

HelmPublish

Overview

The **HelmPublish** step publishes a Helm chart and associated build info from a location in a Git repo to a Helm repository in Artifactory.

This step uses the `helm package` command to publish the collection of Helm files in the [GitRepo](#) to Artifactory.

The step can optionally publish build info with the Helm chart to Artifactory using the `autoPublishBuildInfo` flag.

Once the Helm chart is published to Artifactory, it can be deployed to a Kubernetes cluster using [HelmDeploy](#).

Page Contents

- [Overview](#)
- [YAML Schema](#)
- [Tags](#)
 - [name](#)
 - [type](#)
 - [configuration](#)
 - [execution](#)
- [Examples](#)
 - [Publish to Artifactory using Helm 3](#)
 - [Using lint and values.yaml](#)
 - [Publish Build Info](#)
 - [Publish to Artifactory using Helm 2](#)
- [How it Works](#)

YAML Schema

The YAML schema for HelmPublish native step is as follows:

HelmPublish

```
pipelines:
  - name: <string>
    steps:
      - name: <string>
        type: HelmPublish
        configuration:
          #inherits all the tags from bash; https://www.jfrog.com/confluence/display/JFROG/Bash
          helmVersion: <2 | 3> # optional, defaults to 2
          chartPath: <path string>
          flags: <option string> # optional
          lint: <true | false> # optional, default false
          lintFlags: <string> # optional
          valueFilePaths: # optional
            - <path and filename>
          autoPublishBuildInfo: <true | false> # default false
          inputResources:
            - name: <GitRepo resource> # required
          outputResources:
            - name: <HelmChart resource> # required
            - name: <BuildInfo resource> # optional

        execution:
          onStart:
            - echo "Preparing for work..."
          onSuccess:
            - echo "Job well done!"
          onFailure:
            - echo "uh oh, something went wrong"
          onComplete: #always
            - echo "Cleaning up some stuff"
```

Tags

name

An alphanumeric string (underscores are permitted) that identifies the step.

type

Must be `HelmPublish` for this step type.

configuration

Specifies all configuration selections for the step's execution environment. This step inherits the [Bash/PowerShell](#) step configuration tags, including these pertinent tags:

Tag	Description of usage	Required /Optional
<code>inputResources</code>	Must specify a GitRepo resource that contains the Helm chart. The step will package the chart into a versioned chart archive file.	Required
<code>outputResources</code>	Must specify a HelmChart resource to update. The <code>version</code> property of the HelmChart is automatically set to the new chart version. Must specify a BuildInfo resource when <code>autoPublishBuildInfo</code> is set to true. If <code>JFROG_CLI_BUILD_NAME</code> or <code>JFROG_CLI_BUILD_NUMBER</code> is set as an environment variable for the pipeline or the step, that name and/or number is used for the output BuildInfo. Otherwise, the default <code>buildName</code> and <code>buildNumber</code> are <code>\$pipeline_name</code> and <code>\$run_number</code> .	Required May be required

In addition, these tags can be defined to support the step's native operation:



Tags derived from Bash

All native steps derive from the [Bash](#) step. This means that all steps share the same base set of tags from Bash, while native steps have their own additional tags as well that support the step's particular function. So it's important to be familiar with the [Bash](#) step definition, since it's the core of the definition of all other steps.

Tag	Description of usage	Required /Optional
<code>helmVersion</code>	A number representing the major version of Helm to use. Can be 2 or 3. Defaults to 2.	Optional
<code>chartPath</code>	The path to the Helm chart in the GitRepo	Required
<code>flags</code>	Command line options to pass to the helm package command.	Optional
<code>lint</code>	When true, performs a lint test to verify that the chart is well formed. Defaults to false.	Optional
<code>lintFlags</code>	Options string to pass to the <code>helm lint</code> command.	Optional
<code>valueFilePaths</code>	Specifies values YAML file(s) that will be used in the lint command <code>helm lint</code> command. All environment variable references in the specified file(s) will be automatically replaced with values of matching runtime environment variables.	Optional
<code>autoPublishBuildInfo</code>	When true, publishes build info with the Helm chart. Default is false.	Optional

execution

Declares collections of shell command sequences to perform for pre- and post-execution phases:

Tag	Description of usage	Required/Optional
<code>onStart</code>	Commands to execute in advance of the native operation	Optional
<code>onSuccess</code>	Commands to execute on successful completion	Optional
<code>onFailure</code>	Commands to execute on failed completion	Optional

onComplete	Commands to execute on any completion	Optional
------------	---------------------------------------	----------

The actions performed for the `onExecute` phase are inherent to this step type and may not be overridden.

Examples

The following examples show how to configure a `HelmPublish` step.

Publish to Artifactory using Helm 3

A `HelmPublish` step using Helm 3 to publish to Artifactory.

HelmPublish
<pre>pipelines: - name: helmPublishPipeline steps: - name: helmPublishStep type: HelmPublish configuration: helmVersion: 3 chartPath: "./myChart" inputResources: - name: gitRepoResource outputResources: - name: helmChartResource</pre>

Using lint and values.yaml

A `HelmPublish` step with lint, using a `values.yaml` file, added before publishing the Helm chart.

HelmPublish
<pre>pipelines: - name: helmPublishPipeline steps: - name: helmPublishStep type: HelmPublish configuration: helmVersion: 3 chartPath: "./myChart" lint: true lintFlags: "--strict" valueFilePaths: - values.yaml inputResources: - name: gitRepoResource outputResources: - name: helmChartResource</pre>

Publish Build Info

A `HelmPublish` step that publishes build info.

HelmPublish

```
pipelines:  
- name: helmPublishPipeline  
  steps:  
  - name: helmPublishStep  
    type: HelmPublish  
    configuration:  
      helmVersion: 3  
      chartPath: "./myChart"  
      autoPublishBuildInfo: true  
      inputResources:  
        - name: gitRepoResource  
      outputResources:  
        - name: helmChartResource  
        - name: buildInfoResource
```

Publish to Artifactory using Helm 2

A HelmPublish step using Helm 2 to publish to Artifactory.

HelmPublish

```
pipelines:  
- name: helmPublishPipeline  
  steps:  
  - name: helmPublishStep  
    type: HelmPublish  
    configuration:  
      helmVersion: 2  
      chartPath: "./myChart"  
      inputResources:  
        - name: gitRepoResource  
      outputResources:  
        - name: helmChartResource
```

How it Works

When you use the **HelmPublish** native step in a pipeline, it performs the following functions in the background:

- jfrog rt config (to configure the JFrog CLI with the Artifactory credentials)
- jfrog rt use (to set the current default Artifactory configuration)
- helm init --client-only (if the Helm version is 2)
- replace_envs (if lint is true and there are valueFilePaths, to replace variable placeholders)
- helm lint (if lint is true)
- helm package
- jfrog rt upload (upload the Helm chart)
- write_output (to update the version in the output HelmChart resource)
- jfrog rt build-collect-env (if autoPublishBuildInfo is true, collect environment variables)
- jfrog rt build-publish (if autoPublishBuildInfo is true, publish the build info)
- write_output (if autoPublishBuildInfo is true, to update the version in the output BuildInfo resource)