



OpenShift Dedicated 4

Architecture

An overview of OpenShift Dedicated architecture

OpenShift Dedicated 4 Architecture

An overview of OpenShift Dedicated architecture

Legal Notice

Copyright © 2021 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This document provides an overview of the platform and application architecture in OpenShift Dedicated.

Table of Contents

CHAPTER 1. INTRODUCTION TO OPENSIFT DEDICATED	3
1.1. INTRODUCTION TO OPENSIFT DEDICATED	3
1.1.1. Custom operating system	3
1.1.2. Other key features	3
1.1.3. Internet and Telemetry access for OpenShift Dedicated	4
CHAPTER 2. ARCHITECTURE CONCEPTS	5
2.1. ABOUT KUBERNETES	5
2.2. THE BENEFITS OF CONTAINERIZED APPLICATIONS	5
2.2.1. Operating system benefits	5
2.2.2. Deployment benefits	5
2.3. UNDERSTANDING HOW OPENSIFT DEDICATED DIFFERS FROM OPENSIFT CONTAINER PLATFORM	6

CHAPTER 1. INTRODUCTION TO OPENSIFT DEDICATED

With its foundation in Kubernetes, OpenShift Dedicated is a complete OpenShift Container Platform cluster provided as a cloud service, configured for high availability, and dedicated to a single customer.

1.1. INTRODUCTION TO OPENSIFT DEDICATED

OpenShift Dedicated is professionally managed by Red Hat and hosted on Amazon Web Services (AWS) or Google Cloud Platform (GCP). Each OpenShift Dedicated cluster comes with a fully managed [control plane](#) (Control and Infrastructure nodes), application nodes, installation and management by Red Hat Site Reliability Engineers (SRE), premium Red Hat Support, and cluster services such as logging, metrics, monitoring, notifications portal, and a cluster portal.

OpenShift Dedicated provides enterprise-ready enhancements to Kubernetes, including the following enhancements:

- OpenShift Dedicated clusters are deployed on AWS or GCP environments and can be used as part of a hybrid approach for application management.
- Integrated Red Hat technology. Major components in OpenShift Dedicated come from Red Hat Enterprise Linux and related Red Hat technologies. OpenShift Dedicated benefits from the intense testing and certification initiatives for Red Hat's enterprise quality software.
- Open source development model. Development is completed in the open, and the source code is available from public software repositories. This open collaboration fosters rapid innovation and development.

To learn about options for assets you can create when you build and deploy containerized Kubernetes applications in OpenShift Container Platform, see [Understanding OpenShift Container Platform development](#) in the OpenShift Container Platform documentation.

1.1.1. Custom operating system

OpenShift Dedicated uses Red Hat Enterprise Linux CoreOS (RHCOS), a container-oriented operating system that combines some of the best features and functions of the CoreOS and Red Hat Atomic Host operating systems. RHCOS is specifically designed for running containerized applications from OpenShift Dedicated and works with new tools to provide fast installation, Operator-based management, and simplified upgrades.

RHCOS includes:

- Ignition, which OpenShift Dedicated uses as a firstboot system configuration for initially bringing up and configuring machines.
- CRI-O, a Kubernetes native container runtime implementation that integrates closely with the operating system to deliver an efficient and optimized Kubernetes experience. CRI-O provides facilities for running, stopping, and restarting containers.
- Kubelet, the primary node agent for Kubernetes that is responsible for launching and monitoring containers.

1.1.2. Other key features

Operators are both the fundamental unit of the OpenShift Dedicated code base and a convenient way to deploy applications and software components for your applications to use. In OpenShift Dedicated,

Operators serve as the platform foundation and remove the need for manual upgrades of operating systems and control plane applications. OpenShift Dedicated Operators such as the Cluster Version Operator and Machine Config Operator allow simplified, cluster-wide management of those critical components.

Operator Lifecycle Manager (OLM) and the OperatorHub provide facilities for storing and distributing Operators to people developing and deploying applications.

The Red Hat Quay Container Registry is a Quay.io container registry that serves most of the container images and Operators to OpenShift Dedicated clusters. Quay.io is a public registry version of Red Hat Quay that stores millions of images and tags.

Other enhancements to Kubernetes in OpenShift Dedicated include improvements in software defined networking (SDN), authentication, log aggregation, monitoring, and routing. OpenShift Dedicated also offers a comprehensive web console and the custom OpenShift CLI (**oc**) interface.

1.1.3. Internet and Telemetry access for OpenShift Dedicated

In OpenShift Dedicated, you require access to the Internet to install your cluster. The Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, also requires Internet access. If your cluster is connected to the Internet, Telemetry runs automatically, and your cluster is registered to the [OpenShift Cluster Manager \(OCM\)](#).

Once you confirm that your Red Hat OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually using OCM, use [subscription watch](#) to track your OpenShift Dedicated subscriptions at the account or multi-cluster level.

You must have Internet access to:

- Access the [OpenShift Cluster Manager \(OCM\)](#) page to download the installation program and perform subscription management. If the cluster has Internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Obtain the packages that are required to perform cluster updates.



IMPORTANT

If your cluster cannot have direct Internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the content that is required and use it to populate a mirror registry with the packages that you need to install a cluster and generate the installation program. With some installation types, the environment that you install your cluster in will not require Internet access. Before you update the cluster, you update the content of the mirror registry.

CHAPTER 2. ARCHITECTURE CONCEPTS

Learn about OpenShift and basic container concepts used in the OpenShift Dedicated architecture.

2.1. ABOUT KUBERNETES

Kubernetes is an open source container orchestration engine for automating deployment, scaling, and management of containerized applications. The general concept of Kubernetes is fairly simple:

- Start with one or more worker nodes to run the container workloads.
- Manage the deployment of those workloads from one or more control nodes.
- Wrap containers in a deployment unit called a pod. Using pods provides extra metadata with the container and offers the ability to group several containers in a single deployment entity.
- Create special kinds of assets. For example, services are represented by a set of pods and a policy that defines how they are accessed. This policy allows containers to connect to the services that they need even if they do not have the specific IP addresses for the services. Replication controllers are another special asset that indicates how many pod Replicas are required to run at a time. You can use this capability to automatically scale your application to adapt to its current demand.

To learn more about Kubernetes, see the [Kubernetes documentation](#).

2.2. THE BENEFITS OF CONTAINERIZED APPLICATIONS

Applications were once expected to be installed on operating systems that included all of the dependencies for the application. However, containers provide a standard way to package your application code, configurations, and dependencies into a single unit that can run as a resource-isolated process on a compute server. To run your app in Kubernetes on OpenShift Dedicated, you must first containerize your app by creating a container image that you store in a container registry.

2.2.1. Operating system benefits

Containers use small, dedicated Linux operating systems without a kernel. The file system, networking, cgroups, process tables, and namespaces are separate from the host Linux system, but the containers can integrate with the hosts seamlessly when necessary. Being based on Linux allows containers to use all the advantages that come with the open source development model of rapid innovation.

Because each container uses a dedicated operating system, you can deploy applications that require conflicting software dependencies on the same host. Each container carries its own dependent software and manages its own interfaces, such as networking and file systems, so applications never need to compete for those assets.

2.2.2. Deployment benefits

If you employ rolling upgrades between major releases of your application, you can continuously improve your applications without downtime and still maintain compatibility with the current release.

You can also deploy and test a new version of an application alongside the existing version. Deploy the new application version in addition to the current version. If the container passes your tests, simply deploy more new containers and remove the old ones.

Since all the software dependencies for an application are resolved within the container itself, you can use a generic operating system on each host in your data center. You do not need to configure a specific operating system for each application host. When your data center needs more capacity, you can deploy another generic host system.

2.3. UNDERSTANDING HOW OPENSIFT DEDICATED DIFFERS FROM OPENSIFT CONTAINER PLATFORM

OpenShift Dedicated uses the same code base as OpenShift Container Platform but is installed in an opinionated way to be optimized for performance, scalability, and security. OpenShift Dedicated is a fully managed service; therefore, many of the OpenShift Dedicated components and settings that you manually set up in OpenShift Container Platform are set up for you by default.

Review the following differences between OpenShift Dedicated and a standard installation of OpenShift Container Platform on your own infrastructure:

OpenShift Container Platform	OpenShift Dedicated
The customer installs and configures OpenShift Container Platform.	OpenShift Dedicated is installed through a user-friendly webpage and in a standardized way that is optimized for performance, scalability, and security.
Customers can choose their computing resources.	OpenShift Dedicated is hosted and managed in a public cloud (Amazon Web Services or Google Cloud Platform) either owned by Red Hat or provided by the customer.
Customers have top-level administrative access to the infrastructure.	Customers have a built-in administrator group, though the top-level administration access is available when cloud accounts are provided by the customer.
Customers can use all supported features and configuration settings available in OpenShift Container Platform.	Some OpenShift Container Platform features and configuration settings might not be available or changeable in OpenShift Dedicated .
You set up control plane components such as the API server and etcd on machines that get the control role. You can modify the control plane components, but keep in mind that you are responsible for backing up, restoring, and making control plane data highly available.	Red Hat sets up the control plane and manages the control plane components for you. The control plane is highly available.
You are responsible for updating the underlying infrastructure for the control plane and worker nodes. You can use the OpenShift web console to update OpenShift Container Platform versions.	Red Hat automatically notifies the customer when updates are available. You can manually or automatically schedule upgrades in OpenShift Cluster Manager (OCM).
Support is provided based on the terms of your Red Hat subscription or cloud provider.	Engineered, operated, and supported by Red Hat with a 99.95% uptime SLA and 24x7 coverage.

