



# OpenShift Dedicated 4

## Registry

OpenShift Dedicated 可以从源代码构建镜像、部署和管理其生命周期。



## OpenShift Dedicated 4 Registry

---

OpenShift Dedicated 可以从源代码构建镜像、部署和管理其生命周期。

## 法律通告

Copyright © 2024 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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

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

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> 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<sup>®</sup> 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.

## 摘要

OpenShift Dedicated 提供了一个内部集成的容器镜像 registry，它可以部署到 OpenShift Dedicated 环境中，以便本地管理镜像。

---

# 目录

<b>第 1 章 OPENSIFT 镜像 REGISTRY 概述</b> .....	<b>3</b>
1.1. OPENSIFT 镜像 REGISTRY 常用术语表	3
1.2. 集成的 OPENSIFT 镜像 REGISTRY	4
1.3. 第三方 REGISTRY	4
1.4. RED HAT QUAY REGISTRIES	5
1.5. 启用了身份验证的红帽 REGISTRY	5
<b>第 2 章 OPENSIFT DEDICATED 中的 IMAGE REGISTRY OPERATOR</b> .....	<b>6</b>
2.1. OPENSIFT DEDICATED 上的镜像 REGISTRY	6
<b>第 3 章 访问REGISTRY</b> .....	<b>7</b>
3.1. 检查 REGISTRY POD 的状态	7
3.2. 查看REGISTRY日志	7



# 第 1 章 OPENSIFT 镜像 REGISTRY 概述

OpenShift Dedicated 可以从源代码构建镜像、部署和管理其生命周期。它提供内部集成的容器镜像 registry，它可以部署到 OpenShift Dedicated 环境中，以本地管理镜像。此概述包含 OpenShift Dedicated 常用的 registry 的参考信息和链接，专注于 OpenShift 镜像 registry。

## 1.1. OPENSIFT 镜像 REGISTRY 常用术语表

该术语表定义了 registry 内容中使用的常用术语。

### container

包括软件及其所有依赖项的轻量级和可执行镜像。由于容器虚拟化操作系统，因此您可以在数据中心、公有云或私有云或本地主机中运行容器。

### Image Registry Operator

Image Registry Operator 在 **openshift-image-registry** 命名空间中运行，并管理该位置中的 registry 实例。

### 镜像仓库

镜像仓库是相关容器镜像和标识它们的标签（tag）的集合。

### 镜像 registry

mirror registry 是一个 registry，其中包含 OpenShift Dedicated 镜像的镜像（mirror）。

### namespace

命名空间隔离单个集群中的一组资源。

### pod

pod 是 Kubernetes 中的最小逻辑单元。pod 由一个或多个容器组成，可在 worker 节点上运行。

### 私有 registry

registry 是实现容器镜像 registry API 的服务器。私有 registry 是需要身份验证的 registry，允许用户访问其内容。

### 公共 registry

registry 是实现容器镜像 registry API 的服务器。公共 registry 是以公开方式提供其内容的 registry。

### Quay.io

由红帽提供和维护的公共 Red Hat Quay Container Registry 实例，为 OpenShift Dedicated 集群提供大多数容器镜像和 Operator。

### OpenShift 镜像 registry

OpenShift 镜像 registry 是 OpenShift Dedicated 提供的 registry，用于管理镜像。

### registry 身份验证

要将镜像推送（push）到私有镜像仓库，registry 需要根据凭据验证其用户。

### route

公开服务，以允许从 OpenShift Dedicated 实例外的用户和应用程序对 pod 进行网络访问。

### 缩减（scale down）

减少副本数。

### 扩展（scale up）

增加副本数量。

### service

服务在一组 pod 上公开正在运行的应用程序。

## 1.2. 集成的 OPENSIFT 镜像 REGISTRY

OpenShift Dedicated 提供了一个内建的镜像 registry，它作为一个标准的工作负载在集群中运行。这个 registry 由一个 infrastructure Operator 配置并管理。它为用户提供了一种现成的解决方案，供用户管理在已有集群基础架构上运行的，用于处理实际工作负载的镜像。这个 registry 可以象集群中的其他负载一样进行扩展，且不需要置备特殊的基础架构。此外，它已被集成到集群用户身份验证和授权系统中。这意味着，通过定义镜像资源上的用户权限就可以控制对镜像的创建和访问权限。

该 registry 通常作为集群中构建的镜像的发布目标，以及在集群中运行的工作负载的镜像源。当一个新镜像被推送到 registry 时，集群会收到新镜像的通知，其他组件就可以对更新的镜像做出反应。

镜像数据会存储在两个位置。实际镜像数据存储在可配置的存储位置，例如云存储或一个文件系统卷中。镜像的元数据被保存为标准的 API 资源（镜像(image)及镜像流(imagestream)），它们可以通过标准的集群 API 进行访问。

### 其他资源

- [OpenShift Dedicated 中的 Image Registry Operator](#)

## 1.3. 第三方 REGISTRY

OpenShift Dedicated 可以使用由第三方 registry 提供的镜像创建容器，但是这些 registry 可能不会象集成的 OpenShift 镜像 registry 一样提供相同的镜像通知支持。在这种情况下，OpenShift Dedicated 将在创建镜像流时从远程 registry 中获取 tag。要刷新获取的标签，请运行 **oc import-image <stream>**。当检测到新的镜像时，以前的构建和部署将会被重新创建。

### 1.3.1. 身份验证

OpenShift Dedicated 使用用户提供的凭证与 registry 进行联系来访问私有镜像仓库。这样，OpenShift Dedicated 就可以对私有仓库进行镜像的 push 和 pull 操作。

#### 1.3.1.1. 使用 Podman 进行 registry 身份验证

有些容器镜像 registry 需要访问授权。Podman 是一个开源工具，用于管理容器和容器镜像，并与镜像 registry 交互。您可以使用 Podman 来验证凭证、拉取 registry 镜像，并将本地镜像存储在本地文件系统中。以下是使用 Podman 验证 registry 的通用示例：

### 流程

1. 使用[红帽生态系统目录](#)从红帽仓库搜索特定容器镜像并选择所需的镜像。
2. 点 **Get this image** 来查找您的容器镜像的命令。
3. 运行以下命令，并输入您的用户名和密码进行验证：

```
$ podman login registry.redhat.io
Username:<your_registry_account_username>
Password:<your_registry_account_password>
```

4. 运行以下命令下载镜像并将其保存在本地：

```
$ podman pull registry.redhat.io/<repository_name>
```

## 1.4. RED HAT QUAY REGISTRIES

Red Hat Quay 为您提供了一个企业级的容器镜像 registry。它可以作为一个托管的服务，也可以在您自己的数据中心或环境中安装它。Red Hat Quay 中的高级功能包括跨区域复制、镜像扫描及镜像回滚（roll back）功能。

请通过 [Quay.io](#) 网站设置您自己的托管 Quay registry 帐户。之后，请按照 Quay 教程中的内容登录到 Quay registry 并开始管理镜像。

您可以象访问其他远程镜像 registry 一样，通过 OpenShift Dedicated 访问您的 Red Hat Quay registry。

### 其他资源

- [Red Hat Quay 产品文档](#)

## 1.5. 启用了身份验证的红帽 REGISTRY

红帽生态系统目录的容器镜像部分提供的所有容器镜像都托管在镜像 registry 上（[registry.redhat.io](#)）。

registry（[Registry.redhat.io](#)）需要进行身份验证才能访问 OpenShift Dedicated 上的镜像及内容。当迁移到新 registry 后，现有的 registry 仍将在一段时间内可用。



### 注意

OpenShift Dedicated 从 [Registry.redhat.io](#) 中提取（pull）镜像，因此需要配置集群以使用它。

新 registry 使用标准的 OAuth 机制进行身份验证：

- **身份验证令牌。**令牌（token）是服务帐户，由管理员生成。系统可以使用它们与容器镜像 registry 进行身份验证。服务帐户不受用户帐户更改的影响，因此使用令牌进行身份验证是一个可靠且具有弹性的方法。这是生产环境集群中唯一受支持的身份验证选项。
- **Web 用户名和密码。**这是用于登录到诸如 [access.redhat.com](#) 之类的资源的标准凭据集。虽然可以在 OpenShift Dedicated 上使用此身份验证方法，但在生产环境部署中不支持此方法。此身份验证方法应该只限于在 OpenShift Dedicated 之外的独立项目中使用。

您可以在 **podman login** 中使用您的凭证（用户名和密码，或身份验证令牌）来访问新 registry 中的内容。

所有镜像流都指向使用安装 pull secret 进行身份验证的新 registry。

您必须将凭证放在以下任一位置：

- **OpenShift 命名空间。**您的凭证必须存在于 **openshift** 命名空间中，以便 **openshift** 命名空间中的镜像流可以被导入。
- **您的主机。**您的凭据必须存在于主机上，因为在抓取（pull）镜像时，Kubernetes 会使用主机中的凭据。

### 其他资源

- [registry 服务帐户](#)

## 第 2 章 OPENSIFT DEDICATED 中的 IMAGE REGISTRY OPERATOR

### 2.1. OPENSIFT DEDICATED 上的镜像 REGISTRY

Image Registry Operator 安装一个单独的 OpenShift 镜像 registry 实例，并对 registry 的所有配置进行管理（包括设置 registry 存储）。

当部署了 control plane 后，Operator 将会根据集群中的配置创建一个默认的 **configs.imageregistry.operator.openshift.io** 资源实例。

如果没有足够的信息来定义完整的 **configs.imageregistry.operator.openshift.io** 资源，则将定义不完整的资源，Operator 将更新资源状态以提供缺失的内容。

Image Registry Operator 在 **openshift-image-registry** 命名空间中运行，并管理该位置中的 registry 实例。registry 的所有配置和工作负载资源都位于该命名空间中。

## 第 3 章 访问REGISTRY

在 OpenShift Dedicated 中，Red Hat Site Reliability Engineering (SRE) 会为您管理 registry。但是，您可以检查 registry pod 的状态并查看 registry 日志。

### 3.1. 检查 REGISTRY POD 的状态

作为具有 **dedicated-admin** 角色的管理员，您可以列出在 **openshift-image-registry** 项目中运行的镜像 registry pod，并检查其状态。

#### 先决条件

- 您可以使用具有 **dedicated-admin** 角色的用户访问集群。

#### 流程

- 列出 **openshift-image-registry** 项目中的 pod 并查看其状态：

```
$ oc get pods -n openshift-image-registry
```

#### 输出示例

```
NAME READY STATUS RESTARTS AGE
cluster-image-registry-operator-764bd7f846-qqtph 1/1 Running 0 78m
image-registry-79fb4469f6-llrln 1/1 Running 0 77m
node-ca-hjksc 1/1 Running 0 73m
node-ca-tftj6 1/1 Running 0 77m
node-ca-wb6ht 1/1 Running 0 77m
node-ca-zvt9q 1/1 Running 0 74m
```

### 3.2. 查看REGISTRY日志

使用 **oc logs** 命令可以查看 registry 中的日志信息。

#### 流程

- 使用带有 deployments 的 **oc logs** 命令查看容器镜像 registry 的日志：

```
$ oc logs deployments/image-registry -n openshift-image-registry
```

#### 输出示例

```
2015-05-01T19:48:36.300593110Z time="2015-05-01T19:48:36Z" level=info
msg="version=v2.0.0+unknown"
2015-05-01T19:48:36.303294724Z time="2015-05-01T19:48:36Z" level=info msg="redis not
configured" instance.id=9ed6c43d-23ee-453f-9a4b-031fea646002
2015-05-01T19:48:36.303422845Z time="2015-05-01T19:48:36Z" level=info msg="using
inmemory layerinfo cache" instance.id=9ed6c43d-23ee-453f-9a4b-031fea646002
2015-05-01T19:48:36.303433991Z time="2015-05-01T19:48:36Z" level=info msg="Using
OpenShift Auth handler"
2015-05-01T19:48:36.303439084Z time="2015-05-01T19:48:36Z" level=info msg="listening
on :5000" instance.id=9ed6c43d-23ee-453f-9a4b-031fea646002
```

