OpenShift Dedicated can build images from your source code, deploy them, and manage their lifecycle.
OpenShift Dedicated 4 Registry

OpenShift Dedicated can build images from your source code, deploy them, and manage their lifecycle.
Abstract

OpenShift Dedicated provides an internal, integrated container image registry that can be deployed in your OpenShift Dedicated environment to locally manage images.
# Table of Contents

**CHAPTER 1. OPENSIFT IMAGE REGISTRY OVERVIEW** .......................................................... 3  
1.1. GLOSSARY OF COMMON TERMS FOR OPENSHT IMAGE REGISTRY ........................ 3  
1.2. INTEGRATED OPENSHT IMAGE REGISTRY ......................................................... 4  
1.3. THIRD-PARTY REGISTRIES  ..................................................................................... 4  
  1.3.1. Authentication .................................................................................................... 4  
  1.3.1.1. Registry authentication with Podman .............................................................. 4  
1.4. RED HAT QUAY REGISTRIES ................................................................................... 5  
1.5. AUTHENTICATION ENABLED RED HAT REGISTRY ........................................... 5  

**CHAPTER 2. IMAGE REGISTRY OPERATOR IN OPENSIFT DEDICATED** ............................. 7  
2.1. IMAGE REGISTRY ON OPENSIFT DEDICATED ...................................................... 7  

**CHAPTER 3. ACCESSING THE REGISTRY** ................................................................. 8  
3.1. CHECKING THE STATUS OF THE REGISTRY PODS .......................................... 8  
3.2. VIEWING REGISTRY LOGS ..................................................................................... 8
CHAPTER 1. OPENSIFT IMAGE REGISTRY OVERVIEW

OpenShift Dedicated can build images from your source code, deploy them, and manage their lifecycle. It provides an internal, integrated container image registry that can be deployed in your OpenShift Dedicated environment to locally manage images. This overview contains reference information and links for registries commonly used with OpenShift Dedicated, with a focus on the OpenShift image registry.

1.1. GLOSSARY OF COMMON TERMS FOR OPENSIFT IMAGE REGISTRY

This glossary defines the common terms that are used in the registry content.

**container**

Lightweight and executable images that consist of software and all its dependencies. Because containers virtualize the operating system, you can run containers in a data center, a public or private cloud, or your local host.

**Image Registry Operator**

The Image Registry Operator runs in the `openshift-image-registry` namespace, and manages the registry instance in that location.

**image repository**

An image repository is a collection of related container images and tags identifying images.

**mirror registry**

The mirror registry is a registry that holds the mirror of OpenShift Dedicated images.

**namespace**

A namespace isolates groups of resources within a single cluster.

**pod**

The pod is the smallest logical unit in Kubernetes. A pod is comprised of one or more containers to run in a worker node.

**private registry**

A registry is a server that implements the container image registry API. A private registry is a registry that requires authentication to allow users access its contents.

**public registry**

A registry is a server that implements the container image registry API. A public registry is a registry that serves its contently publicly.

**Quay.io**

A public Red Hat Quay Container Registry instance provided and maintained by Red Hat, that serves most of the container images and Operators to OpenShift Dedicated clusters.

**OpenShift image registry**

OpenShift image registry is the registry provided by OpenShift Dedicated to manage images.

**registry authentication**

To push and pull images to and from private image repositories, the registry needs to authenticate its users with credentials.

**route**

Exposes a service to allow for network access to pods from users and applications outside the OpenShift Dedicated instance.
scale down
To decrease the number of replicas.
scale up
To increase the number of replicas.
service
A service exposes a running application on a set of pods.

1.2. INTEGRATED OPENSHIFT IMAGE REGISTRY

OpenShift Dedicated provides a built-in container image registry that runs as a standard workload on
the cluster. The registry is configured and managed by an infrastructure Operator. It provides an out-of
the-box solution for users to manage the images that run their workloads, and runs on top of the
existing cluster infrastructure. This registry can be scaled up or down like any other cluster workload and
does not require specific infrastructure provisioning. In addition, it is integrated into the cluster user
authentication and authorization system, which means that access to create and retrieve images is
controlled by defining user permissions on the image resources.

The registry is typically used as a publication target for images built on the cluster, as well as being a
source of images for workloads running on the cluster. When a new image is pushed to the registry, the
cluster is notified of the new image and other components can react to and consume the updated
image.

Image data is stored in two locations. The actual image data is stored in a configurable storage location,
such as cloud storage or a filesystem volume. The image metadata, which is exposed by the standard
cluster APIs and is used to perform access control, is stored as standard API resources, specifically
images and imagestreams.

Additional resources

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

1.3. THIRD-PARTY REGISTRIES

OpenShift Dedicated can create containers using images from third-party registries, but it is unlikely
that these registries offer the same image notification support as the integrated OpenShift image
registry. In this situation, OpenShift Dedicated will fetch tags from the remote registry upon
imagstream creation. To refresh the fetched tags, run `oc import-image <stream>`. When new images
are detected, the previously described build and deployment reactions occur.

1.3.1. Authentication

OpenShift Dedicated can communicate with registries to access private image repositories using
credentials supplied by the user. This allows OpenShift Dedicated to push and pull images to and from
private repositories.

1.3.1.1. Registry authentication with Podman

Some container image registries require access authorization. Podman is an open source tool for
managing containers and container images and interacting with image registries. You can use Podman
to authenticate your credentials, pull the registry image, and store local images in a local file system. The
following is a generic example of authenticating the registry with Podman.
Procedure

1. Use the Red Hat Ecosystem Catalog to search for specific container images from the Red Hat Repository and select the required image.

2. Click Get this image to find the command for your container image.

3. Log in by running the following command and entering your username and password to authenticate:

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

4. Download the image and save it locally by running the following command:

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

1.4. RED HAT QUAY REGISTRIES

If you need an enterprise-quality container image registry, Red Hat Quay is available both as a hosted service and as software you can install in your own data center or cloud environment. Advanced features in Red Hat Quay include geo-replication, image scanning, and the ability to roll back images.

Visit the Quay.io site to set up your own hosted Quay registry account. After that, follow the Quay Tutorial to log in to the Quay registry and start managing your images.

You can access your Red Hat Quay registry from OpenShift Dedicated like any remote container image registry.

Additional resources

- Red Hat Quay product documentation

1.5. AUTHENTICATION ENABLED RED HAT REGISTRY

All container images available through the Container images section of the Red Hat Ecosystem Catalog are hosted on an image registry, registry.redhat.io.

The registry, registry.redhat.io, requires authentication for access to images and hosted content on OpenShift Dedicated. Following the move to the new registry, the existing registry will be available for a period of time.

**NOTE**

OpenShift Dedicated pulls images from registry.redhat.io, so you must configure your cluster to use it.

The new registry uses standard OAuth mechanisms for authentication, with the following methods:

- **Authentication token.** Tokens, which are generated by administrators, are service accounts that give systems the ability to authenticate against the container image registry. Service accounts are not affected by changes in user accounts, so the token authentication method is reliable and resilient. This is the only supported authentication option for production clusters.
- **Web username and password.** This is the standard set of credentials you use to log in to resources such as access.redhat.com. While it is possible to use this authentication method with OpenShift Dedicated, it is not supported for production deployments. Restrict this authentication method to stand-alone projects outside OpenShift Dedicated.

You can use `podman login` with your credentials, either username and password or authentication token, to access content on the new registry.

All imagestreams point to the new registry, which uses the installation pull secret to authenticate.

You must place your credentials in either of the following places:

- **openshift namespace.** Your credentials must exist in the `openshift` namespace so that the imagestreams in the `openshift` namespace can import.

- **Your host.** Your credentials must exist on your host because Kubernetes uses the credentials from your host when it goes to pull images.

**Additional resources**

- [Registry service accounts](#)
CHAPTER 2. IMAGE REGISTRY OPERATOR IN OPENSHIFT DEDICATED

2.1. IMAGE REGISTRY ON OPENSHIFT DEDICATED

The Image Registry Operator installs a single instance of the OpenShift image registry, and manages all registry configuration, including setting up registry storage.

After the control plane deploys, the Operator creates a default configs.imageregistry.operator.openshift.io resource instance based on configuration detected in the cluster.

If insufficient information is available to define a complete configs.imageregistry.operator.openshift.io resource, the incomplete resource is defined and the Operator updates the resource status with information about what is missing.

The Image Registry Operator runs in the openshift-image-registry namespace, and manages the registry instance in that location as well. All configuration and workload resources for the registry reside in that namespace.
CHAPTER 3. ACCESSING THE REGISTRY

In OpenShift Dedicated, Red Hat Site Reliability Engineering (SRE) manages the registry for you. However, you can check the status of the registry pods and view the registry logs.

3.1. CHECKING THE STATUS OF THE REGISTRY PODS

As an administrator with the dedicated-admin role, you can list the image registry pods running in the openshift-image-registry project and check their status.

Prerequisites

- You have access to the cluster as a user with the dedicated-admin role.

Procedure

- List the pods in the openshift-image-registry project and view their status:

  $ oc get pods -n openshift-image-registry

Example output

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-image-registry-operator-764bd7f846-qqtpb</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>78m</td>
</tr>
<tr>
<td>image-registry-79fb44696-llrlin</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>77m</td>
</tr>
<tr>
<td>node-ca-hjksc</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>73m</td>
</tr>
<tr>
<td>node-ca-tftj6</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>77m</td>
</tr>
<tr>
<td>node-ca-wb6ht</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>77m</td>
</tr>
<tr>
<td>node-ca-zvt9q</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>74m</td>
</tr>
</tbody>
</table>

3.2. VIEWING REGISTRY LOGS

You can view the logs for the registry by using the oc logs command.

Procedure

- Use the oc logs command with deployments to view the logs for the container image registry:

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

Example output

```
```