Red Hat OpenShift Container Storage 4.5

Planning your deployment

Important considerations when deploying RHOCS 4.5

Last Updated: 2021-04-27

Red Hat OpenShift Container Storage 4.5 Planning your deployment

Important considerations when deploying RHOCS 4.5

Legal Notice

Abstract

Read this document for important considerations when planning your Red Hat OpenShift Container Storage deployment.

Table of Contents

CHAPTER 1. INTRODUCTION TO RED HAT OPENSHIFT CONTAINER STORAGE

Red Hat OpenShift Container Storage is a highly integrated collection of cloud storage and data services for Red Hat OpenShift Container Platform. It is available as part of the Red Hat OpenShift Container Platform Service Catalog, packaged as an operator to facilitate simple deployment and management.

Red Hat OpenShift Container Storage services are primarily made available to applications by way of storage classes that represent the following components:

- Block storage devices, catering primarily to database workloads. Prime examples include Red Hat OpenShift Container Platform logging and monitoring, and PostgreSQL.

- Shared and distributed file system, catering primarily to software development, messaging, and data aggregation workloads. Examples include Jenkins build sources and artifacts, Wordpress uploaded content, Red Hat OpenShift Container Platform registry, and messaging using JBoss AMQ.

- Multicloud object storage, featuring a lightweight S3 API endpoint that can abstract the storage and retrieval of data from multiple cloud object stores.

- On premises object storage, featuring a robust S3 API endpoint that scales to tens of petabytes and billions of objects, primarily targeting data intensive applications. Examples include the storage and access of row, columnar, and semi-structured data with applications like Spark, Presto, Red Hat AMQ Streams (Kafka), and even machine learning frameworks like TensorFlow and Pytorch.

Red Hat OpenShift Container Storage version 4.x integrates a collection of software projects, including:

- Ceph, providing block storage, a shared and distributed file system, and on premises object storage.

- Ceph CSI, to manage provisioning and lifecycle of persistent volumes and claims.

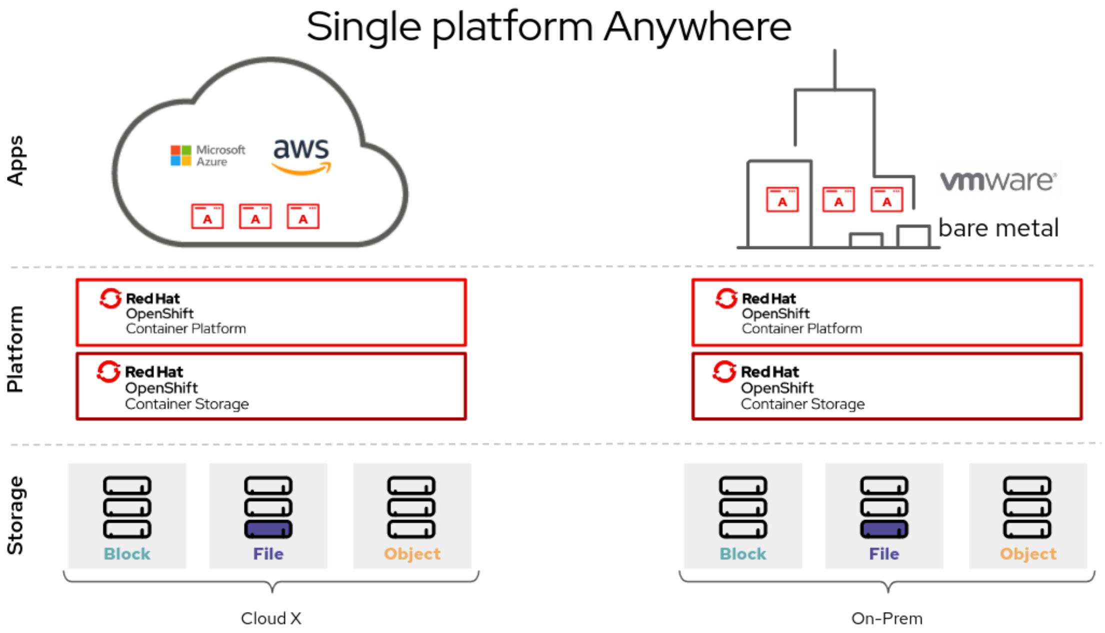
- NooBaa, providing a Multicloud Object Gateway

OpenShift Container Storage, Rook-Ceph, and Noobaa operators to initialize and manage OpenShift Container Storage services.

CHAPTER 2. ARCHITECTURE OF OPENSHIFT CONTAINER STORAGE

Red Hat OpenShift Container Storage provides services for, and can run internally from Red Hat OpenShift Container Platform.

Red Hat OpenShift Container Storage architecture



Red Hat OpenShift Container Storage supports deployment into Red Hat OpenShift Container Platform clusters deployed on Installer Provisioned Infrastructure or User Provisioned Infrastructure. For details about these two approaches, see OpenShift Container Platform - Installation process. To know more about interoperability of components for the Red Hat OpenShift Container Storage and Red Hat OpenShift Container Platform, see the Interoperabilty matrix.

For information about the architecture and lifecycle of OpenShift Container Platform, see OpenShift Container Platform architecture.

2.1. ABOUT OPERATORS

Red Hat OpenShift Container Storage comprises three main operators, which codify administrative tasks and custom resources so that task and resource characteristics can be easily automated. Administrators define the desired end state of the cluster, and the OpenShift Container Storage operators ensure the cluster is either in that state, or approaching that state, with minimal administrator intervention.

OpenShift Container Storage (OCS) operator

A meta-operator that codifies and enforces the recommendations and requirements of a supported Red Hat OpenShift Container Storage deployment by drawing on other operators in specific, tested ways. This operator provides the storage cluster resource that wraps resources provided by the Rook-Ceph and NooBaa operators.

Rook-Ceph operator

This operator automates the packaging, deployment, management, upgrading, and scaling of persistent storage and file, block, and object services. It creates block and file storage classes for all environments, and creates an object storage class and services object bucket claims made against it in on-premises environments.

Additionally, for internal mode clusters, it provides the Ceph cluster resource, which manages the deployments and services representing the following:

- Object storage daemons (OSDs)
- Monitors (MONs)
- Manager (MGR)
- Metadata servers (MDS)
- Object gateways (RGW) on-premises only

NooBaa operator

This operator automates the packaging, deployment, management, upgrading, and scaling of the Multicloud Object Gateway object service. It creates an object storage class and services object bucket claims made against it.

Additionally, it provides the NooBaa cluster resource, which manages the deployments and services for NooBaa core, database, and endpoint.

2.2. STORAGE CLUSTER DEPLOYMENT APPROACHES

Flexibility is a core tenet of Red Hat OpenShift Container Storage, as evidenced by its growing list of operating modalities. This section provides you with information that will help you to select the most appropriate approach for your environments. OpenShift Container Storage can be deployed either entirely within OpenShift Container Platform (Internal approach) or to make available the services from a cluster running outside of OpenShift Container Platform (External approach).

2.2.1. Internal approach

Deployment of Red Hat OpenShift Container Storage entirely within Red Hat OpenShift Container Platform has all the benefits of operator based deployment and management. There are two different deployment modalities available when Red Hat OpenShift Container Storage is running entirely within Red Hat OpenShift Container Platform:

- Simple
- Optimized

Simple deployment

Red Hat OpenShift Container Storage services run co-resident with applications, managed by operators in Red Hat OpenShift Container Platform.

A simple deployment is best for situations where

- Storage requirements are not clear
- OpenShift Container Storage services will run co-resident with applications
- Creating a node instance of a specific size is difficult (bare metal)

In order for Red Hat OpenShift Container Storage to run co-resident with applications, they must have local storage devices, or portable storage devices attached to them dynamically. For example, EBS volumes on EC2, or vSphere Virtual Volumes on VMware.

Optimized deployment

OpenShift Container Storage services run on dedicated infrastructure nodes managed by Red Hat OpenShift Container Platform.

An optimized approach is best for situations when:

- Storage requirements are clear
- OpenShift Container Storage services run on dedicated infrastructure nodes
- Creating a node instance of a specific size is easy (Cloud, Virtualized environment, etc.)

2.2.2. External approach

Red Hat OpenShift Container Storage exposes the Red Hat Ceph Storage services running outside of the OpenShift Container Platform cluster as storage classes.

The external approach is best used when

- Storage requirements are significant (600+ storage devices)
- Multiple OpenShift Container Platform clusters need to consume storage services from a common external cluster.
- Another team (SRE, Storage, etc.) needs to manage the external cluster providing storage services. Possibly pre-existing.

After installing the OpenShift Container Storage operator, you can create a Storage Cluster resource from the OpenShift Console using an installation wizard, or submitted manually; in either Internal or External mode.

2.3. NODE TYPES

Nodes run the container runtime, as well as services, to ensure that containers are running, and maintain network communication and separation between pods. In OpenShift Container Storage, there are three types of nodes.

Table 2.1. Types of nodes

Node Type	Description
Master	These nodes run processes that expose the Kubernetes API, watch and schedule newly created pods, maintain node health and quantity, and control interaction with underlying cloud providers.

Node Type	Description
Infrastructure (Infra)	Infra nodes run cluster level infrastructure services such as logging, metrics, registry, and routing. These are optional in OpenShift Container Platform clusters, but recommended for OpenShift Container Storage in virtualized and cloud environments. To create Infra nodes, you can provision new nodes labeled as Infra. See How to label Red Hat OpenShift Storage "nodes" as an infra node?
Worker	Worker nodes are also known as application nodes since they run applications. When OpenShift Container Storage is deployed in internal mode, a minimal cluster of 3 worker nodes is required, where the nodes are spread across three different racks, or availability zones, to ensure availability. In order for OpenShift Container Storage to run on worker nodes, they must either have local storage devices, or portable storage devices attached to them dynamically. When it is deployed in external mode, it runs on multiple nodes to allow rescheduling by K8S on available nodes in case of a failure. Examples of portable storage devices are EBS volumes on EC2, or vSphere Virtual Volumes on VMware.

NOTE

Nodes that run only storage workloads require a subscription for Red Hat OpenShift Container Storage. Nodes that run other workloads in addition to storage workloads require both Red Hat OpenShift Container Storage and Red Hat OpenShift Container Platform subscriptions. See Chapter 6, *Subscriptions* for more information.

CHAPTER 3. INTERNAL STORAGE SERVICES

Red Hat OpenShift Container Storage service is available for consumption internally to the Red Hat OpenShift Container Platform running on the following infrastructure:

- Amazon Web Services
- Bare Metal
- Google Cloud [Technology Preview]
- Microsoft Azure [Technology Preview]
- VMware vSphere

Ease of deployment and management are the highlights of running OpenShift Container Storage services internally on OpenShift Container Platform. Creation of an internal cluster resource will result in the internal provisioning of the OpenShift Container Storage base services, and make additional storage classes available to applications.

CHAPTER 4. EXTERNAL STORAGE SERVICES

Red Hat OpenShift Container Storage can make services from an external Red Hat Ceph Storage cluster available for consumption from OpenShift Container Platform clusters running on the following platforms:

- Bare Metal
- VMware vSphere

The OpenShift Container Storage operators create and manage services to satisfy persistent volume and object bucket claims against external services. External cluster resources make additional storage classes available to applications. External clusters are not deployed or managed by operators.

CHAPTER 5. SECURITY CONSIDERATIONS

5.1. FIPS-140-2

The Federal Information Processing Standard Publication 140-2 (FIPS-140-2) is a standard defining a set of security requirements for the use of cryptographic modules. This standard is mandated by law for US government agencies and contractors and is also referenced in other international and industry specific standards.

Red Hat OpenShift Container Storage is now using FIPS validated cryptographic modules as delivered by Red Hat Enterprise Linux OS/CoreOS (RHCOS).

The cryptography modules are currently being processed by Cryptographic Module Validation Program (CMVP) and their state can be seen at Modules in Process List . For more up-to-date information, see the knowledge base article.

NOTE

FIPS mode must be enabled on the OpenShift Container Platform, prior to installing OpenShift Container Storage. OpenShift Container Platform must run on RHCOS nodes, as OpenShift Container Storage deployment on RHEL 7 is not supported for this feature.

For more information, see installing a cluster in FIPS mode and support for FIPS cryptography .

5.2. PROXY ENVIRONMENT

A proxy environment is a production environment that denies direct access to the internet and provides an available HTTP or HTTPS proxy instead. Red Hat Openshift Container Platform is configured to use a proxy by modifying the proxy object for existing clusters or by configuring the proxy settings in the install-config.yaml file for new clusters.

Red Hat supports deployment of Openshift Container Storage versions 4.5 and higher in proxy environments when OpenShift Container Platform has been configured according to configuring the cluster-wide proxy.

CHAPTER 6. SUBSCRIPTIONS

6.1. SUBSCRIPTION OFFERINGS

Red Hat OpenShift Container Storage subscription is based on “core-pairs,” similar to Red Hat OpenShift Container Platform. The Red Hat OpenShift Container Storage 2-core subscription is based on the number of logical cores on the CPUs in the system where OpenShift Container Platform runs.

As with OpenShift Container Platform:

- OpenShift Container Storage subscriptions are stackable to cover larger hosts.
- Cores can be distributed across as many virtual machines (VMs) as needed. For example, ten 2-core subscriptions will provide 20 cores that can be used across any number of VMs.
- OpenShift Container Storage subscriptions are available with Premium or Standard support.

6.2. DISASTER RECOVERY SUBSCRIPTIONS

Red Hat OpenShift Container Storage does not offer disaster recovery (DR), cold backup, or other subscription types. Any system with OpenShift Container Storage installed, powered-on or powered-off, running workload or not, requires an active subscription.

6.3. CORES VERSUS VCPUS AND HYPERTHREADING

Making a determination about whether or not a particular system consumes one or more cores is currently dependent on whether or not that system has hyperthreading available. Hyperthreading is only a feature of Intel CPUs. Visit the Red Hat Customer Portal to determine whether a particular system supports hyperthreading.

For systems where hyperthreading is enabled and where one hyperthread equates to one visible system core, the calculation of cores is a ratio of 2 cores to 4 vCPUs. Therefore, a 2-core subscription covers 4 vCPUs in a hyperthreaded system. A large virtual machine (VM) might have 8 vCPUs, equating to 4 subscription cores. As subscriptions come in 2-core units, you will need two 2-core subscriptions to cover these 4 cores or 8 vCPUs.

Where hyperthreading is not enabled, and where each visible system core correlates directly to an underlying physical core, the calculation of cores is a ratio of 2 cores to 2 vCPUs.

6.4. SPLITTING CORES

Systems that require an odd number of cores need to consume a full 2-core subscription. For example, a system that is calculated to require only 1 core will end up consuming a full 2-core subscription once it is registered and subscribed.

When a single virtual machine (VM) with 2 vCPUs uses hyperthreading resulting in 1 calculated vCPU, a full 2-core subscription is required; a single 2-core subscription may not be split across two VMs with 2 vCPUs using hyperthreading. See section Cores versus vCPUs and hyperthreading for more information.

It is recommended that virtual instances be sized so that they require an even number of cores.

6.5. SUBSCRIPTION REQUIREMENTS

OpenShift Container Storage components can run on either OpenShift Container Platform worker or infrastructure nodes, for which either Red Hat CoreOS (RHCOS) or Red Hat Enterprise Linux (RHEL) 7 can be used as the host operating system. When worker nodes are used for OpenShift Container Storage components, those nodes are required to have subscriptions for both OpenShift Container Platform and OpenShift Container Storage. When infrastructure nodes are used, those nodes are only required to have OpenShift Container Storage subscriptions. Labels are used to indicate whether a node should be considered a worker or infrastructure node.

CHAPTER 7. INFRASTRUCTURE REQUIREMENTS

7.1. PLATFORM REQUIREMENTS

Red Hat OpenShift Container Storage can be combined with an OpenShift Container Platform release that is one minor release behind or ahead of the OpenShift Container Storage version.

OpenShift Container Storage 4.5 can run on:

- OpenShift Container Platform 4.4 (one version behind) for internal mode only
- OpenShift Container Platform 4.5 (same version)
- OpenShift Container Platform 4.6 (one version ahead)

For external clusters, OpenShift Container Platform version 4.5 or higher is required to deploy Red Hat OpenShift Container Storage 4.5. For external cluster subscription requirements, see this Red Hat Knowledgebase article.

For a complete list of supported platform versions, see the Red Hat OpenShift Container Storage and Red Hat OpenShift Container Platform interoperability matrix.

7.1.1. Amazon EC2

Supports internal Red Hat Openshift Container Storage clusters only.

An Internal cluster must both meet storage device requirements and have a storage class providing either

- EBS storage via the aws-ebs provisioner
- Instance storage via the Local Storage Operator

7.1.2. Bare Metal

Supports internal clusters and consuming external clusters.

An Internal cluster must both meet storage device requirements and have a storage class providing local SSD (NVMe/SATA/SAS, SAN) via the Local Storage Operator.

7.1.3. VMware

Supports internal clusters and consuming external clusters.

Recommended version: vSphere 6.7, Update 2

See VMware vSphere infrastructure requirements for details.

Additionally, an Internal cluster must both meet storage device requirements and have a storage class providing either

- vSAN or VMFS datastore via the vsphere-volume provisioner
- VMDK, RDM, or DirectPath storage devices via the Local Storage Operator.

7.1.4. Google Cloud [Technology Preview]

Supports internal Red Hat Openshift Container Storage clusters only.

An Internal cluster must both meet storage device requirements and have a storage class providing

- GCE Persistent Disk via the gce-pd provisioner

7.1.5. Microsoft Azure [Technology Preview]

Supports internal Red Hat Openshift Container Storage clusters only.

An Internal cluster must both meet storage device requirements and have a storage class providing

- Azure disk via the azure-disk provisioner

7.1.6. Red Hat Virtualization Platform [Technology Preview]

Supports internal Red Hat Openshift Container Storage clusters only.

An Internal cluster must both meet storage device requirements and have a storage class providing local SSD (NVMe/SATA/SAS, SAN) via the Local Storage Operator.

7.1.7. Red Hat OpenStack Platform [Technology Preview]

Supports internal Red Hat Openshift Container Storage clusters only.

An Internal cluster must both meet storage device requirements and have a storage class providing

- standard disk via the Cinder provisioner

7.2. EXTERNAL MODE REQUIREMENT

7.2.1. Red Hat Ceph Storage

Red Hat Ceph Storage (RHCS) version 4.1.1 or later is required. For more information on versions supported, see this knowledge base article on Red Hat Ceph Storage releases and corresponding Ceph package versions.

For instructions regarding how to install a RHCS 4 cluster, see Installation guide.

7.3. RESOURCE REQUIREMENTS

OpenShift Container Storage services consist of an initial set of base services, followed by additional device sets. All of these OpenShift Container Storage services pods are scheduled by kubernetes on OpenShift Container Platform nodes according to Section 7.4, “Pod placement rules” .

Table 7.1. Aggregate resource requirements

Deployment Mode	Base services	Additional device Set

Deployment Mode	Base services	Additional device Set
Internal	- 30 CPU (logical) - 72 GiB memory - 3 storage devices	- 6 CPU (logical) - 15 GiB memory - 3 storage devices
External	- 4 CPU (logical) - 16 GiB memory	Not applicable

CPU units

In this section, 1 CPU Unit maps to the Kubernetes concept of 1 CPU unit.

- 1 unit of CPU is equivalent to 1 core for non-hyperthreaded CPUs.
- 2 units of CPU are equivalent to 1 core for hyperthreaded CPUs.
- OpenShift Container Storage core-based subscriptions always come in pairs (2 cores).

Example: For a 3 node cluster with a single device set , a minimum of 3 x 10 = 30 units of CPU are required. 30 units of CPU are equivalent to 15 cores which is equivalent to ~8 quantities of Red Hat OpenShift Container Storage (2 core) subscriptions.

For more information, see Chapter 6, *Subscriptions* and CPU units.

7.4. POD PLACEMENT RULES

Kubernetes is responsible for pod placement based on declarative placement rules. The OpenShift Container Storage base service placement rules for Internal cluster can be summarized as follows:

- Nodes are labeled with the **cluster.ocs.openshift.io/openshift-storage** key
- Nodes are sorted into pseudo failure domains if none exist
- Components requiring high availability are spread across failure domains
- A storage device must be accessible in each failure domain

This leads to the requirement that there be at least three nodes, and that nodes be in three distinct rack or zone failure domains in the case of pre-existing topology labels.

For additional device sets, there must be a storage device, and sufficient resources for the pod consuming it, in each of the three failure domains. Manual placement rules can be used to override default placement rules, but generally this approach is only suitable for bare metal deployments.

7.5. STORAGE DEVICE REQUIREMENTS

Use this section to understand the different storage capacity requirements that you can consider when planning internal mode deployments and upgrades. We generally recommend 9 devices or less per node. This recommendation ensures both that nodes stay below cloud provider dynamic storage device attachment limits, and to limit the recovery time after node failures with local storage devices. Expanding the cluster in multiples of three, one node in each failure domain, is an easy way to satisfy pod placement rules.

NOTE

You can expand the storage capacity only in the increment of the capacity selected at the time of installation.

7.5.1. Dynamic storage devices

Red Hat OpenShift Container Storage permits the selection of either 0.5 TiB, 2 TiB or 4 TiB capacities as the request size for dynamic storage device sizes. The number of dynamic storage devices that can run per node is a function of the node size, underlying provisioner limits and resource requirements.

7.5.2. Local storage devices

For local storage deployment, any disk size of 4 TiB or less can be used, and all disks should be of the same size and type. The number of local storage devices that can run per node is a function of the node size and resource requirements. Expanding the cluster in multiples of three, one node in each failure domain, is an easy way to satisfy pod placement rules.

NOTE

Disk partitioning is not supported.

7.5.3. Capacity planning

Always ensure that available storage capacity stays ahead of consumption. Recovery is difficult if available storage capacity is completely exhausted, and requires more intervention than simply adding capacity or deleting or migrating content.

Capacity alerts are issued when cluster storage capacity reaches 75% (near-full) and 85% (full) of total capacity. Always address capacity warnings promptly, and review your storage regularly to ensure that you do not run out of storage space. If you do run out of storage space completely, contact Red Hat Customer Support.

The following tables show example node configurations for Red Hat OpenShift Container Storage with dynamic storage devices.

Table 7.2. Example initial configurations with 3 nodes

Storage Device size	Storage Devices per node	Total capacity	Usable storage capacity
0.5 TiB	1	1.5 TiB	0.5 TiB
2 TiB	1	6 TiB	2 TiB
4 TiB	1	12 TiB	4 TiB

Table 7.3. Example of expanded configurations with 30 nodes (N)

Storage Device size (D)	Storage Devices per node (M)	Total capacity (D * M * N)	Usable storage capacity (D*M*N/3)
0.5 TiB	3	45 TiB	15 TiB
2 TiB	6	360 TiB	120 TiB
4 TiB	9	1080 TiB	360 TiB

CHAPTER 8. DISCONNECTED ENVIRONMENT

Disconnected environment is a network restricted environment where Operator Lifecycle Manager (OLM) cannot access the default Operator Hub and image registries, which require Internet connectivity.

Red Hat supports deployment of OpenShift Container Storage in disconnected environments where OpenShift Container Platform is installed in restricted networks.

NOTE

When you install OpenShift Container Storage in a restricted network environment, apply a custom Network Time Protocol (NTP) configuration to the nodes, because by default, internet connectivity is assumed in OpenShift Container Platform and chronyd is configured to use ***.rhel.pool.ntp.org** servers. See Red Hat Knowledgebase article and Configuring chrony time service for more details.

For more information, see Preparing to deploy OpenShift Container Storage in disconnected environments.

CHAPTER 9. NEXT STEPS

To start deploying your OpenShift Container Storage, you can use the internal mode within OpenShift Container Platform or use external mode to make available services from a cluster running outside of OpenShift Container Platform.

Depending on your requirement, go to the respective deployment guides.

Internal mode

- Deploying OpenShift Container Storage using Amazon web services
- Deploying OpenShift Container Storage using Bare Metal
- Deploying OpenShift Container Storage using Google Cloud [Technology Preview]
- Deploying OpenShift Container Storage using Microsoft Azure [Technology Preview]
- Deploying OpenShift Container Storage using VMware vSphere
- Deploying OpenShift Container Storage using Red Hat Virtualization Platform [Technology Preview]
- Deploying OpenShift Container Storage using Red Hat OpenStack Platform [Technology Preview]

External mode

- Deploying OpenShift Container Storage in external mode