ui-auth), Repository Path (docker-multicloud/auth/gws-ui-auth/), URL to file (https://pureengageuse1.jfrog.io/artifactory/docker-...), and Created (04-10-21 11:23:10 +00:00). Other tabs include Properties, Info, Package Information, Dependency Declaration, Virtual Repository Associations, and Included Dependencies.

The **Distribution** view lists all files IPs and the files within each IP.



The screenshot shows the JFrog Artifactory Edge interface. The left sidebar includes links for 'Artifactory Edge', 'Distribution', 'Pipelines', and 'Security & Compliance'. The main content area is titled 'Release Bundles' and displays a table of packages. The table has columns for 'Name', 'Latest Version', and 'Created'. The packages listed are:

Name	Latest Version	Created
IP_BECRCS_ENU_dockerlinux	8.5.001.39_b2	07-02-22 20:18:46 +0530
IP_BillDataSrv_ENU_dockerlinux	100.0.002.0022_b1	23-11-21 23:59:04 +0530
IP_CCERcnP_ENU_dockerlinux	9.0.000.05_b1	29-05-21 08:36:54 +0530
IP_CldiWDEmlCnctr_ENU_dockerlinux	100.0.076.0290_b1	27-10-21 18:26:23 +0530
IP_CustExInsightOps_ENU_dockerlinux	100.0.021.0000_b1	11-08-21 10:14:06 +0530
IP_CustExplnights_ENU_dockerlinux	100.0.026.0001_b1	03-02-22 01:37:39 +0530
IP_CXContact_ENU_dockerlinux	100.0.028.0001_b1	27-10-21 06:52:01 +0530
IP_DesAppSrv_ENU_dockerlinux	100.0.001.0030_b1	18-12-21 00:54:35 +0530
IP_EZPulse_ENU_dockerlinux	100.0.000.0011_b1	23-12-21 23:59:03 +0530
IP_GAuthSvc_ENU_dockerlinux	100.0.002.3491_b1	04-10-21 16:52:58 +0530

JFrog Cloud
© Copyright 2022 JFrog Ltd.

You can set up automated downloads for any repository using the **Set Me Up** screen for the corresponding repository.

Signing up for update notifications

When you log in for the first time, sign up for email notifications on any updates to the packages in the repository. Note that you can set this up later too. But we recommend you set this up in order to receive regular notifications on any updates to the packages.

1. Click on your username at the top right corner of the screen.
2. Select the **Edit Profile** option from the drop-down.

JFrog Artifactory Edge

Packages

Search packages with wildcards. E.g.: To fir

Welcome, demo-user

Release Bundles

Name

Latest Version

Created

IP_BECRCS_ENU_dockerlinux

IP_BillDataSrv_ENU_dockerlinux

IP_CCERcnP_ENU_dockerlinux

IP_CldiVDEmiCnctr_ENU_dockerlinux

IP_CustExInsightOps_ENU_dockerlinux

IP_CustExplnights_ENU_dockerlinux

IP_CXContact_ENU_dockerlinux

IP_DesAppSrv_ENU_dockerlinux

IP_EZPulse_ENU_dockerlinux

IP_GAuthSvc_ENU_dockerlinux

Set Me Up

Edit Profile

Logout

JFrog Cloud
© Copyright 2022 JFrog Ltd

https://pureengageuse1.jfrog.io/ui/admin/artifactory/user_profile

3. Enter your password and click **Unlock**.

User Profile: demo-user

To edit your user profile, please enter your password.

* Password

ⓘ

Unlock

Authentication Settings

API Key ⓘ

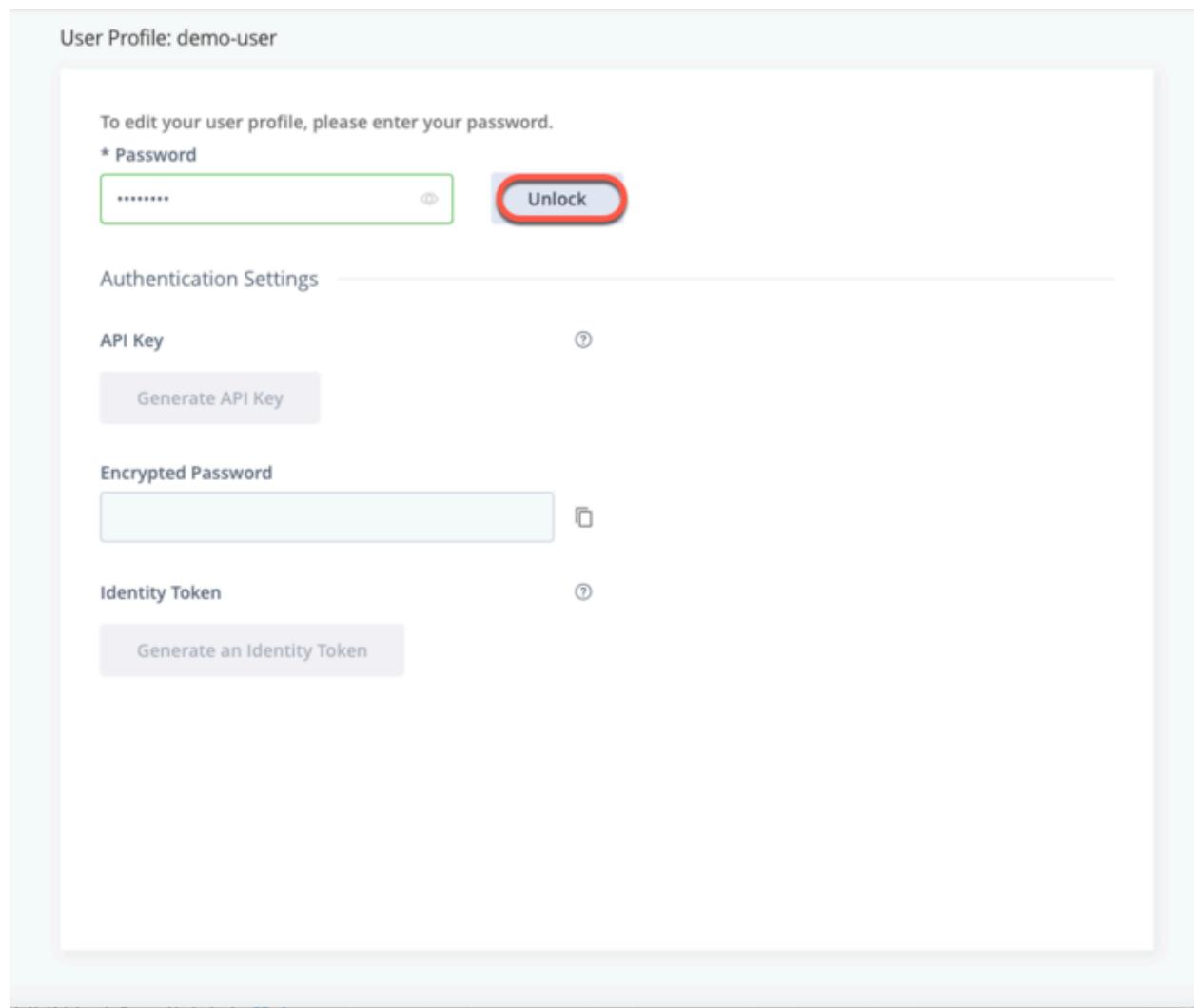
Generate API Key

Encrypted Password

ⓘ

Identity Token ⓘ

Generate an Identity Token



4. Verify your email address. Update if required and click **Save**.

User Profile: demo-user

Personal Settings

* Email Address
*****.*****@genesys.com

Change Password

* New Password

>Password Strength

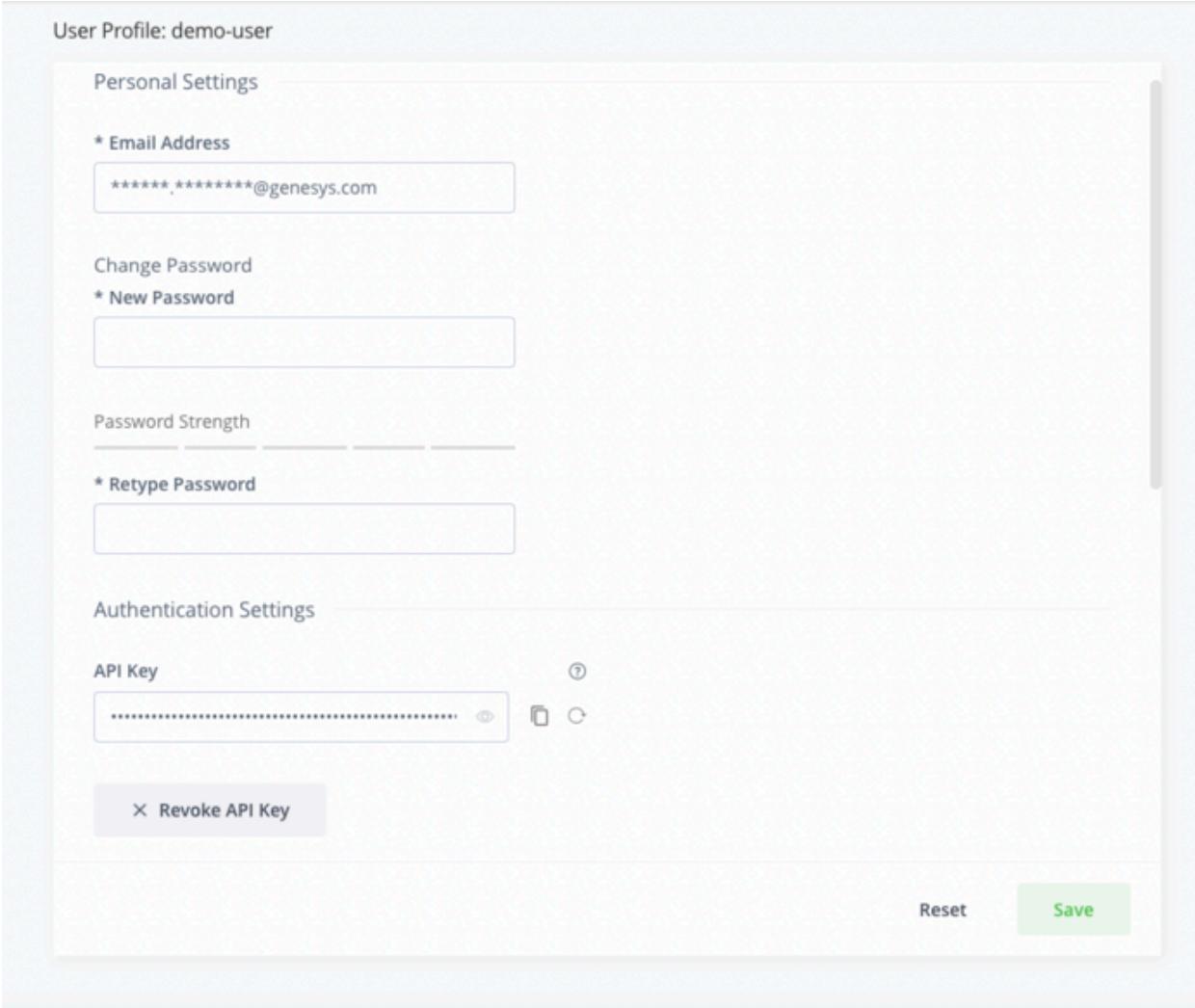
* Retype Password

Authentication Settings

API Key
..... ? Copy Reset

× Revoke API Key

Reset Save



5. Navigate to the repository for which you want update notifications.
6. Click **Actions** at the top right corner, and then click **Follow** from the drop-down.

Happily serving 14,183 artifacts

Filter repositories Clear

My Favorites Tree View:

Set Me Up Deploy

Follow Native Browser

General Properties

Info

Name: docker-multicloud-local

Package Type: Docker

Repository Path: docker-multicloud-local/

URL to file: <https://pureengageuse1.jfrog.io/artifactory/docker-...>

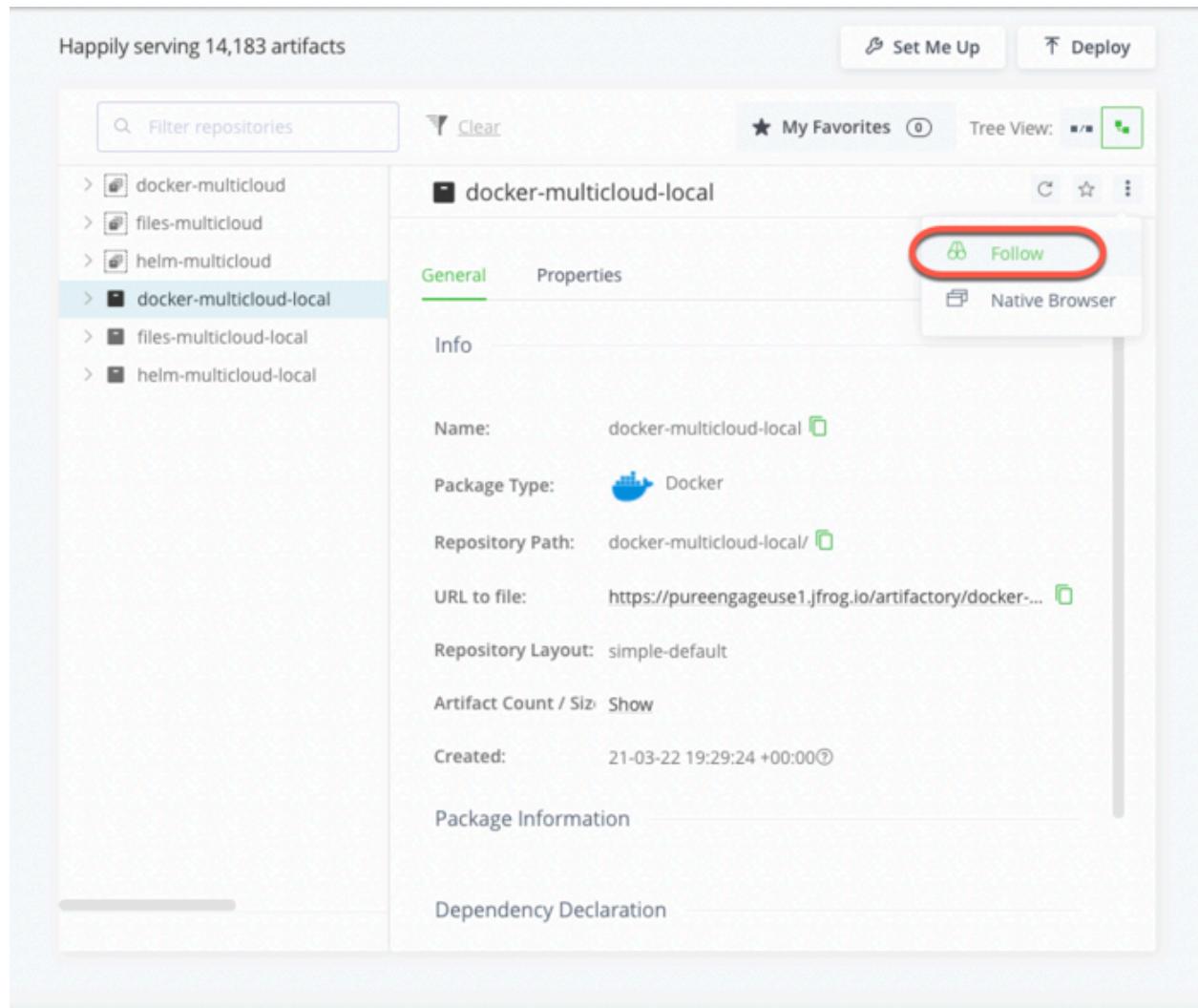
Repository Layout: simple-default

Artifact Count / Size: Show

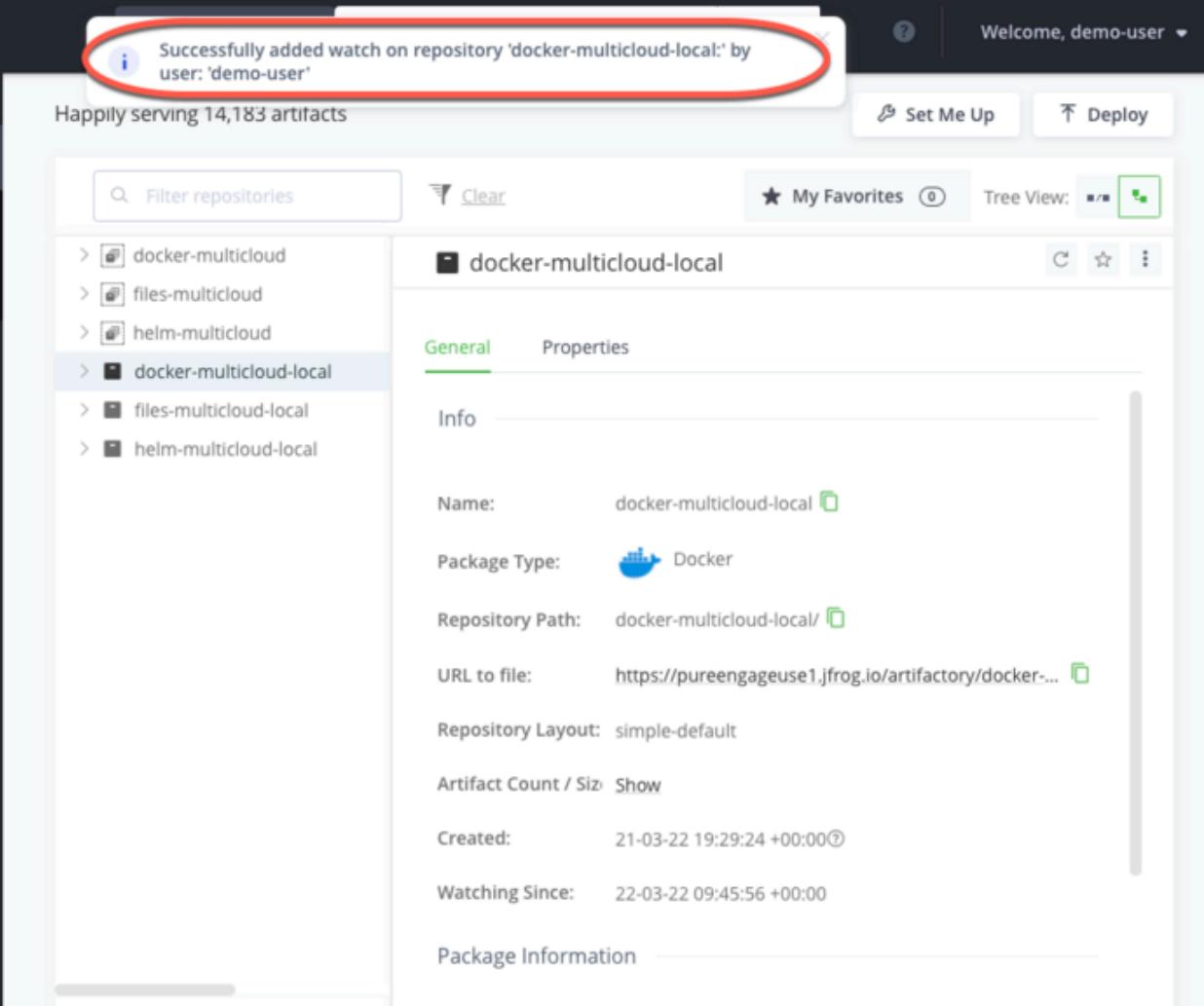
Created: 21-03-22 19:29:24 +00:00

Package Information

Dependency Declaration



A confirmation message is displayed.



The screenshot shows the JFrog Artifactory web interface. At the top, a success message is displayed: "Successfully added watch on repository 'docker-multicloud-local:' by user: 'demo-user'". The repository list on the left shows several entries, with "docker-multicloud-local" selected. The main panel shows the "General" tab for this repository, with the following details:

Setting	Value
Name:	docker-multicloud-local
Package Type:	Docker
Repository Path:	docker-multicloud-local/
URL to file:	https://pureengageuse1.jfrog.io/artifactory/docker-...
Repository Layout:	simple-default
Artifact Count / Size:	Show
Created:	21-03-22 19:29:24 +00:00
Watching Since:	22-03-22 09:45:56 +00:00

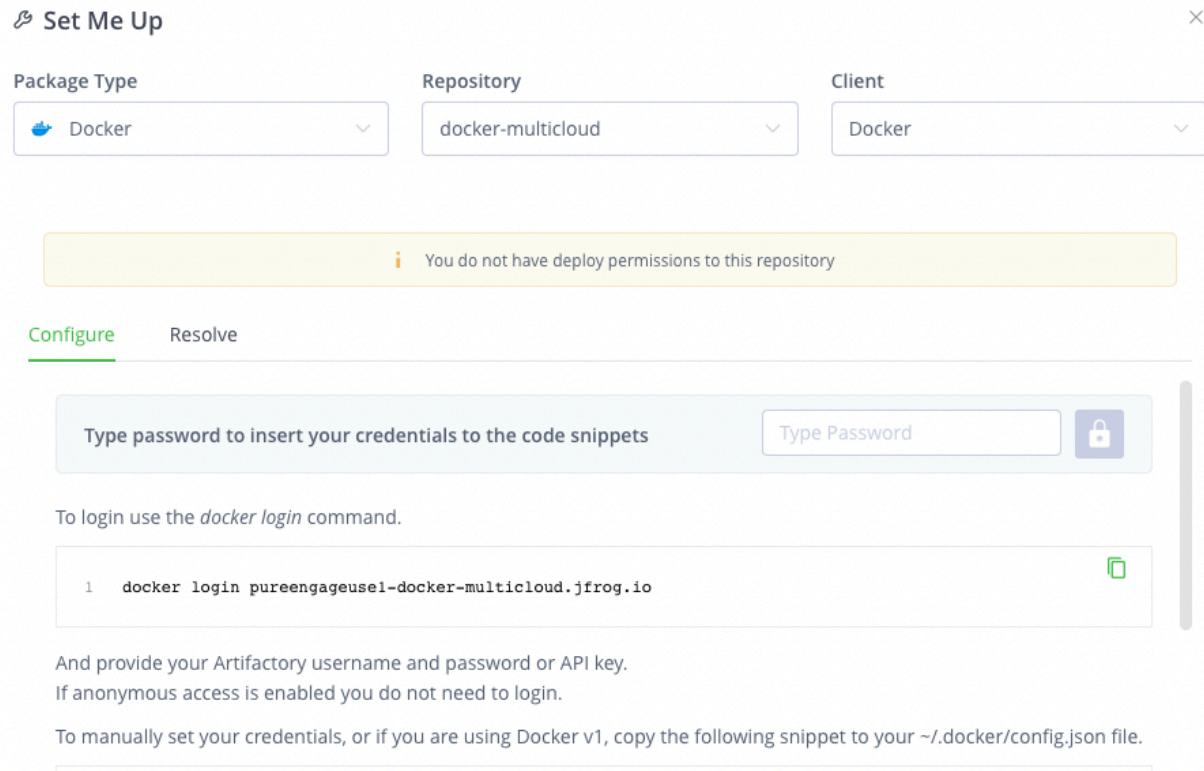
Important

Update notifications are accumulated across 1-minute intervals and sent in a single email.

Setting up automated downloads

You can integrate with external tools to automate your downloads from JFrog. The **Set Me Up** screen provides quick access to information on how to configure your different clients to work with the corresponding repositories you have created.

- Select a repository and click **Set Me Up** on the top right corner to view its **Set Me Up** screen.



The screenshot shows the 'Set Me Up' configuration interface for a Docker repository. At the top, there are three dropdown menus: 'Package Type' (set to 'Docker'), 'Repository' (set to 'docker-multicloud'), and 'Client' (set to 'Docker'). A message box in the center states, 'You do not have deploy permissions to this repository'. Below this, there are two tabs: 'Configure' (which is selected) and 'Resolve'. A text input field is labeled 'Type password to insert your credentials to the code snippets' with a placeholder 'Type Password' and a lock icon. Below this, instructions say 'To login use the `docker login` command.' followed by a code snippet:

```
1 docker login pureengageuse1-docker-multicloud.jfrog.io
```

 with a copy icon. Further down, it says 'And provide your Artifactory username and password or API key. If anonymous access is enabled you do not need to login.' and 'To manually set your credentials, or if you are using Docker v1, copy the following snippet to your `~/.docker/config.json` file.'

⚡ Set Me Up

Package Type Docker **Repository** docker-multicloud **Client** Docker

Configure **Resolve**

Tip You do not have deploy permissions to this repository

```
1  {
2    "auths": {
3      "https://pureengageuse1-docker-multicloud.jfrog.io": {
4        "auth": "<USERNAME>:<PASSWORD> (converted to base 64)",
5        "email": "youremail@email.com"
6      }
7    }
8 }
```

To enter multiple registries see the [following example](#).

Downloading using Docker CLI

1. On the **Set Me Up** screen, select **Docker** from the **Package Type** drop-down.

Set Me Up

Package Type: Docker (highlighted with a red circle)

Repository: docker-multicloud

Client: Docker

Configure (highlighted with a green underline)

Resolve

Tip: You do not have deploy permissions to this repository

```

1  {
2      "auths": {
3          "https://pureengageuse1-docker-multicloud.jfrog.io": {
4              "auth": "<USERNAME>:<PASSWORD> (converted to base 64)",
5              "email": "youremail@email.com"
6          }
7      }
8  }

```

To enter multiple registries see the [following example](#).

2. Provide the following docker login command in the **General** section as shown below:
`docker login pureengageuse1-docker-multicloud.jfrog.io`

3. Provide your Artifactory username and password or the API key in the provided input field.

Important

You can set up your API key from the **Edit Profile** option.

4. To manually set your credentials, or if you are using Docker v1, copy the following snippet to your **~/.docker/config.json** file:

```
{
  "auths": {
    "https://pureengageuse1-docker-multicloud.jfrog.io": {
      "auth": "<USERNAME>:<PASSWORD> (converted to base 64)", "email": "youremail@email.com"
    }
  }
}
```

5. To pull an image use the docker pull command specifying the docker image and tag names:
`docker pull pureengageuse1-docker-multicloud.jfrog.io/`

Important

Tagging allows you to group related container images together.

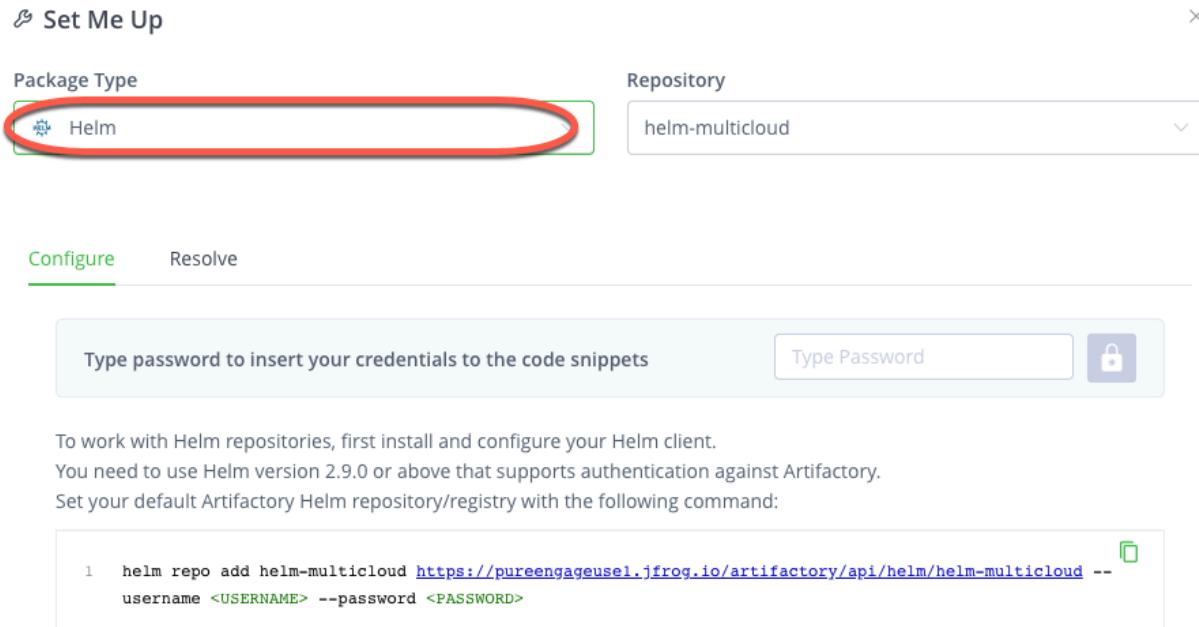
Downloading using the Helm CLI

To work with Helm repositories, you must have a Helm client installed and configured before you perform the following steps:

Important

You must use Helm version 2.9.0 or a higher version that supports authentication against Artifactory.

1. On the **SET ME UP** screen, select **Helm** from the **Package Type** drop-down.



The screenshot shows the 'Set Me Up' configuration screen. The 'Package Type' dropdown is set to 'Helm', which is highlighted with a red oval. The 'Repository' dropdown is set to 'helm-multicloud'. Below the dropdowns are 'Configure' and 'Resolve' buttons. A 'Type Password' input field with a lock icon is shown. Below the input field, instructions for setting up Helm are provided, including a command line snippet.

Type password to insert your credentials to the code snippets Type Password Lock

To work with Helm repositories, first install and configure your Helm client. You need to use Helm version 2.9.0 or above that supports authentication against Artifactory. Set your default Artifactory Helm repository/registry with the following command:

```
1 helm repo add helm-multicloud https://pureengageuse1.jfrog.io/artifactory/api/helm/helm-multicloud --username <USERNAME> --password <PASSWORD>
```

2. In the **General** section, set up your default Artifactory Helm repository/registry with the following command:
`helm repo add helm-multicloud https://pureengageuse1.jfrog.io/artifactory/helm-multicloud --username --password`

3. In the **Resolve** section, provide the following commands to install a Helm Chart from the selected repository using your Helm command line client:
`helm repo update
helm install helm-multicloud/[chartName]`

Downloading using cURL

You can also download a package from the Edge Artifactory by accessing its API through a cURL command.

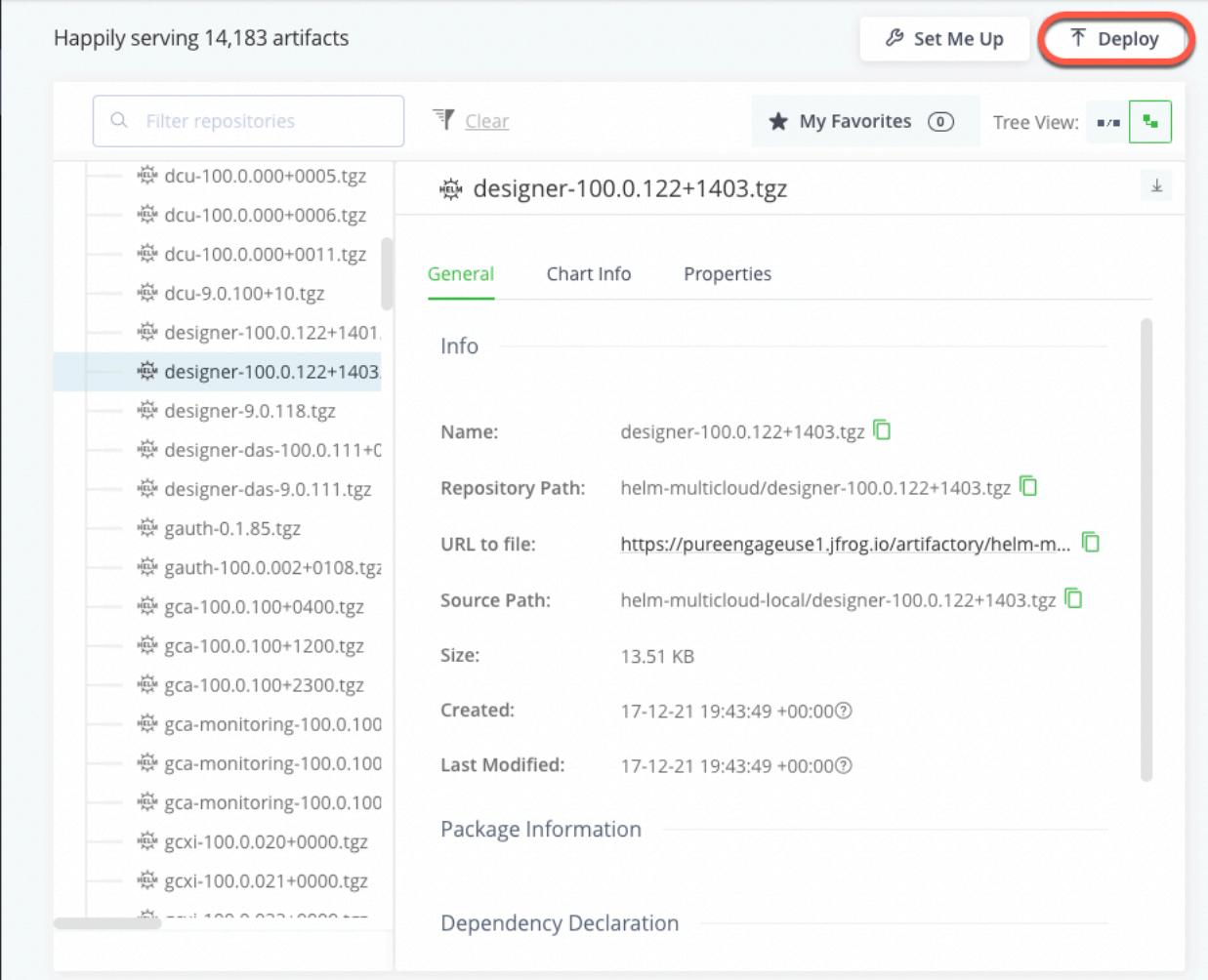
For example,

```
curl -u: -O "https://pureengageuse1.jfrog.io/artifactory/helm-multicloud/  
cxcontact-022.03.121.tgz".
```

Downloading manually

JFrog Artifactory also supports manual downloads if you do not want to set up a CD pipeline.

1. Select the required artifact using the *Tree* browsing method or the *Simple* browsing method.
2. Click the **Deploy** option on the top right corner of the screen.



The screenshot shows the JFrog Artifactory interface. At the top, a message says "Happily serving 14,183 artifacts". There are buttons for "Set Me Up" and "Deploy" (which is circled in red). Below this is a search bar with "Filter repositories" and a "Clear" button. To the right are "My Favorites" and "Tree View" buttons. The main area shows a list of artifacts on the left and a detailed view of one on the right. The artifact details include:

General		Chart Info	Properties
Info			
Name:	designer-100.0.122+1403.tgz		
Repository Path:	helm-multicloud/designer-100.0.122+1403.tgz		
URL to file:	https://pureengageuse1.jfrog.io/artifactory/helm-m...		
Source Path:	helm-multicloud-local/designer-100.0.122+1403.tgz		
Size:	13.51 KB		
Created:	17-12-21 19:43:49 +00:00		
Last Modified:	17-12-21 19:43:49 +00:00		
Package Information			
Dependency Declaration			

For more information on browsing through the artifacts in the Artifact Repository Browser, refer to the [Browsing Artifacts](#) topic on the JFrog documentation site.

Additional reading material

- JFrog Artifactory Edge (an *Edge node*) is an edition of JFrog Artifactory with features customized to serve the primary purpose of distributing software to a runtime system such as a data center, a point-of-sale, or even a mobile device.
For more information, refer to the [JFrog Artifactory Edge](#) topic on the JFrog documentation site.
- Local repositories are physical, locally-managed repositories into which you can deploy artifacts. Whereas, a virtual repository (or repository group) aggregates several repositories with the same package type under a common URL.
For more information, refer to the [Repository Management](#) topic on the JFrog documentation site.