OpenShift Design Document

Table of Contents

[Document Version History 3](#_Toc81767288)

[About this document 4](#_Toc81767289)

[1. Objective / Scope 5](#_Toc81767290)

[2. Deliverables: 5](#_Toc81767291)

[3. Deployment Architecture 6](#_Toc81767292)

[4. Verify Prerequisites: 7](#_Toc81767293)

[5. Image Registry Setup: 8](#_Toc81767294)

[6. OpenShift Installation 10](#_Toc81767295)

[7. OpenShift Verification 11](#_Toc81767296)

[8. OpenShift HTPasswd configuration 12](#_Toc81767297)

[9. OpenShift LDAP configuration 13](#_Toc81767298)

[10. Configure OpenShift Container Storage (OCS) 14](#_Toc81767299)

[Installing Red Hat OpenShift Container Storage Operator 15](#_Toc81767300)

[Creating OpenShift Container Storage cluster on bare metal 15](#_Toc81767301)

# Document Version History

|  |  |
| --- | --- |
| Date | Changes |
| 3/Sep/2021 | * Initial Draft Definition |
| 6/Sep/2021 | * Internal Review |

# About this document

This document covers the Red Hat OpenShift v4.7.23 setup with configurations that require to do on G42 cloud.

# Objective / Scope

**Introduction:**

Objective of the document covers the complete setup instructions and provide the details of the steps to follow the installation.

OpenShift v4.7.23 Setup.

1. OpenShift Cluster v4.7.23 Setup on G42 cloud with 3 masters & 3 workers.

2. Configure HTpassword with OpenShift

3. Configure LDAP & Role assignment with OpenShift

4. Configure OCS Operator and configuration with OpenShift

# Deliverables:

We provide the below documents post the infra setup & configuration.

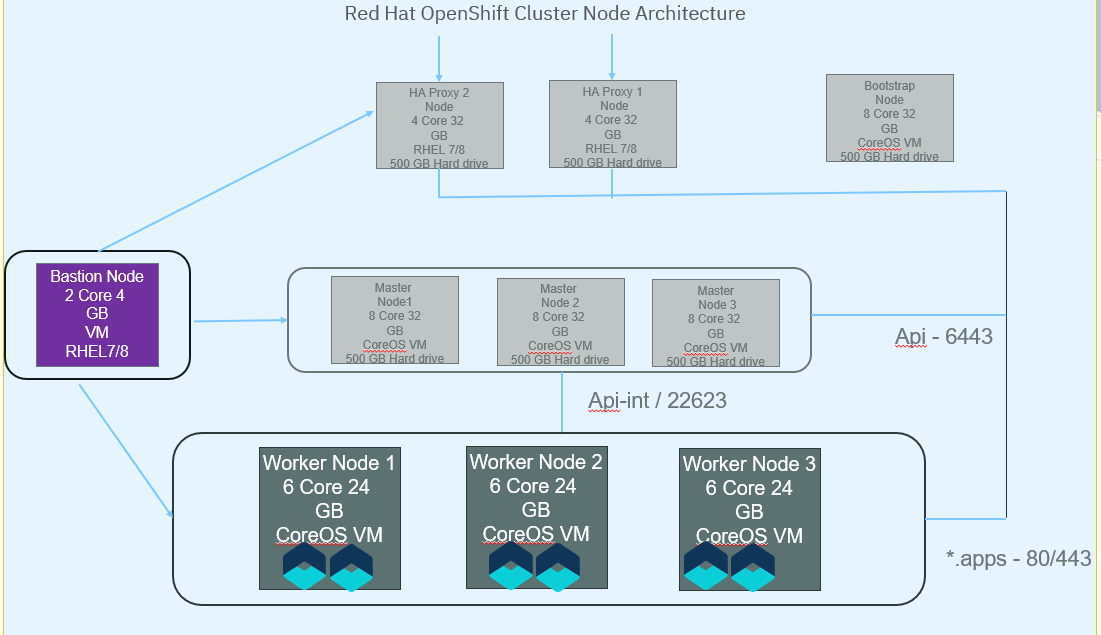
1. OpenShift Cluster v4.7.23 Setup document

2. OpenShift HTpassword setup document with yaml file

3. OpenShift LDAP & Role assignment document with yaml file

4. OCS configuration document with steps.

# Deployment Architecture



Bootstrap Node – 1

Master Nodes – 3

Worker Nodes – 3

Bastion Node – 1

Registry Node – 1

# Verify Prerequisites:

1. Verify the FQDN for nodes should resolve to the correct IP configured on DNS

bootstrap.clustername.domainname.com

master0.clustername.domainname.com

master1.clustername.domainname.com

master2.clustername.domainname.com

worker0.clustername.domainname.com

worker1.clustername.domainname.com

worker2.clustername.domainname.com

registry.clustername.domainname.com

basition.clustername.domainname.com

Load Balancer is assumed to be the existing organization LB.

2. Verify DNS name which should resolve the Load balancer IPs.

api.clustername.domainname.com

api-int.clustername.domainname.com

\*.apps.clustername.domainname.com

3. Check all the required external domains are whitelisted and accessible

mirror.openshift.com

\*.cloudfront.net

\*.apps.<cluster\_name>.<base\_domain>

quay-registry

api.openshift.com

art-rhcos-ci.

api.openshift.com

cloud.redhat.com/openshift

registry.access.redhat.com

4. Ensure the ports are allowed in the firewall

6443 - api

22623 - api-int

80/443 - \*.apps

Additional ports mentioned on the rehat setup guide.

5. Get the pull-secret file from custer.

6. Access the bastion node & make sure you get root access.

7. Download CLI and OpenShift Installer utility and kept on Bastian Node.

8. Download the ISO and metal-core image files and kept on Bastion node.

9. Install JQ, Podman & apache server on Bastian node.

# Image Registry Setup:

**Install cli**

Place the oc binary on PATH.

**Install podman**

yum -y install podman httpd-tools

**Create Directory**

mkdir -p /opt/registry/{auth,certs,data}

**Go to the directory:**

cd /opt/registry/certs

**Create certificates for the mirror host:**

openssl req -newkey rsa:4096 -nodes -sha256 -keyoutdomain.key -x509 -days 365 -out domain.crt

**Create Username and password for registry:**

htpasswd -bBc /opt/registry/auth/htpasswd<username><password>

**Create mirror registry:**

podman run --name mirror-registry -p 5000:5000 -v /opt/registry/data:/var/lib/registry:z -v /opt/registry/auth:/auth:z -e "REGISTRY\_AUTH=htpasswd" -e "REGISTRY\_AUTH\_HTPASSWD\_REALM=Registry Realm" -e REGISTRY\_AUTH\_HTPASSWD\_PATH=/auth/htpasswd -v /opt/registry/certs:/certs:z -e REGISTRY\_HTTP\_TLS\_CERTIFICATE=/certs/domain.crt -e REGISTRY\_HTTP\_TLS\_KEY=/certs/domain.key -e REGISTRY\_COMPATIBILITY\_SCHEMA1\_ENABLED=true -d docker.io/library/registry:2

**Pull Images:**

**Mirror the images to a directory on the removable media**

ocadm release mirror -a ${LOCAL\_SECRET\_JSON} --to-dir=${REMOVABLE\_MEDIA\_PATH}/mirror quay.io/${PRODUCT\_REPO}/${RELEASE\_NAME}:${OCP\_RELEASE}-${ARCHITECTURE}

**Upload the images to the local container registry**

GODEBUG=x509ignoreCN=0 oc image mirror -a ${LOCAL\_SECRET\_JSON} --from-dir=${REMOVABLE\_MEDIA\_PATH}/mirror "file://openshift/release:${OCP\_RELEASE}\*" ${LOCAL\_REGISTRY}/${LOCAL\_REPOSITORY}

**OpenShift Container Platform that you selected; you must extract the installation program from the mirrored content**

GODEBUG=x509ignoreCN=0 ocadm release extract -a ${LOCAL\_SECRET\_JSON} --command=openshift-install "${LOCAL\_REGISTRY}/${LOCAL\_REPOSITORY}:${OCP\_RELEASE}-${ARCHITECTURE}"

# OpenShift Installation

Remove existing bootstrap and master from ~/.ssh/authorized\_keys

1. Create directory called /opt/registry/06172021

Copy the install\_config.yaml to /opt/registry/06172021

2. Create manifest

cd /usr/local/bin

./openshift-install create manifests --dir=/opt/registry/06172021

3. Remove files:

rm -f openshift/99\_openshift-cluster-api\_master-machines-\*.yamlopenshift/99\_openshift-cluster-api\_worker-machineset-\*.yaml

4. go to /opt/registry/06172021/manifests/cluster-scheduler-02-config.yml

change mastersSchedulable from true to false

5. Create ignition configs

cd /usr/local/bin

./openshift-install create ignition-configs --dir=/opt/registry/06172021

6. Obtain infra ID:

jq -r .infraID /opt/registry/06172021/metadata.json

dpsocp-dev-xbmrq

7. Copy file to http server

cp bootstrap.ign and other ign files to /var/www/html/

cmhmod 777 bootstrap.ign

make sure you are able to access it.

8. Install Cluster:

./openshift-install --dir=/opt/registry/06172021 wait-for bootstrap-complete --log-level=debug

9. Export Kubeconfig

export KUBECONFIG=/opt/registry/06172021/auth/kubeconfig

10. Get CSR Status & Approve Nodes:

oc get csr&

oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}{{end}}{{end}}' | xargsocadm certificate approve

# OpenShift Verification

1. Login to the cluster API using OpenSift-config file
2. Verify the CSR status (pending csr) and approve all certificates
3. Verify all Operator status and make sure no operator is degraded.
4. Login to the OpenShift console and verify the events.

# OpenShift HTPasswd configuration

1. Create or update your flat file with a username and hashed password:

2. Create an OpenShift Container Platform Secret object that contains the HTPasswduser’s file.

3. Create CR file.

4. Adding an identity provider to your clusters by applying the CR

5. Alocate the role.

# OpenShift LDAP configuration

1. Log in to your Linux host by using a Secure Shell (SSH) client.

2. Create an ad\_sync.yaml Active Directory-based LDAP sync configuration file with the following configuration and save the file

3. Modify the below YAML file

kind: LDAPSyncConfig

apiVersion: v1

url: ldaps://dc01rpl.rainpole.io

insecure: false

ca: <path>/rainpole-root-ca.crt

bindDN: CN=svc-domain-join,OU=Security Users,DC=rainpole,DC=io

bindPassword: <rainpole/svc-domain-join password>

groupUIDNameMapping:

"CN=ug-kub-admins,OU=Security Groups,DC=rainpole,DC=io": rainpoleadmins

"CN=ug-kub-readonly,OU=Security Groups,DC=rainpole,DC=io": rainpolereadonly

activeDirectory:

usersQuery:

baseDN: "OU=Security Users,DC=rainpole,DC=io"

scope: sub

derefAliases: never

filter: (objectclass=person)

pageSize: 0

userNameAttributes: [ sAMAccountName ]

groupMembershipAttributes: [ memberOf ]

4. Sync the ug-kub-admins and ug-kub-readonly groups.

5. Apply the respective cluster roles to each group.

# Configure OpenShift Container Storage (OCS)

This solution is designed to be deployed where latencies do not exceed 4 milliseconds round-trip time (RTT) between locations. Contact Red Hat Customer Support if you are planning to deploy with higher latencies.

Each node must be pre-labeled with its zone label. To label the nodes use the following command:

$ oc label nodes <NodeNames> topology.kubernetes.io/zone='<label>'

For example, you can label the nodes as follows:

topology.kubernetes.io/zone=arbiter to master or worker node

topology.kubernetes.io/zone=datacenter1 to at least two worker nodes

topology.kubernetes.io/zone=datacenter2 to at least two worker nodes

Compact mode requirements [Technology Preview]

OpenShift Container Storage can be installed on a three-node OpenShift compact bare metal cluster, where all the workloads run on three strong master nodes. There are no worker or storage nodes.

To configure OpenShift Container Platform in compact mode, see Configuring a three-node cluster and Delivering a Three-node Architecture for Edge Deployments. [Technology Preview]

Minimum starting node requirements [Technology Preview]

An OpenShift Container Storage cluster will be deployed with minimum configuration when the standard deployment resource requirement is not met. See Resource requirements section in Planning guide.

## Installing Red Hat OpenShift Container Storage Operator

You can install Red Hat OpenShift Container Storage Operator using the Red Hat OpenShift Container Platform Operator Hub.

**Prerequisites**

* Access to an OpenShift Container Platform cluster using an account with cluster-admin and Operator installation permissions.
* You have at least three worker nodes in the RHOCP cluster.
* For additional resource requirements, see [Planning your deployment](https://access.redhat.com/documentation/en-us/red_hat_openshift_container_storage/4.7/html-single/planning_your_deployment/index).

## Creating OpenShift Container Storage cluster on bare metal

**Procedure**

1. Log into the OpenShift Web Console.
2. Click **Operators → Installed Operators** to view all the installed operators.

Ensure that the **Project** selected is openshift-storage.

1. Click **OpenShift Container Storage** → **Create Instance** link of Storage Cluster.
2. Select mode as **Internal-Attached devices**.

You are prompted to install the Local Storage Operator if it is not already installed. Click **Install** and follow the procedure as described in [Installing Local Storage Operator](https://access.redhat.com/documentation/en-us/red_hat_openshift_container_storage/4.7/html-single/deploying_openshift_container_storage_using_bare_metal_infrastructure/index#installing-local-storage-operator_rhocs).

You can create a dedicated storage class to consume storage by filtering a set of storage volumes.

1. Discover disks
   1. Choose one of the following:
      * **All nodes** to discover disks in all the nodes.
      * **Select nodes** to discover disks from a subset of available nodes.
2.  Click **Next**.

 Create Storage class

1. Enter the **Volume Set Name**.
2. Enter the **Storage Class Name**. By default, the volume set name appears for the storage class name. You can also change the name.
3. The nodes selected for disk discovery in the previous step are displayed in the **Filter Disks** section.

Choose one of the following:

* + **All nodes** to select all the nodes for which you discovered the devices.
  + **Select nodes** to select a subset of the nodes for which you discovered the devices. Spread the worker nodes across three different physical nodes, racks or failure domains for high availability.

Select the required **Disk Type** from the following available options:

|  |  |
| --- | --- |
| All | Selects all types of disks present on the nodes. By default, this option is selected. |
| SSD/NVME | Selects only SSD type of disks. |
| HDD | Selects only HDD type of disks. |

It is recommended to select SSD/NVME disk type.

1. Expand the **Advanced** section and set the following options:

|  |  |
| --- | --- |
| Volume Mode | Block is selected by default. |
| Device Type | Select disk types. By default, Disk and Part are selected. |
| Disk Size | Minimum and maximum available size of the device that needs to be included.  Note  You must set a minimum size of 100GB for the device. |
| Max Disk Limit | This indicates the maximum number of PVs that can be created on a node. If this field is left empty, then PVs are created for all the available disks on the matching nodes. |

1. Click **Next**. A pop-up to confirm creation of the new storage class is displayed.
2. Click **Yes** to continue.

 Set Storage and nodes

1. Select **Storage Class**. By default, the new storage class created in the previous step is selected.
2. (Optional) Select **Enable arbiter** checkbox if you want to use the stretch clusters. This option is available only when all the prerequisites for arbiter are fulfilled and the Selected Nodes are populated. For more details, see [Arbiter stretch cluster requirements [Technology Preview](https://access.redhat.com/documentation/en-us/red_hat_openshift_container_storage/4.7/html-single/deploying_openshift_container_storage_using_bare_metal_infrastructure/index#requirements-for-installing-openshift-container-storage-using-local-storage-devices_rhocs)].
   * Select the **arbiter zone** from the available drop down list.
3. **Selected Nodes** shows the nodes selected in the previous step. This list takes a few minutes to reflect the disks that were discovered in the previous step.
4. Click **Next**.