Validation Results: NetApp HCI for Red Hat OpenShift on RHV

HCI

NetApp
July 08, 2020

Table of Contents

Validation Results: NetApp HCI for Red Hat OpenShift on RHV

This section provides the steps to deploy a continuous integration/continuous delivery or deployment (CI/CD) pipeline with Jenkins in order to validate the operation of the solution.

Create the Resources Required for Jenkins Deployment

To create the resources required for deploying the Jenkins application, complete the following steps:

1. Create a new project named Jenkins.

Create Project

Name *

Jenkins

Display Name

Description

Cancel   Create

2. In this example, we deployed Jenkins with persistent storage. To support the Jenkins build, create the PVC. Navigate to Storage > Persistent Volume Claims and click Create Persistent Volume Claim. Select the storage class that was created, make sure that the Persistent Volume Claim Name is jenkins, select the appropriate size and access mode, and then click Create.

Project: jenkins ▼

Create Persistent Volume Claim

Storage Class

SC basic ▼

Storage class for the new claim.

Persistent Volume Claim Name *

jenkins

A unique name for the storage claim within the project.

Access Mode *

◉ Single User (RWO) ○ Shared Access (RWX) ○ Read Only (ROX)

Permissions to the mounted drive.

Size *

100 GiB ▼

Desired storage capacity.

☐ Use label selectors to request storage

Use label selectors to define how storage is created.

Create   Cancel

Deploy Jenkins with Persistent Storage

To deploy Jenkins with persistent storage, complete the following steps:

1. In the upper left corner, change the role from Administrator to Developer. Click +Add and select From Catalog. In the Filter by Keyword bar, search jenkins. Select Jenkins Service, with Persistent Storage.

Project: jenkins ▼

Developer Catalog

Add shared apps, services, or source-to-image builders to your project from the Developer Catalog. Cluster admins can install additional apps which will show up here automatical

All Items

Languages

Databases

Middleware

CI/CD

Other

Type
- ☑ Operator Backed (0)
- ☐ Helm Charts (0)
- ☑ Builder Image (0)
- ☑ Template (4)
- ☐ Service Class (0)

All Items

jenkins

Group By: None ▼

	Template
Jenkins	
provided by Red Hat, Inc.	
Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...	

	Template
Jenkins	
provided by Red Hat, Inc.	
Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...	

	Template
Jenkins (Ephemeral)	
provided by Red Hat, Inc.	
Jenkins service, without persistent storage. WARNING: Any data stored will be lost upon...	

	Template
Jenkins (Ephemeral)	
provided by Red Hat, Inc.	
Jenkins service, without persistent storage. WARNING:	

2. Click Instantiate Template.

Jenkins
Provided by Red Hat, Inc.

✕

Instantiate Template

Provider
Red Hat, Inc.

Support
Get support ☑

Created At
🌐 May 26, 3:58 am

Description

Jenkins service, with persistent storage.

NOTE: You must have persistent volumes available in your cluster to use this template.

Documentation

https://docs.okd.io/latest/using_images/other_images/jenkins.html ☑

3. By default, the details for the Jenkins application are populated. Based on your requirements, modify the parameters, and click Create. This process creates all the required resources for

supporting Jenkins on OpenShift.

Instantiate Template

Namespace *

> PR jenkins ▾

Jenkins Service Name

> jenkins

The name of the OpenShift Service exposed for the Jenkins container.

Jenkins JNLP Service Name

> jenkins-jnlp

The name of the service used for master/slave communication.

Enable OAuth in Jenkins

> true

Whether to enable OAuth OpenShift integration. If false, the static account 'admin' will be initialized with the password 'password'.

Memory Limit

> 1Gi

Maximum amount of memory the container can use.

Volume Capacity *

> 50Gi

Volume space available for data, e.g. 512Mi, 2Gi.

Jenkins ImageStream Namespace

> openshift

The OpenShift Namespace where the Jenkins ImageStream resides.

Disable memory intensive administrative monitors

> false

Whether to perform memory intensive, possibly slow, synchronization with the Jenkins Update Center on start. If true, the Jenkins core update monitor and site warnings monitor are disabled.

Jenkins ImageStreamTag

> jenkins:2

Name of the ImageStreamTag to be used for the Jenkins image.

Fatal Error Log File

> false

When a fatal error occurs, an error log is created with information and the state obtained at the time of the fatal error.

Allows use of Jenkins Update Center repository with invalid SSL certificate

> false

Whether to allow use of a Jenkins Update Center that uses invalid certificate (self-signed, unknown CA). If any value other than 'false', certificate check is bypassed. By default, certificate check is enforced.

[Create] [Cancel]

Jenkins
INSTANT-APP JENKINS
View documentation ⧉ Get support ⧉

Jenkins service, with persistent storage.

NOTE: You must have persistent volumes available in your cluster to use this template.

The following resources will be created:

- DeploymentConfig
- PersistentVolumeClaim
- RoleBinding
- Route
- Service
- ServiceAccount

4. The Jenkins pods take approximately 10–12 minutes to enter the Ready state.

Pods

Create Pod   Filter by name...

| 1 Running | 0 Pending | 0 Terminating | 0 CrashLoopBackOff | 1 Completed | 0 Failed | 0 Unknown |

Select all filters   1 of 2 Items

Name ↑	Namespace ↕	Status ↕	Ready ↕	Owner ↕	Memory ↕	CPU ↕	
P jenkins-1-c77n9	NS jenkins	↻ Running	1/1	RC jenkins-1	-	0.004 cores	⋮

5. After the pods are instantiated, navigate to Networking > Routes. To open the Jenkins webpage, click the URL provided for the jenkins route.

Project: jenkins ▼

Routes

Create Route   Filter by name...

| 1 Accepted | 0 Rejected | 0 Pending | Select all filters | 1 Item

Name ↓	Namespace ↕	Status	Location ↕	Service ↕	
RT jenkins	NS jenkins	✓ Accepted	https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com ☑	S jenkins	⋮

6. Because the OpenShift OAuth was used while creating the Jenkins app, click Log in with OpenShift.
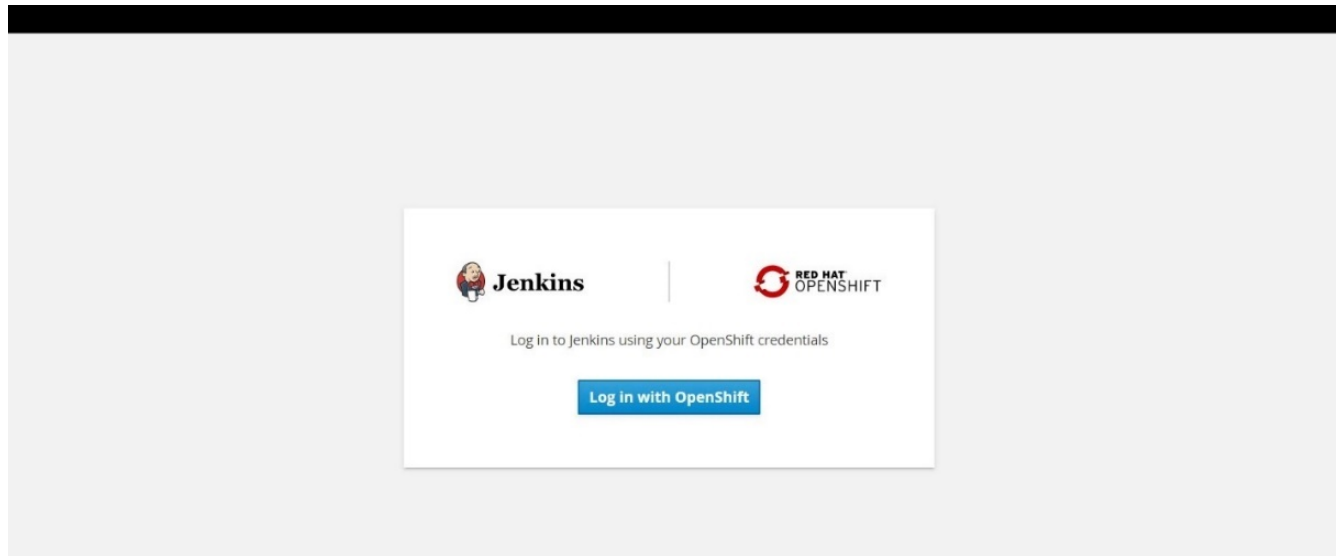
7. Authorize jenkins service-account to access the OpenShift users.

8. The Jenkins welcome page is displayed. Because we are using a Maven build, complete the Maven installation first. Navigate to Manage Jenkins > Global Tool Configuration, then in the Maven subhead, click Add Maven. Enter the name of your choice and make sure that the Install Automatically option is selected. Click Save.

9. You can now create a pipeline to demonstrate the CI/CD workflow. On the home page, click Create New Jobs or New Item from the left- hand menu.

10. On the Create Item page, enter the name of your choice, select Pipeline, and click Ok.

Enter an item name

sample-demo

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Bitbucket Team/Project
Scans a Bitbucket Cloud Team (or Bitbucket Server Project) for all repositories matching some defined markers.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

GitHub Organization
Scans a GitHub organization (or user account) for all repositories matching some defined markers.

OK

Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

11. Select the Pipeline tab. From the Try Sample Pipeline drop- down menu, select Github + Maven. The code is automatically populated. Click Save.

General   Build Triggers   Advanced Project Options   **Pipeline**

Advanced...

Pipeline

Definition   Pipeline script ▼

Script

```
1 ▼ node {
2        def mvnHome
3 ▼      stage('Preparation') { // for display purposes
4            // Get some code from a GitHub repository
5            git 'https://github.com/jglick/simple-maven-project-with-tests.git'
6            // Get the Maven tool.
7            // ** NOTE: This 'M3' Maven tool must be configured
8            // **       in the global configuration.
9            mvnHome = tool 'M3'
10       }
11 ▼     stage('Build') {
12           // Run the maven build
13 ▼         withEnv(["MVN_HOME=$mvnHome"]) {
14 ▼             if (isUnix()) {
15                   sh '"$MVN_HOME/bin/mvn" -Dmaven.test.failure.ignore clean package'
16 ▼             } else {
17                   bat(/"%MVN_HOME%\bin\mvn" -Dmaven.test.failure.ignore clean package/)
```

GitHub + Maven ▼

☑ Use Groovy Sandbox

Pipeline Syntax

Save   Apply

12. Click Build Now to trigger the development through the preparation, build, and testing phase. It can take several minutes to complete the whole build process and display the results of the build.

Jenkins

↑ Back to Dashboard
🔍 **Status**
📝 Changes
▶ Build Now
🚫 Delete Pipeline
⚙ Configure
🔍 Full Stage View
🔵 Open Blue Ocean
📝 Rename
❓ Pipeline Syntax

Pipeline sample-demo

Last Successful Artifacts
📄 simple-maven-project-with-tests-1.0-SNAPSHOT.jar   1.71 KB view

Recent Changes

Stage View

	Preparation	Build	Results
Average stage times: (Average full run time: ~7s)	2s	4s	69ms
#1 May 27 08:53 No Changes ⊙	2s	4s	69ms

📋 Latest Test Result (no failures)

Permalinks

- Last build (#1), 1 min 23 sec ago
- Last stable build (#1), 1 min 23 sec ago
- Last successful build (#1), 1 min 23 sec ago
- Last completed build (#1), 1 min 23 sec ago

⚙ **Build History**   trend ▬

find   x

🔵 **#1**   May 27, 2020 3:53 PM

🔶 Atom feed for all 🔶 Atom feed for failures

13. Whenever there are any code changes, the pipeline can be rebuilt to patch the new version of software enabling continuous integration and continuous delivery. Click Recent Changes to track the changes from the previous version.

Jenkins

🔼 Back to Dashboard

🔍 **Status**

📝 Changes

🔵 Build Now

🚫 Delete Pipeline

⚙ Configure

🔍 Full Stage View

🔵 Open Blue Ocean

📝 Rename

❓ Pipeline Syntax

☀ **Build History**　　trend ━

find	x

🌐 **#2**　　May 27, 2020 3:56 PM

🌐 **#1**　　May 27, 2020 3:53 PM

📶 Atom feed for all 📶 Atom feed for failures

Pipeline sample-demo

📦 **Last Successful Artifacts**
　　📄 simple-maven-project-with-tests-1.0-SNAPSHOT.jar　　1.71 KB view

📝 **Recent Changes**

Stage View

	Preparation	Build	Results
Average stage times: (Average full run time: ~6s)	2s	4s	86ms
#2 May 27 08:56 No Changes ⊕	1s	4s	104ms
#1 May 27 08:53 No Changes ⊕	2s	4s	69ms
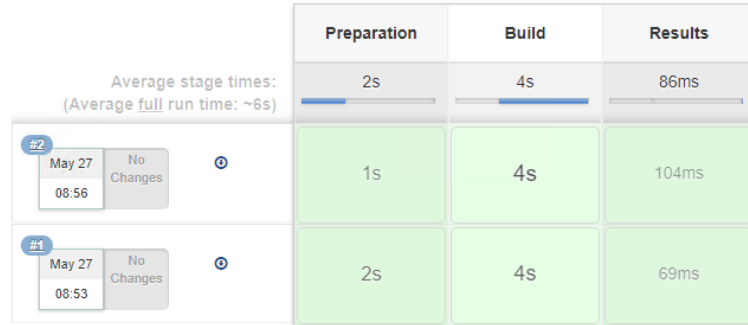
📋 **Latest Test Result** (no failures)

Permalinks

- Last build (#2), 19 sec ago
- Last stable build (#2), 19 sec ago
- Last successful build (#2), 19 sec ago
- Last completed build (#2), 19 sec ago