OpenShift Container Platform 4.17

安装概述

安装 OpenShift Container Platform 的内容概述

Last Updated: 2025-11-25

OpenShift Container Platform 4.17 安装概述

安装 OpenShift Container Platform 的内容概述

Legal Notice

Abstract

本文档概述了如何安装 OpenShift Container Platform。

Table of Contents

第 1 章 OPENSHIFT CONTAINER PLATFORM 安装概述

1.1. 关于 OPENSHIFT CONTAINER PLATFORM 安装

OpenShift Container Platform 安装程序提供了四个部署集群的方法，相关信息包括在以下列表中：

- **交互式** ：您可以使用基于 Web 的 辅助安装程序（Assisted Installer）部署集群。对于可以连接到互联网的网络，这是一个理想的方法。Assisted Installer 是安装 OpenShift Container Platform 的最简单方法，它提供智能默认值，并在安装集群前执行预动态验证。它还提供了一个 RESTful API 用于自动化和高级配置场景。

- **本地基于代理的** ：对于断开连接的环境或有网络限制的环境，您可以使用基于代理的安装程序。它提供了 Assisted Installer 的许多优点，但您必须首先下载并配置基于代理的安装程序。使用命令行界面完成配置。这个方法适用于断开连接的环境。

- **自动** ：您可以在安装程序置备的基础架构中部署集群。安装程序使用每个集群主机的基板管理控制器 (BMC) 进行置备。您可以在有连接或断开连接的环境中部署集群。

- **完全控制**：您可以在自己准备和维护的基础架构上部署集群，这种方法提供了最大的定制性。您可以在有连接或断开连接的环境中部署集群。

每种方法部署的集群具有以下特征：

- 没有单点故障的高可用性基础架构，默认可用。

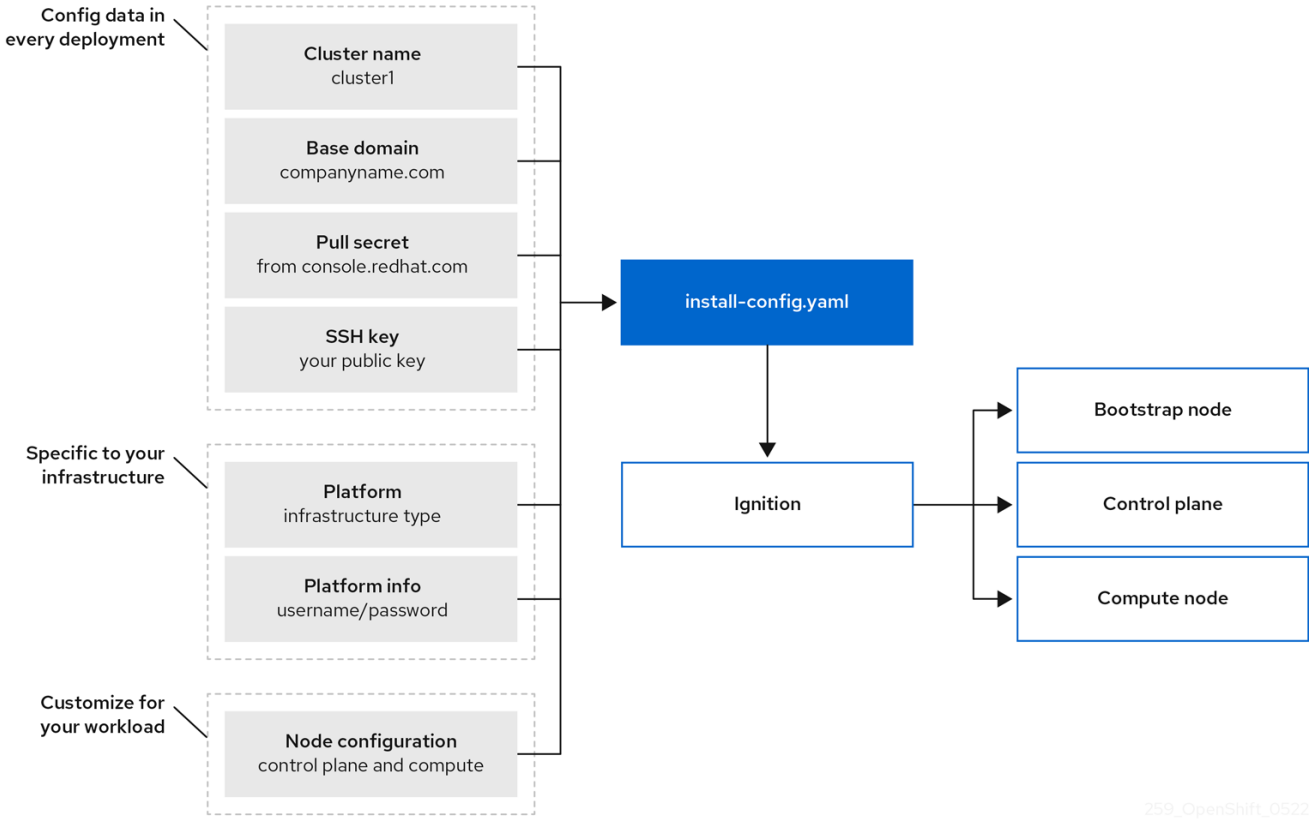
- 管理员可以控制要应用的更新，以及应用的时间。

1.1.1. 关于安装程序

您可以使用安装程序部署每种集群。安装程序会生成主要资产，如 bootstrap、control plane 和计算机器的 Ignition 配置文件。您可以使用这三个机器配置开始使用 OpenShift Container Platform 集群，它为您提供了正确配置的基础架构。

OpenShift Container Platform 安装程序使用一组目标和依赖项来管理集群安装。安装程序具有一组必须实现的目标，并且每个目标都有一组依赖项。因为每个目标仅关注其自己的依赖项，所以安装程序可以并行地实现多个目标，最终组成一个正常运行的集群。安装程序会识别并使用现有组件，而不是运行命令来再次创建它们，因为程序满足依赖项。

图 1.1. OpenShift Container Platform 安装目标和依赖项



1.1.2. 关于 Red Hat Enterprise Linux CoreOS (RHCOS)

在安装后，每一个集群机器都将使用 Red Hat Enterprise Linux CoreOS (RHCOS) 作为操作系统。RHCOS 是 Red Hat Enterprise Linux (RHEL) 的不可变容器主机版本，具有默认启用 SELinux 的 RHEL 内核。RHCOS 包括作为 Kubernetes 节点代理的 **kubelet**，以及为 Kubernetes 优化的 CRI-O 容器运行时。

OpenShift Container Platform 4.17 集群中的每一 control plane 机器都必须使用 RHCOS，其中包括一个关键的首次启动置备工具，称为 Ignition。这一工具让集群能够配置机器。操作系统更新作为可引导容器镜像（使用 **OSTree** 作为后端）提供，该镜像由 Machine Config Operator 在集群中部署。实际的操作系统更改通过使用 **rpm-ostree** 在每台机器上作为原子操作原位进行。通过结合使用这些技术，OpenShift Container Platform 可以像管理集群上的任何其他应用程序一样管理操作系统，通过原位升级使整个平台保持最新状态。这些原位更新可以减轻运维团队的负担。

如果将 RHCOS 用作所有集群机器的操作系统，则集群将管理其组件和机器的所有方面，包括操作系统在内。因此，只有安装程序和 Machine Config Operator 才能更改机器。安装程序使用 Ignition 配置文件设置每台机器的确切状态，安装后则由 Machine Config Operator 完成对机器的更多更改，例如应用新证书或密钥等。

1.1.3. OpenShift Container Platform 安装的常见术语表

术语表定义了与安装内容相关的常用术语。参阅以下术语列表以更好地了解安装过程。

支持的安装程序

在 console.redhat.com 中托管的安装程序，它提供基于 web 用户界面或 RESTful API 用于创建集群配置。Assisted Installer 会生成一个发现镜像。集群机器使用发现镜像进行引导，它会安装 RHCOS 和代理。Assisted Installer 和代理一起为集群提供了预安装验证和安装功能。

基于代理的安装程序

与 Assisted Installer 类似的安装程序，但您必须首先下载基于代理的安装程序。基于代理的安装程序是断开连接的环境的理想选择。

Bootstrap 节点

一个临时的机器，它运行最小需要的 Kubernetes 配置来部署 OpenShift Container Platform 控制平面（control plane）。

Control plane（控制平面）

一个容器编配层，用于公开 API 和接口来定义、部署和管理容器的生命周期。也称为 control plane 机器。

Compute 节点

负责执行集群用户工作负载的节点。也称为 worker 节点。

断开连接的安装

在有些情况下，数据中心的部分环境可能无法访问互联网，甚至无法通过代理服务器访问。您仍可在这些环境中安装 OpenShift Container Platform，但需要先下载所需的软件和镜像，并将其保存在离线环境中。

OpenShift Container Platform 安装程序

置备基础架构并部署集群的程序。

安装程序置备的基础架构

安装程序部署并配置运行集群的基础架构。

Ignition 配置文件

Ignition 工具用于在操作系统初始化过程中配置 Red Hat Enterprise Linux CoreOS (RHCOS)的文件。安装程序生成不同的 Ignition 配置文件来初始化 bootstrap、control plane 和 worker 节点。

Kubernetes 清单

JSON 或 YAML 格式的 Kubernetes API 对象的规格。配置文件可以包含部署、配置映射、secret 和 daemonset 等。

Kubelet

在集群的每个节点上运行的一个主节点代理，以确保容器在 pod 中运行。

负载均衡器

负载均衡器是客户端的单点联系。API 的负载均衡器在 control plane 节点之间分布传入的流量。

Machine Config Operator

一个 Operator，管理并应用基本操作系统和容器运行时的配置和更新，包括内核和 kubelet 之间的所有配置和更新。

Operator

在 OpenShift Container Platform 集群中打包、部署和管理 Kubernetes 应用程序的首选方法。Operator 将人类操作知识编码到一个软件程序中，易于打包并与客户共享。

用户置备的基础架构

您可以在自己提供的基础架构上安装 OpenShift Container Platform。您可以使用安装程序来生成置备集群基础架构所需的资产，再创建集群基础架构，然后将集群部署到您提供的基础架构中。

1.1.4. 安装过程

除了 Assisted Installer 外，当安装 OpenShift Container Platform 集群时，您必须从 OpenShift Cluster Manager Hybrid Cloud Console 上的适当的 **Cluster Type** 页面下载安装程序。此控制台管理：

- 帐户的 REST API。
- registry 令牌，这是用于获取所需组件的 pull secret。

- 集群注册，将集群身份与您的红帽帐户相关联，以便收集使用指标。

在 OpenShift Container Platform 4.17 中，安装程序是对一组资产执行一系列文件转换的 Go 二进制文件。与安装程序交互的方式因您的安装类型而异。考虑以下安装用例：

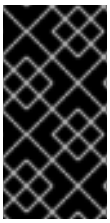
- 要使用 Assisted Installer 部署集群，您可以使用 Assisted Installer 配置集群设置。没有安装程序可以下载和配置。设置完集群配置后，您可以下载发现 ISO，然后使用该镜像引导集群机器。您可以使用 Assisted Installer 在完全集成的 Nutanix、vSphere 和裸机上安装集群，以及以前没有集成的环境中安装集群。如果在裸机上安装，您需要提供所有集群基础架构和资源，包括网络、负载均衡、存储和所有集群机器。

- 要使用基于代理的安装程序部署集群，您可以首先下载基于代理的安装程序。然后，您可以配置集群并生成发现镜像。您可以使用发现镜像引导集群机器，它会安装一个与安装程序进行通信的代理，并为您处理置备，您不需要与安装程序进行交互或自行设置置备程序机器。您需要提供所有集群基础架构和资源，包括网络、负载均衡、存储和单个集群机器。这个方法适用于断开连接的环境。

- 对于具有安装程序置备的基础架构集群，您可以将基础架构启动和置备委派给安装程序，而不是亲自执行。安装程序将创建支持集群所需的所有网络、机器和操作系统，除非您裁裸机上安装。如果在裸机上安装，您必须提供所有集群基础架构和资源，包括 bootstrap 机器、网络、负载均衡、存储和单个集群机器。

- 如果亲自为集群置备和管理基础架构，则必须提供所有集群基础架构和资源，包括 Bootstrap 机器、网络、负载均衡、存储和独立的集群机器。

对于安装程序，在安装过程中会使用三组文件：名为 **install-config.yaml** 的安装配置文件、Kubernetes 清单，以及您的集群类型的 Ignition 配置文件。



重要

在安装过程中，您可以修改控制基础 RHCOS 操作系统的 Kubernetes 和 Ignition 配置文件。但是，没有可用的验证机制来确认您对这些对象所做修改是适当的。如果修改了这些对象，集群可能会无法运行。由于存在这种风险，修改 Kubernetes 和 Ignition 配置文件不受支持，除非您遵循记录的流程或在红帽支持指示下操作。

安装配置文件转换为 Kubernetes 清单，然后清单嵌套到 Ignition 配置文件中。安装程序使用这些 Ignition 配置文件来创建集群。

运行安装程序时，所有配置文件会被修剪，因此请务必备份需要再次使用的所有配置文件。

重要

安装之后，您无法修改在安装过程中设置的参数，但可以修改一些集群属性。

使用辅助安装程序的安装过程

使用辅助安装程序进行安装涉及使用基于 Web 的用户界面或使用 RESTful API 以互动方式创建集群配置。Assisted Installer 用户界面会提示您输入所需的值，并为其余参数提供合理的默认值，除非在用户界面或使用 API 中更改它们。Assisted Installer 生成发现镜像，您可以下载并用来引导集群机器。镜像安装 RHCOS 和代理，代理会为您处理置备。您可以使用 Assisted Installer 安装 OpenShift Container Platform，并在 Nutanix、vSphere 和裸机上完全集成。另外，您可以在其他没有集成的情况下使用 Assisted Installer 安装 OpenShift Container Platform。

OpenShift Container Platform 管理集群的所有方面，包括操作系统本身。每台机器在启动时使用的配置引用其加入的集群中托管的资源。此配置允许集群在应用更新时自行管理。

如果可能，请使用 Assisted Installer 功能来避免下载和配置基于代理的安装程序。

基于代理的基础架构的安装过程
基于代理的安装与使用 Assisted Installer 类似，唯一的不同是需要在最初下载并安装基于代理的安装程序。当您希望利用 Assisted Installer 所带来的变量，并需要在断开连接的环境中安装集群时，可以使用基于代理的安装。

如果可能，请使用基于代理的安装功能来避免创建带有 bootstrap 虚拟机的置备程序机器，然后置备和维护集群基础架构。

采用安装程序置备的基础架构的安装过程
默认安装类型为使用安装程序置备的基础架构。默认情况下，安装程序充当安装向导，提示您输入它无法自行确定的值，并为其余参数提供合理的默认值。您还可以自定义安装过程来支持高级基础架构场景。安装程序将为集群置备底层基础架构。

您可以安装标准集群或自定义集群。对于标准集群，您要提供安装集群所需的最低限度详细信息。对于自定义集群，您可以指定有关平台的更多详细信息，如 control plane 使用的机器数量、集群部署的虚拟机的类型，或 Kubernetes 服务网络的 CIDR 范围。

若有可能，可以使用此功能来避免置备和维护集群基础架构。在所有其他环境中，可以使用安装程序来生成置备集群基础架构所需的资产。

对于安装程序置备的基础架构的集群，OpenShift Container Platform 可以管理集群的所有方面，包括操作系统本身。每台机器在启动时使用的配置引用其加入的集群中托管的资源。此配置允许集群在应用更新时自行管理。

采用用户置备的基础架构的安装过程
您还可以在自己提供的基础架构上安装 OpenShift Container Platform。您可以使用安装程序来生成置备集群基础架构所需的资产，再创建集群基础架构，然后将集群部署到您提供的基础架构中。

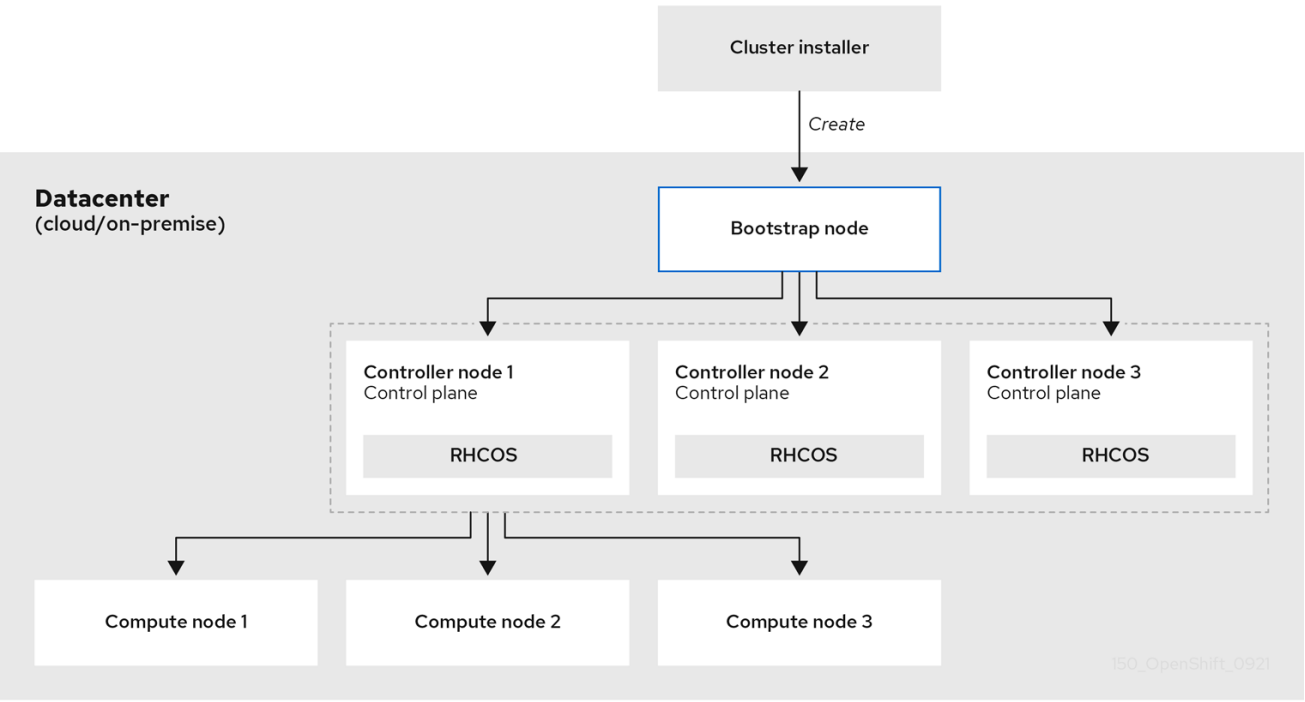
如果不使用安装程序置备的基础架构，您必须自己管理和维护集群资源。以下列表详细介绍了其中一些自我管理的资源：

- 组成集群的 control plane 和计算机器的底层基础架构
- 负载均衡器
- 集群网络，包括 DNS 记录和所需的子网
- 集群基础架构和应用程序的存储

如果您的集群使用用户置备的基础架构，您可以选择将 RHEL 计算机器添加到集群中。

安装过程详细信息
置备集群时，集群中的每台机器都需要有关集群的信息。OpenShift Container Platform 在初始配置过程中使用临时 bootstrap 机器为永久 control plane 提供所需的信息。临时 bootstrap 机器使用一个带有描述如何创建集群的 Ignition 配置文件进行引导。bootstrap 机器创建组成控制平面（control plane）的 control plane 机器。然后，control plane 机器创建计算（compute）机器。下图说明了这一过程：

图 1.2. 创建 bootstrap、control plane 和计算机器



集群机器初始化后，Bootstrap 机器将被销毁。所有集群都使用 Bootstrap 过程来初始化集群，但若您自己置备集群的基础架构，则必须手动完成许多步骤。

重要

- 安装程序生成的 Ignition 配置文件包含在 24 小时后过期的证书，然后在过期时进行续订。如果在更新证书前关闭集群，且集群在 24 小时后重启，集群会自动恢复过期的证书。一个例外是，您必须手动批准待处理的 **node-bootstrapper** 证书签名请求(CSR)来恢复 kubelet 证书。如需更多信息，*请参阅从过期的 control plane 证书* 中恢复的文档。
- 建议您在 Ignition 配置文件生成后的 12 小时内使用它们，因为 24 小时的证书会在集群安装后的 16 小时到 22 小时间进行轮转。通过在 12 小时内使用 Ignition 配置文件，您可以避免在安装过程中因为执行了证书更新而导致安装失败的问题。

bootstrapp 集群涉及以下步骤：

1. bootstrap 机器启动并开始托管 control plane 机器引导所需的远程资源。如果您置备基础架构，此步骤需要人工干预。
2. bootstrap 机器启动单节点 etcd 集群和一个临时 Kubernetes control plane。
3. control plane 机器从 bootstrap 机器获取远程资源并完成启动。如果您置备基础架构，此步骤需要人工干预。
4. 临时 control plane 将生产环境的 control plane 调度到生产环境 control plane 机器。
5. Cluster Version Operator（CVO）在线并安装 etcd Operator。etcd Operator 在所有 control plane 节点上扩展 etcd。
6. 临时 control plane 关机，并将控制权交给生产环境 control plane。

7. bootstrap 机器将 OpenShift Container Platform 组件注入生产环境 control plane。
8. 安装程序关闭 bootstrap 机器。如果您置备基础架构，此步骤需要人工干预。
9. control plane 设置计算节点。
10. control plane 以一组 Operator 的形式安装其他服务。

完成此 bootstrap 过程后，将生成一个全面运作的 OpenShift Container Platform 集群。然后，集群会下载并配置日常运作所需的其余组件，包括在受支持的环境中创建计算（compute）机器。

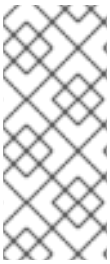
其他资源

- Red Hat OpenShift Network Calculator

1.1.5. 安装后验证节点状态

当以下安装健康检查成功时，OpenShift Container Platform 安装会完成：

- 置备程序可以访问 OpenShift Container Platform Web 控制台。
- 所有 control plane 节点都已就绪。
- 所有集群 Operator 都可用。



注意

安装完成后，负责 worker 节点的特定集群 Operator 持续尝试置备所有 worker 节点。在所有 worker 节点都报告为 **READY** 之前需要一段时间。对于在裸机上的安装，请等待至少 60 分钟，然后对 worker 节点进行故障排除。对于所有其他平台上安装，请等待至少 40 分钟后再对 worker 节点进行故障排除。负责 worker 节点的集群 Operator 的 **DEGRADED** 状态取决于 Operator 自己的资源，而不是节点的状态。

安装完成后，您可以继续监控集群中的节点条件。

先决条件

- 安装程序在终端中成功解决。

流程

1. 显示所有 worker 节点的状态：

```
$ oc get nodes
```

输出示例

```
NAME                         STATUS   ROLES    AGE   VERSION
example-compute1.example.com   Ready    worker   13m   v1.21.6+bb8d50a
example-compute2.example.com   Ready    worker   13m   v1.21.6+bb8d50a
example-compute4.example.com   Ready    worker   14m   v1.21.6+bb8d50a
example-control1.example.com   Ready    master   52m   v1.21.6+bb8d50a
example-control2.example.com   Ready    master   55m   v1.21.6+bb8d50a
example-control3.example.com   Ready    master   55m   v1.21.6+bb8d50a
```

2. 显示所有 worker 机器节点的阶段：

```
$ oc get machines -A
```

输出示例

```
NAMESPACE               NAME                              PHASE     TYPE   REGION   ZONE   AGE
openshift-machine-api   example-zbbt6-master-0            Running                          95m
openshift-machine-api   example-zbbt6-master-1            Running                          95m
openshift-machine-api   example-zbbt6-master-2            Running                          95m
openshift-machine-api   example-zbbt6-worker-0-25bhp      Running                          49m
openshift-machine-api   example-zbbt6-worker-0-8b4c2      Running                          49m
openshift-machine-api   example-zbbt6-worker-0-jkbqt      Running                          49m
openshift-machine-api   example-zbbt6-worker-0-qrl5b      Running                          49m
```

其他资源

- 获取 BareMetalHost 资源
- 监控安装进度
- 验证安装
- 基于代理的安装程序
- OpenShift Container Platform 支持的安装程序

安装范围
OpenShift Container Platform 安装程序的作用范围特意设计得比较狭窄。它旨在简化操作并确保成功。安装完成后，您可以完成更多的配置任务。

其他资源

- 如需有关 OpenShift Container Platform 配置资源的详细信息，请参阅可用的集群自定义。

1.1.6. OpenShift Local 概述

OpenShift Local 支持快速应用程序开发，以开始构建 OpenShift Container Platform 集群。OpenShift Local 设计为在本地计算机上运行，以简化设置和测试，并使用开发基于容器的应用所需的所有工具在本地模拟云环境。

无论您使用什么编程语言，OpenShift Local 都可以托管您的应用程序，并将最小预配置的 Red Hat OpenShift Container Platform 集群引入本地 PC，而无需基于服务器的基础架构。

在托管环境中，OpenShift Local 可以创建微服务，将它们转换为镜像，并在运行 Linux、macOS 或 Windows 10 或更高版本的笔记本电脑或桌面上直接运行它们。

如需有关 OpenShift Local 的更多信息，请参阅 Red Hat OpenShift Local Overview 。

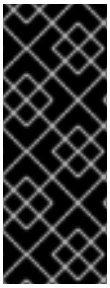
1.2. OPENSHIFT CONTAINER PLATFORM 集群支持的平台

在 OpenShift Container Platform 4.17 中，您可以在以下平台上安装使用安装程序置备的基础架构集群：

- Amazon Web Services (AWS)

- 裸机
- Google Cloud Platform (GCP)
- IBM Cloud®
- Microsoft Azure
- Microsoft Azure Stack Hub
- Nutanix
- Red Hat OpenStack Platform (RHOSP)
    - 最新的 OpenShift Container Platform 版本支持最新的 RHOSP 长生命版本和中间版本。如需完整的 RHOSP 发行版本兼容性信息，请参阅 RHOSP 上的 OpenShift Container Platform 支持列表。
- VMware vSphere

对于所有这些集群，包括用来运行安装过程的计算机在内的所有机器都必须可直接访问互联网，以便为平台容器拉取镜像并向红帽提供 telemetry 数据。



重要

安装后，不支持以下更改：

- 混合云供应商平台.
- 混合云供应商组件。例如，在安装集群的平台上使用另一个平台的持久性存储框架。

在 OpenShift Container Platform 4.17 中，您可以在以下平台上安装使用用户置备的基础架构集群：

- AWS
- Azure
- Azure Stack Hub
- 裸机
- GCP
- IBM Power®
- IBM Z® 或 IBM® LinuxONE
- RHOSP
    - 最新的 OpenShift Container Platform 版本支持最新的 RHOSP 长生命版本和中间版本。如需完整的 RHOSP 发行版本兼容性信息，请参阅 RHOSP 上的 OpenShift Container Platform 支持列表。
- AWS 上的 VMware Cloud
- VMware vSphere

根据平台支持的情况，您可以在用户置备的基础架构上执行安装，以便您可以运行具有完整互联网访问的机器，将集群放在一个代理的后面，或者执行断开连接的安装。

在断开连接的网络安装中，您可以下载安装集群所需的镜像（image），将它们放在镜像 registry（mirror registry）中，然后使用那些数据安装集群。虽然您需要访问互联网来为平台容器拉取镜像，但在 vSphere 或裸机基础架构上进行断开连接的网络安装，您的集群机器不需要直接访问互联网。

OpenShift Container Platform 4.x Tested Integrations 页面中提供了有关针对不同平台进行集成测试的详细信息。

其他资源

- 如需了解每个支持的平台可用的安装类型的更多信息，请参阅不同平台支持的安装方法。
- 有关选择安装方法以及准备所需资源的信息，请参阅选择集群安装方法并为用户准备。
- Red Hat OpenShift Network Calculator 可帮助您在部署和扩展阶段设计集群网络。它解决了与集群网络相关的常见问题，并以方便的 JSON 格式提供输出。

第 2 章 选择集群安装方法并为用户准备它

在安装 OpenShift Container Platform 前，请确定您拥有为用户准备集群所需的所有所需资源。

2.1. 选择集群安装类型

在安装 OpenShift Container Platform 集群前，需要选择最佳安装说明。请考虑您对以下问题的回答，以选择最佳选择。

2.1.1. 您要自己安装和管理 OpenShift Container Platform 集群吗？

如果要自己安装和管理 OpenShift Container Platform，您可以在以下平台上安装它：

- 64 位 x86 实例上的 Amazon Web Services (AWS)
- 64 位 ARM 实例上的 Amazon Web Services (AWS)
- 64 位 x86 实例上的 Microsoft Azure
- 64 位 ARM 实例上的 Microsoft Azure
- Microsoft Azure Stack Hub
- 64 位 x86 实例上的 Google Cloud Platform (GCP)
- 64 位 ARM 实例上的 Google Cloud Platform (GCP)
- Red Hat OpenStack Platform (RHOSP)
- IBM Cloud®
- IBM Z® 或 IBM® LinuxONE
- IBM Z® or IBM® LinuxONE for Red Hat Enterprise Linux (RHEL) KVM
- IBM Power®
- IBM Power® Virtual Server
- Nutanix
- VMware vSphere
- 裸机或其他平台基础架构

您可以将 OpenShift Container Platform 4 集群部署到内部硬件环境，或部署到云托管服务中，但集群中的所有机器都必须位于相同的数据中心或云托管服务中。

如果要使用 OpenShift Container Platform，但您不想自行管理集群，您可以从几个受管服务选项中选择。如果要完全由红帽管理的集群，您可以使用 OpenShift Dedicated。您还可以在 Azure、AWS、IBM Cloud® 或 Google Cloud Platform 上使用 OpenShift 作为受管服务。有关受管服务的更多信息，请参阅 OpenShift 产品页。如果您安装了使用云虚拟机作为虚拟裸机的 OpenShift Container Platform 集群，则其对应的基于云的存储不被支持。

2.1.2. 您是否已使用了 OpenShift Container Platform 3 且要使用 OpenShift Container Platform 4?

如果您已使用了 OpenShift Container Platform 3 并希望尝试 OpenShift Container Platform 4，则需要了解 OpenShift Container Platform 4 的不同。OpenShift Container Platform 4 将无缝地集成了软件包、部署和管理 Kubernetes 应用程序以及平台在 Red Hat Enterprise Linux CoreOS（RHCOS）上运行的 Operator。与其他需要部署机器并配置其操作系统以便在其中安装 OpenShift Container Platform 的系统不同，RHCOS 操作系统是 OpenShift Container Platform 集群的一个内部组成部分。为集群机器部署操作系统是 OpenShift Container Platform 的安装过程的一部分。请参阅 OpenShift Container Platform 3 和 4 之间的差别。

由于需要置备机器作为 OpenShift Container Platform 集群安装过程的一部分，所以无法将 OpenShift Container Platform 3 集群升级到 OpenShift Container Platform 4。相反，您必须创建新的 OpenShift Container Platform 4 集群，并将 OpenShift Container Platform 3 工作负载迁移到它们。有关迁移的更多信息，请参阅从 OpenShift Container Platform 3 迁移到 4 概述 。由于必须迁移到 OpenShift Container Platform 4，因此可以使用任何类型的生产环境集群安装过程来创建新集群。

2.1.3. 您是否希望在您的集群中使用已存在的组件？

由于操作系统是 OpenShift Container Platform 集成的一部分，因此让安装程序可以更轻松地支持所有基础架构。它们被称为*安装程序置备的基础架构*安装。在这种安装中，您可以为集群提供一些现有的基础架构，但安装程序会部署集群初始需要的所有机器。

您可以在不对集群或其底层机器自定义 AWS, Azure, Azure Stack Hub, GCP, 或 Nutanix 的情况下部署安装程序置备的基础架构集群。

如果需要为安装程序置备的基础架构集群执行基本配置，如集群机器的实例类型，您可以自定义 AWS、Azure 、GCP、Nutanix 的安装。

对于安装程序置备的基础架构安装，您可以使用现存的 VPC in AWS, vNet in Azure, 或 VPC in GCP。您还可以重复使用网络基础架构的一部分，以便 AWS、Azure、GCP 中的集群可以与环境中的现有 IP 地址分配共存，并与现有的 MTU 和 VXLAN 配置集成。如果在这些云上已有帐户和凭证，您可以重复使用这些帐户，但可能需要修改帐户，以便具有在它们上安装 OpenShift Container Platform 集群所需的权限。

您可以使用安装程序置备的基础架构方法，在硬件上为 vSphere 和裸机 创建适当的机器实例。另外，对于 vSphere，您还可以在安装过程中自定义额外网络参数。

对于一些安装程序置备的基础架构安装，例如在 VMware vSphere 和裸机平台上，达到入口虚拟 IP (VIP) 的外部流量在默认 **IngressController** 副本之间没有平衡。对于超过基准 **IngressController** 路由器性能的 vSphere 和裸机安装程序置备的基础架构安装，您必须配置外部负载均衡器。配置外部负载均衡器可达到多个 **IngressController** 副本的性能。有关基准 **IngressController** 性能的更多信息，请参阅 Baseline Ingress Controller (router)性能。有关配置外部负载均衡器的更多信息，请参阅配置用户管理的负载均衡器。

如果要重复使用广泛的云基础架构，可以完成*用户置备的基础架构*安装。使用这些安装，您可以在安装过程中手动部署集群所需的机器。如果在 AWS、Azure、Azure Stack Hub 上执行用户置备的基础架构安装，您可以使用提供的模板来帮助备份所有需要的组件。您还可以重复使用一个共享的 VPC on GCP。或者，您可以使用 供应商安装方法 将集群部署到其他云中。

您还可以在现有硬件上完成用户置备的基础架构安装。如果您使用 RHOSP、IBM Z® 或 IBM® LinuxONE、IBM Z® 和 IBM® LinuxONE with RHEL KVM 、IBM Power 或 vSphere，请使用特定的安装说明来部署集群。如果您使用其他支持的硬件，请按照裸机安装过程进行操作。对于其中一些平台，如 vSphere 和裸机，您也可以在安装过程中自定义额外网络参数。

2.1.4. 您的集群是否需要额外的安全性？

如果使用用户置备的安装方法，您可以为集群配置代理。这些说明包含在每个安装过程中。

如果要防止公共云中的集群从外部公开端点，您可以在 AWS、Azure 或 GCP 上使用安装程序置备的基础架构部署私有集群。

如果您需要安装对互联网有限访问的集群，如断开连接的或受限的网络集群，您可以镜像安装软件包并从中安装集群。按照用户置备的基础架构安装到 AWS、GCP、IBM Z® 或 IBM® LinuxONE、IBM Z® 或 IBM® LinuxONE with RHEL KVM 的IBM Z® 或 IBM® LinuxONE with RHEL KVM 、IBM Power®、vSphere 或裸机 的受限网络中。您还可以按照 AWS、GCP、IBM Cloud®、Nutx、RHOSP 和 vSphere 的详细信息，使用安装程序置备的基础架构将集群安装到受限网络中。https://docs.redhat.com/en/documentation/openshift_container_platform/4.17/html-single/installing_on_nutanix/#installing-restricted-networks-nutanix-installer-provisioned

如果需要将集群部署到 AWS GovCloud 区域、AWS 中国区域或 Azure 政府区域，您可以在安装程序置备的基础架构安装过程中配置这些自定义区域。

您还可以将集群机器配置为使用 RHEL 加密库在安装过程中为 FIPS 140-2/140-3 Validation 提交给 NIST。

重要

当以 FIPS 模式运行 Red Hat Enterprise Linux (RHEL) 或 Red Hat Enterprise Linux CoreOS (RHCOS)时，OpenShift Container Platform 核心组件使用 RHEL 加密库，在 x86_64、ppc64le 和 s390x 架构上提交到 NIST FIPS 140-2/140-3 Validation。

2.2. 安装后为用户准备集群

在安装集群时不需要进行一些配置，但建议在用户访问集群前进行操作。您可以通过自定义组成集群的 Operator，并将集群与其他所需系统（如身份提供程序）集成，从而自定义集群本身。

对于生产环境集群，您必须配置以下集成：

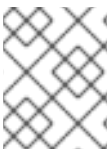
- 持久性存储
- 身份供应商
- 监控 OpenShift Container Platform 核心组件

2.3. 为工作负载准备集群

根据工作负载需要，您可能需要在开始部署应用程序前执行额外的步骤。例如，在为应用程序构建策略准备了基础架构后，您可能需要为低延迟工作负载置备或保护敏感工作负载。您还可以为应用程序工作负载配置 监控。如果您计划运行 Windows 工作负载，则必须在安装过程中启用 带有 OVN-Kubernetes 的混合网络; 安装集群后无法启用混合网络。

2.4. 支持的用于不同平台的安装方法

您可以在不同的平台上执行不同类型的安装。



注意

不是所有安装选项都支持所有平台，如下表所示。勾选标记代表支持的选项，并链接到相关部分。

表 2.1. 安装程序置备的基础架构选项

	AWS (64位 x86)	AWS (64位 ARM)	Azure (64位 x86)	Azure (64位 ARM)	Azure Stack Hub	GCP (64位 x86)	GCP (64位 ARM)	Nutanix	RHOSP	裸机 (64位 x86)	裸机 (64位 ARM)	vSphere	IBM Cloud®	IBM Z®	IBM Power®	IBM Power® Virtual Server
Default（默认）	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓			
Custom	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓			✓
网络自定义	✓	✓	✓	✓	✓	✓	✓					✓	✓			
Restricted network	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓			✓
私有集群	✓	✓	✓	✓		✓	✓						✓			✓

	AWS (64 位 x86)	AWS (64 位 ARM)	Azure (64 位 x86)	Azure (64 位 ARM)	Azure Stack Hub	GCP (64 位 x86)	GCP (64 位 ARM)	Nutanix	RHOSP	裸机 (64 位 x86)	裸机 (64 位 ARM)	vSphere	IBM Cloud®	IBM Z®	IBM Power®	IBM Power® Virtual Server
现有的虚拟私有网络	✓	✓	✓	✓		✓	✓						✓			✓
政府区域	✓		✓													
Secret 区域	✓															
中国区域	✓															

表 2.2. 用户置备的基础架构

	AWS (64位x86)	AWS (64位ARM)	Azure (64位x86)	Azure (64位ARM)	Azure Stack Hub	GCP (64位x86)	GCP (64位ARM)	Nutanix	RHOSP	裸机 (64位x86)	裸机 (64位ARM)	vSphere	IBM Cloud®	IBM Z®	使用RHEL KVM的IBM Z®	IBM Power®	平台无关
Custom	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		✓	✓	✓	✓
网络自定义										✓	✓	✓					
Restricted network	✓	✓				✓	✓			✓	✓	✓		✓	✓	✓	
在集群项目外托管共享VPC						✓	✓										

第 3 章 集群功能

集群管理员可在安装前使用集群功能启用或禁用可选组件。集群管理员可以在安装后随时启用集群功能。

注意

集群管理员无法在启用集群后禁用集群功能。

3.1. 启用集群功能

如果您使用包含自定义集群的安装方法，通过创建 **install-config.yaml** 文件，您可以选择您要在集群中可用的集群功能。

注意

如果通过启用或禁用特定集群功能自定义集群，则需要手动维护 **install-config.yaml** 文件。新的 OpenShift Container Platform 更新可能会为现有组件声明新的功能处理，或者完全引入新的组件。自定义 **install-config.yaml** 文件的用户应该会在 OpenShift Container Platform 更新时定期更新其 **install-config.yaml** 文件。

您可以使用以下配置参数来选择集群功能：

```
capabilities:
  baselineCapabilitySet: v4.11 (1)
  additionalEnabledCapabilities: (2)
  - CSISnapshot
  - Console
  - Storage
```

1. 定义要安装的一组基准功能。有效值为 **None**、**vCurrent** 和 **v4.x**。如果选择 **None**，则禁用所有可选功能。默认值为 **vCurrent**，它启用了所有可选功能。

注意

v4.x 代表最高为当前集群版本（包括当前版本）的任何值。例如，OpenShift Container Platform 4.12 集群的有效值为 **v4.11** 和 **v4.12**。

2. 定义要显式启用的功能列表。除了 **baselineCapabilitySet** 中指定的功能外，这些功能也会启用。

注意

在本例中，默认能力被设置为 **v4.11**。**additionalEnabledCapabilities** 字段启用了默认的 **v4.11** 功能集的额外功能。

下表描述了 **baselineCapabilitySet** 值。

表 3.1. 集群功能 baselineCapabilitySet 值描述

值	描述
vCurrent	当您要自动添加新版本中引入的新功能时，指定这个选项。
v4.11	当要为 OpenShift Container Platform 4.11 启用默认功能时指定这个选项。通过指定 **v4.11**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.11 中的默认功能是 **baremetal**、**MachineAPI**、**marketplace** 和 **openshift-samples**。
v4.12	当您要为 OpenShift Container Platform 4.12 启用默认功能时指定这个选项。通过指定 **v4.12**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.12 中的默认功能是 **baremetal**,**MachineAPI**,**marketplace**,**openshift-samples**,**Console**,**Insights**,**Storage**, 和 **CSISnapshot**。
v4.13	当要为 OpenShift Container Platform 4.13 启用默认功能时，指定这个选项。通过指定 **v4.13**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.13 中的默认功能是 **baremetal**,**MachineAPI**,**marketplace**,**openshift-samples**,**Console**,**Insights**,**Storage**,**CSISnapshot**, 和 **NodeTuning**。
v4.14	当要为 OpenShift Container Platform 4.14 启用默认功能时指定这个选项。通过指定 **v4.14**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.14 中的默认功能是 **baremetal**,**MachineAPI**,**marketplace**,**openshift-samples**,**Console**,**Insights**,**Storage**,**CSISnapshot**,**NodeTuning**,**ImageRegistry**,**Build**, 和 **DeploymentConfig**。
v4.15	当要为 OpenShift Container Platform 4.15 启用默认功能时指定这个选项。通过指定 **v4.15**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.15 中的默认功能是 **baremetal**,**MachineAPI**,**marketplace**,**OperatorLifecycleManager**,**openshift-samples**,**Console**,**Insights**,**Storage**,**CSISnapshot**,**NodeTuning**,**ImageRegistry**,**Build**,**CloudCredential**, 和 **DeploymentConfig**。

值	描述
v4.16	当要为 OpenShift Container Platform 4.16 启用默认功能时指定这个选项。通过指定 **v4.16**，不会启用较新版本的 OpenShift Container Platform 中引入的功能。OpenShift Container Platform 4.16 中的默认功能是 **baremetal**, **MachineAPI**, **marketplace**, **OperatorLifecycleManager**, **openshift-samples**, **Console**, **Insights**, **Storage**, **CSISnapshot**, **NodeTuning**, **ImageRegistry**, **Build**, **CloudCredential**, **DeploymentConfig**, 和 **CloudControllerManager**。
None	指定其他集合太大，您不需要任何功能，或者想要 **通过额外的 EnabledCapabilities** 进行微调。

其他资源

- 使用自定义在 AWS 上安装集群
- 使用自定义在 GCP 上安装集群

3.2. OPENSHIFT CONTAINER PLATFORM 4.17 中的可选集群功能

目前，集群 Operator 为这些可选功能提供功能。以下总结了每个功能提供的功能，并在禁用时丢失的功能。

其他资源

- 集群 Operator 参考

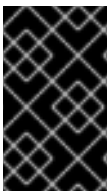
3.2.1. 裸机功能

用途
Cluster Baremetal Operator 为 **baremetal** 功能提供功能。

Cluster Baremetal Operator (CBO) 会部署使裸机服务器成为一个可完全正常工作的节点以运行 OpenShift Container Platform 计算节点所需的所有组件。CBO 确保 metal3 部署（由 Bare Metal Operator (BMO) 和 Ironic 容器组成）在 OpenShift Container Platform 集群内的一个 control plane 节点上运行。CBO 还会侦听 OpenShift Container Platform 对资源的更新，它会监视并采取适当的操作。

使用安装程序置备的基础架构部署需要裸机功能。禁用裸机功能可能会导致这些部署出现意外问题。

建议集群管理员仅在带有集群中没有任何 **BareMetalHost** 资源的用户置备的基础架构禁用裸机功能。



重要

如果禁用裸机功能，集群将无法置备或管理裸机节点。只有在部署中没有 **BareMetalHost** 资源时才禁用该功能。**baremetal** 能力取决于 **MachineAPI** 功能。如果启用 **baremetal** 功能，还必须启用 **MachineAPI**。

其他资源

- 在裸机上部署安装程序置备的集群
- 准备裸机集群安装
- 使用 Bare Metal Operator 配置

3.2.2. 构建功能

用途
Build 功能启用 **Build** API。**Build** API 管理 **Build** 和 **BuildConfig** 对象的生命周期。

重要

如果您禁用 **Build** 功能，集群中没有提供以下资源：

- **Build** 和 **BuildConfig** 资源
- **builder** 服务帐户

仅在不需要 **Build** 或 **BuildConfig** 资源，或集群中有 **builder** 服务帐户时，才禁用 **Build** 功能。

3.2.3. 云控制器管理器功能

用途
Cloud Controller Manager Operator 为 **CloudControllerManager** 提供功能。

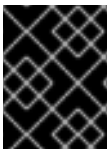
注意

目前，在所有平台上都不支持禁用 **CloudControllerManager** 功能。

您可以通过检查集群的安装配置 (**install-config.yaml**) 文件中的值来确定集群是否支持禁用 **CloudControllerManager** 功能。

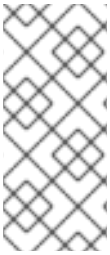
在 **install-config.yaml** 文件中，找到 **platform** 参数。

- 如果 **platform** 参数的值是 **Baremetal** 或 **None**，您可以在集群中禁用 **CloudControllerManager** 功能。
- 如果 **platform** 参数的值是 **External**，找到 **platform.external.cloudControllerManager** 参数。如果 **platform.external.cloudControllerManager** 参数的值为 **None**，您可以在集群中禁用 **CloudControllerManager** 功能。



重要

如果这些参数包含除列出的值以外的其他值，则无法在集群中禁用 **CloudControllerManager** 功能。



注意

对于 Amazon Web Services (AWS), Google Cloud Platform (GCP), IBM Cloud®, global Microsoft Azure, Microsoft Azure Stack Hub, Nutanix, Red Hat OpenStack Platform (RHOSP), 和 VMware vSphere，这个 Operator 的状态是正式发布（GA）。

对于 IBM Power® Virtual Server，Operator 作为技术预览提供

Cloud Controller Manager Operator 管理并更新在 OpenShift Container Platform 上部署的云控制器管理器。Operator 基于 Kubebuilder 框架和 **controller-runtime** 库。它通过 Cluster Version Operator (CVO) 安装。

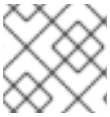
它包含以下组件：

- Operator
- 云配置观察

默认情况下，Operator 通过 **metrics** 服务公开 Prometheus 指标数据。

3.2.4. 云凭证功能

用途
Cloud Credential Operator 为 **CloudCredential** 提供功能。



注意

目前，禁用 **CloudCredential** 只在裸机集群中被支持。

Cloud Credential Operator(CCO)将云供应商凭证作为 Kubernetes 自定义资源定义(CRD)进行管理。**CredentialsRequest** 自定义资源（CR）的 CCO 同步，允许 OpenShift Container Platform 组件使用集群运行所需的特定权限请求云供应商凭证。

通过在 **install-config.yaml** 文件中为 **credentialsMode** 参数设置不同的值，可将 CCO 配置为以几种不同模式操作。如果没有指定模式，或将 **credentialsMode** 参数被设置为空字符串（""）。

其他资源

- 关于 Cloud Credential Operator

3.2.5. 集群 Image Registry 功能

用途
Cluster Image Registry Operator 为 **ImageRegistry** 功能提供功能。

Cluster Image Registry Operator 管理 OpenShift 镜像 registry 的单个实例。它管理 registry 的所有配置，包括创建存储。

在初始启动时，Operator 会基于集群中检测到的配置创建默认的 **image-registry** 资源实例。这代表了根据云供应商要使用的云存储类型。

如果没有足够的信息来定义完整的 **image-registry** 资源，则会定义一个不完整的资源，Operator 将更新资源状态以提供缺失的内容。

Cluster 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。

要将镜像 registry 集成到集群的用户身份验证和授权系统中，会为集群中的每个服务帐户生成一个镜像 pull secret。

重要

如果您禁用 **ImageRegistry** 功能，或者在 Cluster Image Registry Operator 配置中禁用集成的 OpenShift 镜像 registry，则不会为每个服务帐户生成镜像 pull secret。

如果禁用 **ImageRegistry** 功能，您可以在 Telco 环境中减少 OpenShift Container Platform 的整体资源占用空间。根据您的部署，如果需要，可以禁用此组件。

项目
cluster-image-registry-operator

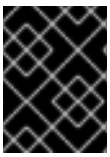
其他资源

- OpenShift Container Platform中的Image Registry Operator
- 自动生成的 secret

3.2.6. 集群存储功能

用途
Cluster Storage Operator 为**存储**功能提供功能。

Cluster Storage Operator 设置 OpenShift Container Platform 集群范围内的存储默认设置。它确保了 OpenShift Container Platform 集群存在默认**存储类**。它还安装 Container Storage Interface (CSI) 驱动程序，使集群能够使用各种存储后端。



重要

如果禁用了集群存储功能，集群将没有默认的 **storageclass** 或任何 CSI 驱动程序。具有管理员特权的用户可以创建默认**存储类**，并在禁用集群存储功能时手动安装 CSI 驱动程序。

备注

- Operator 创建的存储类可以通过编辑其注解来实现非默认设置，但只要 Operator 运行，这个存储类就无法被删除。

3.2.7. 控制台功能

用途
Console Operator 为 **Console** 功能提供功能。

Console Operator 在集群中安装和维护 OpenShift Container Platform web 控制台。Console Operator 会被默认安装，并自动维护控制台。

其他资源

- Web 控制台概述

3.2.8. CSI 快照控制器功能

用途
Cluster CSI Snapshot Controller Operator 为 **CSISnapshot** 功能提供功能。

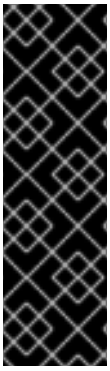
Cluster CSI Snapshot Controller Operator 安装和维护 CSI Snapshot Controller。CSI Snapshot Controller 负责监视 **VolumeSnapshot** CRD 对象，并管理卷快照的创建和删除生命周期。

其他资源

- CSI 卷快照

3.2.9. DeploymentConfig 功能

用途
DeploymentConfig 功能启用和管理 **DeploymentConfig** API。



重要

如果禁用 **DeploymentConfig** 功能，集群中将无法使用以下资源：

- **DeploymentConfig** 资源
- **deployer** 服务帐户

仅在不需要 **DeploymentConfig** 资源以及集群中有 **deployer** 服务帐户时才禁用 **DeploymentConfig** 功能。

3.2.10. Insights 功能

用途
Insights Operator 为 **Insights** 功能提供功能。

Insights Operator 收集 OpenShift Container Platform 配置数据并将其发送到红帽。数据用于生成有关集群可能暴露的潜在问题的主动分析建议。这些建议通过 console.redhat.com 上的 Insights Advisor 与集群管理员通信。

备注
Insights Operator 补充 OpenShift Container Platform Telemetry。

其他资源

- 使用 Insights Operator

3.2.11. 机器 API 功能

用途
machine-api-operator、**cluster-autoscaler-operator** 和 **cluster-control-plane-machine-set-operator** Operator 提供了与 **MachineAPI** 功能相关的功能。只有在使用用户置备的基础架构安装集群时，才能禁用此功能。

Machine API 功能负责集群中的所有机器配置和管理。如果在安装过程中禁用 Machine API 功能，则需要手动管理所有与机器相关的任务。

其他资源

- 机器管理概述
- Machine API Operator
- Cluster Autoscaler Operator
- Control Plane Machine Set Operator

3.2.12. Marketplace 功能

用途
Marketplace Operator 提供了 **marketplace** 功能。

Marketplace Operator 通过使用集群中的一组默认 Operator Lifecycle Manager (OLM) 目录简化了将非集群 Operator 引入集群的过程。安装 Marketplace Operator 时，它会创建 **openshift-marketplace** 命名空间。OLM 确保在 **openshift-marketplace** 命名空间中安装的目录源可用于集群中的所有命名空间。

如果禁用 **marketplace** 功能，Marketplace Operator 不会创建 **openshift-marketplace** 命名空间。目录源仍可在集群中配置和管理，但 OLM 依赖于 **openshift-marketplace** 命名空间，以便目录可供集群中的所有命名空间使用。有权创建带 **openshift-** 前缀的命名空间（如系统或集群管理员）的用户可以手动创建 **openshift-marketplace** 命名空间。

如果启用 **marketplace** 功能，您可以通过配置 Marketplace Operator 来启用和禁用单个目录。

其他资源

- 红帽提供的 Operator 目录

3.2.13. 节点调优功能

用途
Node Tuning Operator 为 **NodeTuning** 功能提供功能。

Node Tuning Operator 可以帮助您通过编排 TuneD 守护进程来管理节点级别的性能优化，并使用 Performance Profile 控制器获得低延迟性能。大多数高性能应用程序都需要一定程度的内核级性能优化。Node Tuning Operator 为用户提供了一个统一的、节点一级的 sysctl 管理接口，并可以根据具体用户的需要灵活地添加自定义性能优化设置。

如果您禁用了 NodeTuning 功能，一些默认的性能优化设置不会应用到 control-plane 节点。这可能会限制具有 900 个节点或 900 路由的大型集群的可扩展性和性能。

其他资源

- 使用 Node Tuning Operator

3.2.14. OpenShift 示例功能

用途
Cluster Samples Operator 为 **openshift-samples** 功能提供功能。

Cluster Samples Operator 管理存储在 **openshift** 命名空间中的示例镜像流和模板。

在初始启动时，Operator 会创建默认样本配置资源来启动镜像流和模板的创建。配置对象是一个集群范围内的对象，它带有一个键 **cluster** 和类型 **configs.samples**。

镜像流是基于 Red Hat Enterprise Linux CoreOS (RHCOS) 的 OpenShift Container Platform 镜像流，指向 **registry.redhat.io** 上的镜像。同样，模板也被归类为 OpenShift Container Platform 模板。

如果您禁用示例功能，用户无法访问它提供的镜像流、示例和模板。根据您的部署，如果不需要，您可能需要禁用此组件。

其他资源

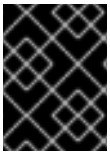
- 配置 Cluster Samples Operator

3.2.15. Operator Lifecycle Manager 功能

用途
Operator Lifecycle Manager （OLM）可帮助用户安装、更新和管理所有 Kubernetes 原生应用程序（Operator）以及在 OpenShift Container Platform 集群中运行的关联服务的生命周期。它是 Operator Framework 的一部分，后者是一个开源工具包，用于以有效、自动化且可扩展的方式管理 Operator。

如果 Operator 需要以下 API，则必须启用 **OperatorLifecycleManager** 功能：

- **ClusterServiceVersion**
- **CatalogSource**
- **订阅**
- **InstallPlan**
- **OperatorGroup**



重要

marketplace 功能取决于 **OperatorLifecycleManager** 功能。您无法禁用 **OperatorLifecycleManager** 功能并启用 **marketplace** 功能。

其他资源

- Operator Lifecycle Manager 概念和资源

3.3. 查看集群功能

作为集群管理员，您可以使用 **clusterversion** 资源状态来查看功能。

先决条件

- 已安装 OpenShift CLI（**oc**）。

流程

- 要查看集群功能的状态，请运行以下命令：

```
$ oc get clusterversion version -o jsonpath='{.spec.capabilities}{"\n"}{.status.capabilities}{"\n"}'
```

输出示例

```
{"additionalEnabledCapabilities":["openshift-samples"],"baselineCapabilitySet":"None"}
{"enabledCapabilities":["openshift-samples"],"knownCapabilities":
["CSISnapshot","Console","Insights","Storage","baremetal","marketplace","openshift-
samples"]}
```

3.4. 通过设置基准功能集启用集群功能

作为集群管理员，您可以通过设置 **baselineCapabilitySet** 配置参数，在 OpenShift Container Platform 安装后随时启用集群功能。

先决条件

- 已安装 OpenShift CLI（**oc**）。

流程

- 要设置 **baselineCapabilitySet** 配置参数，请运行以下命令：

```
$ oc patch clusterversion version --type merge -p '{"spec":{"capabilities":
{"baselineCapabilitySet":"vCurrent"}}}' 1
```

1 对于 **baselineCapabilitySet**，您可以指定 **vCurrent**、**v4.17** 或 **None**。

3.5. 通过设置其他启用的功能来启用集群功能

作为集群管理员，您可以通过设置 **additionalEnabledCapabilities** 配置参数，在 OpenShift Container Platform 安装后随时启用集群功能。

先决条件

- 已安装 OpenShift CLI（**oc**）。

流程

1. 运行以下命令查看附加启用的功能：

```
$ oc get clusterversion version -o jsonpath='{.spec.capabilities.additionalEnabledCapabilities}
{"\n"}'
```

输出示例

```
["openshift-samples"]
```

2. 要设置 **additionalEnabledCapabilities** 配置参数，请运行以下命令：

```
$ oc patch clusterversion/version --type merge -p '{"spec":{"capabilities":
{"additionalEnabledCapabilities":["openshift-samples", "marketplace"]}}}'
```

重要

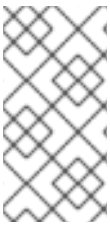
无法禁用集群中已经启用的功能。集群版本 Operator (CVO) 继续协调集群中已经启用的功能。

如果您尝试禁用某个功能，则 CVO 会显示相关的 spec:

```
$ oc get clusterversion version -o jsonpath='{.status.conditions[?
(@.type=="ImplicitlyEnabledCapabilities")]}{"\n"}'
```

输出示例

```
{"lastTransitionTime":"2022-07-22T03:14:35Z","message":"The following capabilities could not be
disabled: openshift-
samples","reason":"CapabilitiesImplicitlyEnabled","status":"True","type":"ImplicitlyEnabledCapabilities"}
```



注意

在集群升级过程中，可以隐式启用给定功能。如果在升级前已在集群上运行资源，那么将启用属于资源的任何功能。例如，在集群升级过程中，已在集群中运行的资源已更改为系统已作为 **marketplace** 功能的一部分。即使集群管理员没有明确启用了 **marketplace** 功能，它也会被系统隐式启用。

第 4 章 支持 FIPS 加密

您可以使用 FIPS 模式安装 OpenShift Container Platform 集群。

OpenShift Container Platform 专为 FIPS 设计。当以 FIPS 模式运行 Red Hat Enterprise Linux (RHEL) 或 Red Hat Enterprise Linux CoreOS (RHCOS)时，OpenShift Container Platform 核心组件使用 RHEL 加密库，在 x86_64、ppc64le 和 s390x 架构上提交到 NIST FIPS 140-2/140-3 Validation。

有关 NIST 验证程序的更多信息，请参阅加密模块验证程序。有关为验证提交的 RHEL 加密库的单独版本的最新 NIST 状态，请参阅 Compliance Activities 和 Government Standards。

重要

要为集群启用 FIPS 模式，您必须从一个配置为以 FIPS 模式运行的 RHEL 9 计算机中运行安装程序，且必须使用安装程序支持的 FIPS 版本。请参阅 *使用 'oc adm extract' 的 FIPS 的安装程序*部分。

有关在 RHEL 中配置 FIPS 模式的更多信息，请参阅在 FIPS 模式中安装该系统。

对于集群中的 Red Hat Enterprise Linux CoreOS(RHCOS)机器，当机器根据 **install-config.yaml** 文件中的选项的状态进行部署时，会应用这个更改，该文件管理用户在集群部署过程中可以更改的集群选项。在 Red Hat Enterprise Linux(RHEL)机器中，您必须在计划用作 worker 机器的机器上安装操作系统时启用 FIPS 模式。

因为 FIPS 必须在集群首次引导的操作系统前启用，所以您不能在部署集群后启用 FIPS。

4.1. 使用 OC ADM EXTRACT 获取支持 FIPS 的安装程序

OpenShift Container Platform 需要使用支持 FIPS 的安装二进制文件来在 FIPS 模式中安装集群。您可以使用 OpenShift CLI (**oc**) 从发行镜像中提取该二进制文件。获取二进制文件后，您可以继续集群安装，将 **openshift-install** 命令的所有实例替换为 **openshift-install-fips**。

先决条件

- 已使用版本 4.16 或更新版本安装了 OpenShift CLI (**oc**)。

流程

1. 运行以下命令，从安装程序中提取支持 FIPS 的二进制文件：

```
$ oc adm release extract --registry-config "${pullsecret_file}" --command=openshift-install-fips --to "${extract_dir}" ${RELEASE_IMAGE}
```

其中：

<pullsecret_file>

指定包含 pull secret 的文件的名称。

<extract_dir>

指定您要提取二进制文件的目录。

<RELEASE_IMAGE>

指定您使用的 OpenShift Container Platform 发行版本的 Quay.io URL。有关查找发行镜像的更多信息，请参阅*提取 OpenShift Container Platform 安装程序*。

2. 继续集群安装，将 **openshift-install** 命令的所有实例替换为 **openshift-install-fips**。

其他资源

- 提取 OpenShift Container Platform 安装程序

4.2. 使用公共 OPENSHIFT 镜像获取支持 FIPS 的安装程序

OpenShift Container Platform 需要使用支持 FIPS 的安装二进制文件来在 FIPS 模式中安装集群。您可以通过从公共 OpenShift 镜像下载来获取此二进制文件。获取二进制文件后，进行集群安装，将 **openshift-install** 二进制文件的所有实例替换为 **openshift-install-fips**。

先决条件

- 您可以访问互联网。

流程

1. 从 https://mirror.openshift.com/pub/openshift-v4/clients/ocp/latest-4.17/openshift-install-rhel9-amd64.tar.gz 下载安装程序。
2. 提取安装程序。例如，在使用 Linux 操作系统的计算机上运行以下命令：

```
$ tar -xvf openshift-install-rhel9-amd64.tar.gz
```

3. 继续集群安装，将 **openshift-install** 命令的所有实例替换为 **openshift-install-fips**。

4.3. OPENSHIFT CONTAINER PLATFORM 中的 FIPS 验证

OpenShift Container Platform 在 RHEL 和 RHCOS 中使用特定的 FIPS 验证或 Modules In Process 模块用于使用它们的操作系统组件。请参阅 RHEL 核心加密组件。例如，当用户使用 SSH 连接到 OpenShift Container Platform 集群和容器时，这些连接会被正确加密。

OpenShift Container Platform 组件以 Go 语言编写，并使用红帽的 golang 编译器构建。当您为集群启用 FIPS 模式时，需要加密签名的所有 OpenShift Container Platform 组件都会调用 RHEL 和 RHCOS 加密库。

表 4.1. OpenShift Container Platform 4.17 中的 FIPS 模式属性和限制

属性	限制：
RHEL 9 和 RHCOS 操作系统支持 FIPS。	FIPS 实现没有使用在单一步骤中执行哈希计算和签名生成或验证的功能。在以后的 OpenShift Container Platform 版本中，将继续评估并改进此限制。
CRI-O 运行时支持 FIPS。	
OpenShift Container Platform 服务支持 FIPS。	
从 RHEL 9 和 RHCOS 二进制文件和镜像中获得的 FIPS 验证或模块加密模块和算法。	

属性	限制：
使用 FIPS 兼容 golang 编译器。	TLS FIPS 并不完善，但计划在将来的 OpenShift Container Platform 版本中被支持。
支持多个架构中的 FIPS。	目前，只有使用 **x86_64**、**ppc64le** 和 **s390x** 架构的 OpenShift Container Platform 部署中才支持 FIPS。

4.4. 集群使用的组件支持 FIPS

虽然 OpenShift Container Platform 集群本身使用 FIPS 验证或Modules In Process 模块，但请确保支持 OpenShift Container Platform 集群的系统使用 FIPS 验证的或模块 In Process 模块进行加密。

4.4.1. etcd

要确保存储在 etcd 中的 secret 使用 FIPS 验证的/Modules in Process 加密，以 FIPS 模式引导节点。在以 FIPS 模式安装集群后，您可以使用 FIPS 批准的 **aes cbc** 加密算法加密 etcd 数据。

4.4.2. Storage

对于本地存储，使用 RHEL 提供的磁盘加密或使用 RHEL 提供的磁盘加密的容器原生存储。通过将所有数据存储到使用 RHEL 提供的磁盘加密的卷中，并为您的集群启用 FIPS 模式，静态数据和正在启动的数据或网络数据都受到 FIPS 验证的/Modules in Process 加密的保护。您可以将集群配置为加密每个节点的根文件系统，如 自定义节点 中所述。

4.4.3. runtimes

要确保容器知道它们在使用 FIPS 验证的/Modules in Process 加密模块的主机上运行，请使用 CRI-O 管理您的运行时。

4.5. 在 FIPS 模式下安装集群

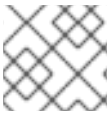
要使用 FIPS 模式安装集群，请按照在首选基础架构上安装自定义集群的说明进行。在部署集群前，请确定在 **install-config.yaml** 文件中设置了 **fips: true**。

重要

要为集群启用 FIPS 模式，您必须从配置为以 FIPS 模式操作的 RHEL 计算机运行安装程序。有关在 RHEL 中配置 FIPS 模式的更多信息，请参阅在 FIPS 模式中安装该系统。

- Amazon Web Services
- Microsoft Azure
- 裸机
- Google Cloud Platform
- IBM Cloud®
- IBM Power®

- IBM Z® 和 IBM® LinuxONE
- IBM Z® 和 IBM® LinuxONE with RHEL KVM
- Red Hat OpenStack Platform(RHOSP)
- VMware vSphere



注意

如果使用 Azure File 存储，则无法启用 FIPS 模式。

要将 **AES CBC** 加密应用到 etcd 数据存储中，请在安装集群后按照 加密 etcd 数据 过程进行操作。

如果您在集群中添加 RHEL 节点，请确保在机器初始引导前启用 FIPS 模式。请参阅将 RHEL 计算机器添加到 OpenShift Container Platform 集群，以及以 FIPS 模式安装系统。