



# Red Hat Quay 3.12

## 升级 Red Hat Quay

升级 Red Hat Quay





## 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

升级 Red Hat Quay

---

# 目录

<b>第 1 章 升级概述</b> .....	<b>3</b>
<b>第 2 章 升级 RED HAT QUAY OPERATOR 概述</b> .....	<b>4</b>
2.1. OPERATOR LIFECYCLE MANAGER .....	4
2.2. 升级 RED HAT QUAY OPERATOR .....	4
2.3. 升级 QUAYREGISTRY 资源 .....	7
2.4. 升级 QUAYECOSYSTEM .....	8
<b>第 3 章 独立升级</b> .....	<b>11</b>
3.1. 访问镜像 .....	12
3.2. 从 3.11.Z 升级到 3.12.Z .....	12
3.3. 从 3.10.Z 升级到 3.12.Z .....	13
3.4. 从 3.10.Z 升级到 3.11.Z .....	13
3.5. 从 3.9.Z 升级到 3.11.Z .....	13
<b>第 4 章 升级独立 RED HAT QUAY 的地理复制部署</b> .....	<b>15</b>
<b>第 5 章 在 OPENSIFT CONTAINER PLATFORM 上升级 RED HAT QUAY 的地理复制部署</b> .....	<b>19</b>
<b>第 6 章 升级 QUAY BRIDGE OPERATOR</b> .....	<b>23</b>
<b>第 7 章 降级 RED HAT QUAY</b> .....	<b>25</b>



---

## 第 1 章 升级概述

Red Hat Quay 的升级步骤取决于您使用的安装类型。

Red Hat Quay Operator 提供了部署和管理 Red Hat Quay 集群的简单方法。这是在 OpenShift 中部署 Red Hat Quay 的首选流程。

Red Hat Quay Operator 应该使用 [Operator Lifecycle Manager \(OLM\)](#) 升级，如“使用 Quay Operator 升级 Quay”部分中所述。

升级概念验证或高可用性 Red Hat Quay 和 Clair 安装的步骤记录在“独立升级”部分。

## 第 2 章 升级 RED HAT QUAY OPERATOR 概述

Red Hat Quay Operator 遵循一个同步版本方案，这意味着每个 Operator 版本都与 Red Hat Quay 的版本及其管理的组件相关联。**QuayRegistry** 自定义资源上没有设置 **要部署** Red Hat Quay 版本的字段。Operator 只能部署所有组件的单个版本。选择这个方案是为了确保所有组件都可以正常工作，并降低 Operator 的复杂性，了解如何管理 Kubernetes 上许多不同版本的 Red Hat Quay 版本。

### 2.1. OPERATOR LIFECYCLE MANAGER

应该使用 [Operator Lifecycle Manager \(OLM\)](#) 安装和升级 Red Hat Quay Operator。当使用默认 **approvalStrategy: Automatic** 创建订阅时，每当有新版本可用时，OLM 将自动升级 Red Hat Quay Operator。



#### 警告

当 Operator Lifecycle Manager 安装 Red Hat Quay Operator 时，可能会将其配置为支持自动或手动升级。这个选项会在安装过程中显示在 Red Hat Quay Operator 的 OperatorHub 页面中。它还可以通过 approvalStrategy 字段在 Red Hat Quay Operator Subscription 对象中找到。选择 Automatic 意味着，您的 Red Hat Quay Operator 会在发布新的 Operator 版本时自动升级。如果这不是需要的，则应选择 Manual 批准策略。

### 2.2. 升级 RED HAT QUAY OPERATOR

在 OpenShift Container Platform 中升级已安装的 Operator 的标准方法包括在 [升级安装的 Operator](#) 中。

通常，Red Hat Quay 仅支持从以前的(N-1)次版本进行升级。例如，不支持直接从 Red Hat Quay 3.0.5 升级到 3.5 的最新版本。相反，用户需要升级如下：

1. **3.0.5 → 3.1.3**
2. **3.1.3 → 3.2.2**
3. **3.2.2 → 3.3.4**



4. **3.3.4 → 3.4.z**

5. **3.4.z → 3.5.z**

这需要确保正确完成所有必要的数据库迁移，并在升级过程中按照正确顺序完成。

在某些情况下，Red Hat Quay 支持从以前的(N-2, N-3)次版本直接、单步升级。这简化了旧版本的客户的升级步骤。Red Hat Quay 3.12 支持以下升级路径：

- **3.10.z → 3.12.z**
- **3.11.z → 3.12.z**

有关 Red Hat Quay 的独立部署中的用户，希望升级到 3.12，请参阅 [独立](#) 升级指南。

### 2.2.1. 将 Red Hat Quay 升级到 3.12 版本

要将 Red Hat Quay 从一个次版本升级到下一个次版本，如 3.11 → 3.12，您必须更改 Red Hat Quay Operator 的更新频道。

#### 流程

1. 在 OpenShift Container Platform Web 控制台中导航至 **Operators → Installed Operators**。
2. 点 **Red Hat Quay Operator**。
3. 进入到 **Subscription** 选项卡。
4. 在 **Subscription details** 下，点 **Update channel**。

5. 选择 **stable-3.12** → **Save**。
6. 在 **Upgrade status** 下检查新安装的进度。在继续操作前，等待升级 状态变为 1。
7. 在 **OpenShift Container Platform** 集群中，进入到 **Workloads** → **Pods**。现有 pod 应该被终止，或正在终止。
8. 等待以下 pod（负责升级数据库和现有数据的模式迁移）到启动：**clair-postgres-upgrade**、**quay-postgres-upgrade**、**quay-postgres-upgrade** 和 **quay-app-upgrade**。
9. 在 **clair-postgres-upgrade**、**quay-postgres-upgrade** 和 **quay-app-upgrade** pod 标记为 **Completed** 后，您的 Red Hat Quay 部署剩余的 pod 会启动。这大约需要十分钟。
10. 验证 **quay-database** 和 **clair-postgres** pod 现在使用 **postgresql-13** 镜像。
11. 在 **quay-app** pod 标记为 **Running** 后，就可以访问 Red Hat Quay registry。

### 2.2.2. 升级到下一个次版本

对于 z 流升级，例如 3.11.1 → 3.11.2，用户在安装过程中最初选择的主次频道中会发布更新。执行 z 流升级的过程取决于上述 **approvalStrategy**。如果批准策略被设置为 **Automatic**，Red Hat Quay Operator 会自动升级到最新的 z 流。这会导致自动的、将 Red Hat Quay 更新部署到较新的 z 流，而无需停机。否则，必须在开始安装前手动批准更新。

### 2.2.3. 更改 Red Hat Quay Operator 的更新频道

已安装的 Operator 的订阅指定一个更新频道，用于跟踪和接收 Operator 的更新。要升级 Red Hat Quay Operator 以开始跟踪并从更新频道接收更新，请在安装的 Red Hat Quay Operator 的 **Subscription** 选项卡中更改更新频道。对于带有自动批准策略的订阅，升级会自动开始，并可在列出 **Installed Operators** 的页面中监控。

### 2.2.4. 手动批准待处理的 Operator 升级

如果已安装的 Operator 的订阅设置为 **Manual**，则当其当前更新频道中发布新的更新时，必须在开始安装前手动批准更新。如果 Red Hat Quay Operator 有一个待处理的升级，这个状态将显示在 **Installed Operators** 列表中。在 Red Hat Quay Operator 的 **Subscription** 选项卡中，您可以预览安装计划，并

查看列出可用于升级的资源。如果满意，点 **Approve** 并返回到列出 **Installed Operators** 的页面，以监控升级的进度。

下图显示了 UI 中的 **Subscription** 选项卡，包括 **更新频道**、**批准策略**、**升级状态** 和 **InstallPlan**：

Project: quay-enterprise

Installed Operators > Operator details

**Red Hat Quay**  
3.4.3 provided by Red Hat

Details YAML **Subscription** Events Quay Registry

### Subscription details

<b>Channel</b> quay-v3.4	<b>Approval</b> Automatic	<b>Upgrade status</b> Up to date 1 installed 0 installing
-----------------------------	------------------------------	--

**Name**  
quay-operator

**Namespace**  
quay-enterprise

**Labels**  
operators.coreos.com/quay-operator.quay-enterprise

**Created at**  
Mar 25, 12:17 pm

**Owner**  
No owner

**Installed version**  
quay-operator.v3.4.3

**Starting version**  
quay-operator.v3.4.3

**CatalogSource**  
redhat-operators **Healthy**

**InstallPlan**  
install-wf26n

**Installed Operators** 列表提供了当前 Quay 安装的高级概述：

Project: quay-enterprise

### Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#). Or create an Operator and ClusterServiceVersion using the [Operator SDK](#).

Name Search by name...

Name	Managed Namespaces	Status	Last updated	Provided APIs
<b>Red Hat Quay</b> 3.4.3 provided by Red Hat	quay-enterprise	Succeeded Up to date	Mar 25, 12:18 pm	Quay Registry

## 2.3. 升级 QUAYREGISTRY 资源

当 Red Hat Quay Operator 启动时，它会立即查找它要监视的命名空间中找到的任何 Quay Registries。当找到时，使用以下逻辑：

- 如果 `status.currentVersion` 未设置，请正常进行协调。
- 如果 `status.currentVersion` 与 Operator 版本相同，请正常进行协调。
- 如果 `status.currentVersion` 不等于 Operator 版本，请检查是否可以升级它。如果可以，执行升级任务，并在完成后将 `status.currentVersion` 设置为 Operator 的版本。如果无法升级，则返回错误，并只保留 QuayRegistry 及其部署的 Kubernetes 对象。

## 2.4. 升级 QUAYECOSYSTEM

从以前的 Operator 版本支持升级，这些 Operator 使用 QuayEcosystem API 进行一组有限的配置。为确保迁移不会意外发生，需要将一个特殊标签应用到 QuayEcosystem 以便它被迁移。将创建一个新的 QuayRegistry 以供 Operator 管理，但旧的 QuayEcosystem 将保留，直到手动删除为止，以确保您可以在任何错误时回滚并仍然访问 Quay。要将现有的 QuayEcosystem 迁移到新的 QuayRegistry，请使用以下流程：

### 流程

1. 将 `"quay-operator/migrate": "true"` 添加到 QuayEcosystem 的 `metadata.labels` 中。

```
$ oc edit quayecosystem <quayecosystemname>
```

```
metadata:
  labels:
    quay-operator/migrate: "true"
```
2. 等待使用与 QuayEcosystem 相同的 `metadata.name` 创建 QuayRegistry。QuayEcosystem 将标记为标签 `"quay-operator/migration-complete": "true"`。
3. 在设置了新 QuayRegistry 的 `status.registryEndpoint` 后，访问 Red Hat Quay 并确认所有数据和设置都已成功迁移。
4. 如果一切正常工作，您可以删除 QuayEcosystem 和 Kubernetes 垃圾回收清理所有旧资源。

### 2.4.1. 恢复 QuayEcosystem 升级

如果在从 QuayEcosystem 升级到 QuayRegistry 的过程中出现错误，请按照以下步骤恢复为使用 QuayEcosystem：

## 流程

1. 使用 UI 或 kubectl 删除 QuayRegistry：

```
$ kubectl delete -n <namespace> quayregistry <quayecosystem-name>
```

2. 如果使用 Route 提供外部访问，将 Route 更改为指回原始的 Service（使用 UI 或 kubectl）。



### 注意

如果您的 QuayEcosystem 管理 PostgreSQL 数据库，升级过程会将您的数据迁移到由升级 Operator 管理的新 PostgreSQL 数据库。您的旧数据库不会被更改或删除，但 Red Hat Quay 在迁移完成后不再使用它。如果数据迁移过程中出现问题，升级过程将退出，建议您继续将数据库作为非受管组件。

## 2.4.2. 支持的升级的 QuayEcosystem 配置

如果迁移 QuayEcosystem 组件失败或不受支持，Red Hat Quay Operator 会在日志中报告错误，并在 `status.conditions` 中报告错误。所有非受管组件都应成功迁移，因为不需要采用 Kubernetes 资源，并且 Red Hat Quay 的 `config.yaml` 文件中已提供了所有必要的值。

### 数据库

不支持临时数据库（必须设置 `volumeSize` 字段）。

### Redis

不需要特别要求。

### 外部访问

自动迁移只支持 **passthrough Route** 访问。其他方法所需的手动迁移。

- **没有自定义主机名的 LoadBalancer** : 当 QuayEcosystem 使用标签 "quay-operator/migration-complete": "true" 标记后, 在删除 QuayEcosystem 前需要从现存的 Service 中删除 metadata.ownerReferences 字段, 这可以防止 Kubernetes 对 Service 进行垃圾回收并删除负载均衡器。一个新的 Service 将被创建, 带有 metadata.name 格式 <QuayEcosystem-name>-quay-app。编辑现有 Service 的 spec.selector, 使其与新 Service 的 spec.selector 匹配, 以便进入旧负载均衡器端点的流量现在会被定向到新的 pod。旧的 Service 将会被您管理; Quay Operator 将不再管理它。
- **带有自定义主机名的 LoadBalancer/NodePort/Ingress** : 将创建一个类型为 LoadBalancer 的新的 Service, 带有 metadata.name 格式 <QuayEcosystem-name>-quay-app。将您的 DNS 设置更改为指向由新 Service 提供的 status.loadBalancer 端点。

## Clair

不需要特别要求。

## 对象存储

QuayEcosystem 没有受管对象存储组件, 因此对象存储始终标记为非受管。不支持本地存储。

## 仓库镜像

不需要特别要求。

## 第 3 章 独立升级

通常，Red Hat Quay 仅支持从以前的(N-1)次版本进行升级。例如，不支持直接从 Red Hat Quay 3.0.5 升级到 3.5 的最新版本。相反，用户需要升级如下：

1. **3.0.5 → 3.1.3**
2. **3.1.3 → 3.2.2**
3. **3.2.2 → 3.3.4**
4. **3.3.4 → 3.4.z**
5. **3.4.z → 3.5.z**

这需要确保正确完成所有必要的数据库迁移，并在升级过程中按照正确顺序完成。

在某些情况下，Red Hat Quay 支持从以前的(N-2, N-3)次版本直接、单步升级。这个例外是，只有普通次版本的升级，升级会简化旧版本客户的升级步骤。Red Hat Quay 3.12 支持以下升级路径：

- **3.10.z → 3.12.z**
- **3.11.z → 3.12.z**

有关希望升级 Red Hat Quay Operator 的用户，[请参阅升级 Red Hat Quay Operator 概述](#)。

本文档描述了执行每个独立升级所需的步骤。确定您的当前版本，然后按顺序执行这些步骤，从您的当前版本开始，并遵循您所需的目標版本。

- [从 3.11.z 升级到 3.12.z](#)

- [从 3.10.z 升级到 3.12.z](#)
- [从 3.10.z 升级到 3.11.z](#)
- [从 3.9.z 升级到 3.11.z](#)

有关独立版本的功能信息，请参阅 [Red Hat Quay 发行注记](#)。

手动升级的一般步骤包括以下步骤：

1. 停止 Quay 和 Clair 容器。
2. 备份数据库和镜像存储（可选但推荐）。
3. 使用镜像的新版本启动 Clair。
4. 等待 Clair 准备好接受连接，然后再启动新版本的 Quay。

### 3.1. 访问镜像

来自版本 3.4.0 及更新版本的 Red Hat Quay 镜像包括在 [registry.redhat.io](https://registry.redhat.io) 和 [registry.access.redhat.com](https://registry.access.redhat.com) 中，设置身份验证，如 [Red Hat Container Registry Authentication](#) 所述。

### 3.2. 从 3.11.Z 升级到 3.12.Z

#### 3.2.1. 目标镜像

- quay : [registry.redhat.io/quay/quay-rhel8:v3.12.3](https://registry.redhat.io/quay/quay-rhel8:v3.12.3)
- Clair: [registry.redhat.io/quay/clair-rhel8:v3.12.3](https://registry.redhat.io/quay/clair-rhel8:v3.12.3)



- PostgreSQL: registry.redhat.io/rhel8/postgresql-13:1-109
- redis : registry.redhat.io/rhel8/redis-6:1-110

### 3.3. 从 3.10.Z 升级到 3.12.Z

#### 3.3.1. 目标镜像

- quay : registry.redhat.io/quay/quay-rhel8:v3.12.3
- Clair: registry.redhat.io/quay/clair-rhel8:v3.12.3
- PostgreSQL: registry.redhat.io/rhel8/postgresql-13:1-109
- redis : registry.redhat.io/rhel8/redis-6:1-110

### 3.4. 从 3.10.Z 升级到 3.11.Z

#### 3.4.1. 目标镜像

- quay : registry.redhat.io/quay/quay-rhel8:v3.11.0
- Clair: registry.redhat.io/quay/clair-rhel8:v3.11.0
- PostgreSQL: registry.redhat.io/rhel8/postgresql-13:1-109
- redis : registry.redhat.io/rhel8/redis-6:1-110

### 3.5. 从 3.9.Z 升级到 3.11.Z

#### 3.5.1. 目标镜像

- **quay : registry.redhat.io/quay/quay-rhel8:v3.11.0**
- **Clair: registry.redhat.io/quay/clair-rhel8::v3.11.0**
- **PostgreSQL: registry.redhat.io/rhel8/postgresql-13:1-109**
- **redis : registry.redhat.io/rhel8/redis-6:1-110**

## 第 4 章 升级独立 RED HAT QUAY 的地理复制部署

使用以下步骤升级您的地理复制 Red Hat Quay 部署。



## 重要

- 将 geo-replication Red Hat Quay 部署升级到下一个 y-stream 版本（例如，Red Hat Quay 3.7 → Red Hat Quay 3.8）或 geo-replication 部署，您必须在升级前停止操作。
- 间歇性停机时间从一个 y-stream 版本升级到下一个版本。
- 强烈建议您在升级前备份 Red Hat Quay 部署。

## 先决条件

- 已登陆到 registry.redhat.io



## 流程

此流程假设您在三个（或更多）系统上运行 Red Hat Quay 服务。如需更多信息，[请参阅准备 Red Hat Quay 高可用性](#)。

1. 获取运行 Red Hat Quay 实例的每个系统上所有 Red Hat Quay 实例的列表。
  - a. 在 System A 中输入以下命令以显示 Red Hat Quay 实例：

```
$ sudo podman ps
```

## 输出示例

```
CONTAINER ID IMAGE COMMAND CREATED
STATUS PORTS NAMES
ec16ece208c0 registry.redhat.io/quay/quay-rhel8:v3.7.0 registry 6 minutes
ago Up 6 minutes ago 0.0.0.0:80->8080/tcp, 0.0.0.0:443->8443/tcp quay01
```

- b. 在 System B 中输入以下命令以显示 Red Hat Quay 实例：

```
$ sudo podman ps
```

输出示例

```
CONTAINER ID IMAGE COMMAND CREATED
STATUS PORTS NAMES
7ae0c9a8b37d registry.redhat.io/quay/quay-rhel8:v3.7.0 registry 5 minutes
ago Up 2 seconds ago 0.0.0.0:82->8080/tcp, 0.0.0.0:445->8443/tcp quay02
```

- c. 在 System C 中输入以下命令以显示 Red Hat Quay 实例：

```
$ sudo podman ps
```

输出示例

```
CONTAINER ID IMAGE COMMAND CREATED
STATUS PORTS NAMES
e75c4aebfee9 registry.redhat.io/quay/quay-rhel8:v3.7.0 registry 4 seconds
ago Up 4 seconds ago 0.0.0.0:84->8080/tcp, 0.0.0.0:447->8443/tcp quay03
```

2. 临时关闭每个系统上的所有 Red Hat Quay 实例。

- a. 在 System A 中输入以下命令来关闭 Red Hat Quay 实例：

```
$ sudo podman stop ec16ece208c0
```

- b. 在 System B 中输入以下命令来关闭 Red Hat Quay 实例：

```
$ sudo podman stop 7ae0c9a8b37d
```

- c. 在 System C 中输入以下命令来关闭 Red Hat Quay 实例：

```
$ sudo podman stop e75c4aebfee9
```

3. 获取最新的 Red Hat Quay 版本，例如：每个系统上的 Red Hat Quay 3.12。

- a. 在 System A 中输入以下命令以获取最新的 Red Hat Quay 版本：

```
$ sudo podman pull registry.redhat.io/quay/quay-rhel8:v3.8.0
```

- b. 在 System B 中输入以下命令以获取最新的 Red Hat Quay 版本：

```
$ sudo podman pull registry.redhat.io/quay/quay-rhel8:v3.8.0
```

- c. 在 System C 中输入以下命令以获取最新的 Red Hat Quay 版本：

```
$ sudo podman pull registry.redhat.io/quay/quay-rhel8:v3.8.0
```

4. 在高可用性 Red Hat Quay 部署的 System A 中，运行新镜像版本，如 Red Hat Quay 3.12：

```
# sudo podman run --restart=always -p 443:8443 -p 80:8080 \  
--sysctl net.core.somaxconn=4096 \  
--name=quay01 \  
-v /mnt/quay/config:/conf/stack:Z \  
-v /mnt/quay/storage:/datastorage:Z \  
-d registry.redhat.io/quay/quay-rhel8:v3.8.0
```

5. 等待新的 Red Hat Quay 容器在 System A 上完全正常工作。您可以输入以下命令来检查容器的状态：

```
$ sudo podman ps
```

输出示例

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
70b9f38c3fb4	registry.redhat.io/quay/quay-rhel8:v3.8.0	registry	2 seconds ago	Up 2 seconds ago
	0.0.0.0:82->8080/tcp, 0.0.0.0:445->8443/tcp	quay01		

6.

可选：通过进入到 Red Hat Quay UI 来确保 Red Hat Quay 的完全操作。

7.

确保 System A 上的 Red Hat Quay 完全正常工作后，在 System B 和 System C 上运行新镜像版本。

a.

在高可用性 Red Hat Quay 部署的 System B 中，运行新镜像版本，如 Red Hat Quay 3.12：

```
# sudo podman run --restart=always -p 443:8443 -p 80:8080 \
--sysctl net.core.somaxconn=4096 \
--name=quay02 \
-v /mnt/quay/config:/conf/stack:Z \
-v /mnt/quay/storage:/datastorage:Z \
-d registry.redhat.io/quay/quay-rhel8:v3.8.0
```

b.

在高可用性 Red Hat Quay 部署的 System C 中，运行新镜像版本，如 Red Hat Quay 3.12：

```
# sudo podman run --restart=always -p 443:8443 -p 80:8080 \
--sysctl net.core.somaxconn=4096 \
--name=quay03 \
-v /mnt/quay/config:/conf/stack:Z \
-v /mnt/quay/storage:/datastorage:Z \
-d registry.redhat.io/quay/quay-rhel8:v3.8.0
```

8.

您可以输入以下命令来检查 System B 和 System C 上的容器状态：

```
$ sudo podman ps
```

## 第 5 章 在 OPENSIFT CONTAINER PLATFORM 上升级 RED HAT QUAY 的地理复制部署

使用以下步骤升级 OpenShift Container Platform 部署中的地理复制 Red Hat Quay。



### 重要

- 当 OpenShift Container Platform 上的 geo-replicated Red Hat Quay 升级到下一个 y-stream 版本（例如，Red Hat Quay 3.7 → Red Hat Quay 3.8）时，您必须在升级前停止操作。
- 间歇性停机时间从一个 y-stream 版本升级到下一个版本。
- 在升级前，强烈建议在 OpenShift Container Platform 部署中备份 Red Hat Quay。



### 流程

此流程假设您在三个或更多系统上运行 Red Hat Quay registry。在此过程中，我们将假设三个系统名为 System A、System B 和 System C。系统 A 将作为部署 Red Hat Quay Operator 的主要系统。

1.

在 System B 和 System C 中，缩减 Red Hat Quay registry。这可以通过禁用自动扩展并覆盖 Red Hat Quay、mirror worker 和 Clair 的副本计数（如果被管理）。使用以下 quayregistry.yaml 文件作为参考：

```
apiVersion: quay.redhat.com/v1
kind: QuayRegistry
metadata:
  name: registry
  namespace: ns
spec:
  components:
    ...
    - kind: horizontalpodautoscaler
      managed: false 1
    - kind: quay
      managed: true
      overrides: 2
        replicas: 0
    - kind: clair
      managed: true
      overrides:
```

```

replicas: 0
- kind: mirror
  managed: true
  overrides:
    replicas: 0
...

```

1

禁用 Quay、Clair 和 Mirroring worker 的自动扩展

2

对于访问数据库和 objectstorage 的组件，将副本数设置为 0。



注意

您必须保持在 System A 上运行 Red Hat Quay registry。不要更新 System A 上的 quayregistry.yaml 文件。

2.

等待 registry-quay-app、registry-quay-mirror 和 registry-clair-app pod 消失。输入以下命令检查其状态：

```
oc get pods -n <quay-namespace>
```

输出示例

```

quay-operator.v3.7.1-6f9d859bd-p5ftc      1/1   Running   0          12m
quayregistry-clair-postgres-7487f5bd86-xnxpr  1/1   Running   1 (12m ago) 12m
quayregistry-quay-app-upgrade-xq2v6      0/1   Completed 0          12m
quayregistry-quay-redis-84f888776f-hhgms   1/1   Running   0          12m

```

3.

在 System A 中，启动 Red Hat Quay 升级到最新的 y-stream 版本。这是一个手动过程。有关升级安装的 Operator 的更多信息，请参阅 [升级已安装的 Operator](#)。有关 Red Hat Quay 升级路径的更多信息，请参阅 [升级 Red Hat Quay Operator](#)。

4.

安装新的 Red Hat Quay registry 后，集群上的必要升级会自动完成。之后，新的 Red Hat Quay Pod 使用最新的 y-stream 版本启动。另外，新的 Quay pod 会被调度并启动。



5. 进入 Red Hat Quay UI 确认更新是否正常工作：

- a. 在 OpenShift 控制台中，导航到 Operators → Installed Operators，然后单击 Registry Endpoint 链接。



#### 重要

在 Red Hat Quay UI 可用前，不要执行以下步骤。在 System A 上，不要升级 System B 上的 Red Hat Quay registry 和 System C。

6. 确认更新已在 System A 上正常工作，在 System B 和 System C 上启动 Red Hat Quay 升级。Operator 升级会导致升级 Red Hat Quay 安装，pod 被重启。



#### 注意

因为数据库模式对于新的 y-stream 安装是正确的，所以 System B 和 System C 上的新 pod 应该快速启动。

7. 更新后，通过删除组件的覆盖 来恢复此流程中第 1 步中所做的更改。例如：

```

apiVersion: quay.redhat.com/v1
kind: QuayRegistry
metadata:
  name: registry
  namespace: ns
spec:
  components:
    ...
    - kind: horizontalpodautoscaler
      managed: true 1
    - kind: quay
      managed: true
    - kind: clair
      managed: true
    - kind: mirror
      managed: true
    ...

```

**1**



## 第 6 章 升级 QUAY BRIDGE OPERATOR

要升级 Quay Bridge Operator (QBO), 请将 Subscription 选项卡中的 Channel Subscription 更新频道改为所需的频道。

当将 QBO 从版本 3.5 升级到 3.7 时, 需要一些额外的步骤 :

1. 您需要创建一个新的 QuayIntegration 自定义资源。这可以在 Web 控制台中, 或者从命令行完成。

upgrade-quay-integration.yaml

```
- apiVersion: quay.redhat.com/v1
  kind: QuayIntegration
  metadata:
    name: example-quayintegration-new
  spec:
    clusterID: openshift 1
    credentialsSecret:
      name: quay-integration
      namespace: openshift-operators
    insecureRegistry: false
    quayHostname: https://registry-quay-quay35.router-
      default.apps.cluster.openshift.com
```

**1**

确保 clusterID 与现有 QuayIntegration 资源的值匹配。

2. 创建新的 QuayIntegration 自定义资源 :

```
$ oc create -f upgrade-quay-integration.yaml
```

3. 删除旧的 QuayIntegration 自定义资源。

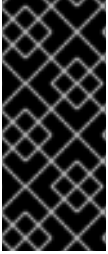
4.

删除旧的 mutatingwebhookconfigurations :

```
$ oc delete mutatingwebhookconfigurations.admissionregistration.k8s.io quay-bridge-operator
```

## 第 7 章 降级 RED HAT QUAY

Red Hat Quay 只支持回滚到以前的 z-stream 版本，如 3.7.2 → 3.7.1。不支持回滚到之前的 y-stream 版本(3.7.0 → 3.6.0)。这是因为 Red Hat Quay 更新可能包含升级到 Red Hat Quay 的新版本时应用的数据库架构升级。数据库架构升级不被视为向后兼容。



### 重要

基于 Operator 的部署或基于虚拟机的部署不支持降级到以前的 z-streams。只有在极端情况下才应进行降级。回滚 Red Hat Quay 部署的决定必须与 Red Hat Quay 支持和开发团队结合使用。如需更多信息，请联系红帽 Quay 支持。