ReleaseBundle

Overview

A ReleaseBundle resource specifies a set of artifacts in Artifactory that are distributed to Artifactory Edge nodes as a JFrog Distribution Release Bundle.

This resource can be used with CreateReleaseBundle or DistributeReleaseBundle. This resource can be used to represent both signed and unsigned Release Bundle.

Page Contents

YAML Schema

```
resources:
  - name:                 <string>
    type:                 ReleaseBundle
    configuration:
      sourceDistribution:   distributionIntegration
      name:                                 <string>
      version:                                 <string>
      isSigned:                          <boolean>
```

Tags

name

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

type

Must be ReleaseBundle for this resource type.

configuration

Specifies all configuration selections for the resource.

Tag	Description	Required /Optional
sourceDistribution	The name of a Distribution Integration	Required
name	Name of the release bundle	Required
version	Version number of the release bundle	Required
isSigned	When **true**, the release bundle is signed. Defaults to **false**.	Optional

Environment Variables

Whenever ReleaseBundle is used in a step, a set of environment variables is automatically made available that you can use in your step.

Environment Variable	Description
res_<resource_name>_name	Name of the resource

res_<resource_name>_sourceDistribution	Name of the Distribution integration
res_<resource_name>_name	Name of the release bundle
res_<resource_name>_version	Version number of the release bundle
res_<resource_name>_isSigned	Indicates whether the release bundle is signed

Examples

- This examples requires an Artifactory Integration and a Distribution Integration.
- The Pipelines DSL for this example is available in this repository in the JFrog GitHub account.

```
resources:
  # Build info of first build to bundle
  - name: gosvc_promoted_build_info
    type: BuildInfo
    configuration:
      sourceArtifactory: myArtifactory
      buildName: svc_build
      buildNumber: 1

  # Build info of second build to bundle
  - name: appl_promoted_build_info
    type: BuildInfo
    configuration:
      sourceArtifactory: demoArt
      buildName: backend_build
      buildNumber: 1

  # Release bundle
  - name: release_bundle
    type: ReleaseBundle
    configuration:
      sourceDistribution: myDist
      name: demo_rb
      version: v1.0.0

  # Signed version of the same release bundle
  - name: signed_bundle
    type: ReleaseBundle
    configuration:
      sourceDistribution: myDist
      name: demo_rb
      version: v1.0.0

  # Distribution rules
  - name: distribution_rules
    type: DistributionRule
    configuration:
      sourceDistribution: myDist
      serviceName: "*"
      siteName: "*"
      cityName: "*"
      countryCodes:
        - "CN"
        - "GB"

pipelines:
  - name: demo_release_mgmt
    steps:
      - name: bundle
        type: CreateReleaseBundle
        configuration:
          releaseBundleName: demo_rb
          releaseBundleVersion: v1.0.${run_number}
          dryRun: false
          sign: false
          description: "some random test description"
          inputResources:
            - name: gosvc_promoted_build_info
              trigger: true
            - name: appl_promoted_build_info
              trigger: true
          outputResources:
            - name: release_bundle
          releaseNotes:
            syntax: markdown
            content: |
              ## Heading
                * Bullet
                * Points

      - name: sign
        type: SignReleaseBundle
        configuration:
          inputResources:
            - name: release_bundle
          outputResources:
            - name: signed_bundle

      - name: distribute
        type: DistributeReleaseBundle
        configuration:
          dryRun: false
          inputResources:
            - name: signed_bundle
            - name: distribution_rules
```