OpenShift Container Platform 4.2

在 vSphere 上安装

安装 OpenShift Container Platform 4.2 vSphere 集群

Last Updated: 2020-08-21

OpenShift Container Platform 4.2 在 vSphere 上安装

安装 OpenShift Container Platform 4.2 vSphere 集群

法律通告

摘要

本文档提供在 VMware vSphere 上安装和卸载 OpenShift Container Platform 4.2 集群的说明。

目录

第 1 章 在 VSPHERE 上安装

1.1. 在 VSPHERE 上安装集群

在 OpenShift Container Platform 版本 4.2 中，您可以在您置备的 VMware vSphere 基础架构上安装集群。

先决条件

- 为集群置备持久性存储。若要部署私有镜像 registry，您的存储必须提供 ReadWriteMany 访问模式。
- 查看有关 OpenShift Container Platform 安装和更新 流程的详细信息。
- 如果使用防火墙，则必须将其配置为允许集群需要访问的站点。

注意

如果您要配置代理，请务必也要查看此站点列表。

1.1.1. OpenShift Container Platform 对互联网和 Telemetry 的访问

在 OpenShift Container Platform 4.2 中，您需要访问互联网来安装集群。默认运行的 Telemetry 服务提供有关集群健康状况和成功更新的指标，这也需要访问互联网。如果您的集群连接到互联网，Telemetry 会自动运行，而且集群会注册到 Red Hat OpenShift Cluster Manager （OCM）。

确认 Red Hat OpenShift Cluster Manager 清单正确后，可以由 Telemetry 自动维护，也可以使用 OCM 手动维护，使用订阅监控 来跟踪帐户或多集群级别的 OpenShift Container Platform 订阅。

您必须具有以下互联网访问权限：

- 访问 Red Hat OpenShift Cluster Manager 页面，以下载安装程序并执行订阅管理。如果集群可以访问互联网，并且没有禁用 Telemetry，该服务会自动授权您的集群。
- 访问 Quay.io，以获取安装集群所需的软件包。
- 获取执行集群更新所需的软件包。

重要

如果您的集群无法直接访问互联网，则可以在置备的某些类基础架构上执行受限网络安装。在此过程中，您要下载所需的内容，并使用它在镜像 registry（mirror registry） 中填充安装集群并生成安装程序所需的软件包。对于某些安装类型，集群要安装到的环境不需要访问互联网。在更新集群之前，要更新 registry 镜像系统中的内容。

1.1.2. VMware vSphere 基础架构要求

您必须在 VMware vSphere 版本 6.5 或 6.7U2 或更高版本的实例上安装 OpenShift Container Platform 集群。

VMware 建议您将 Sphere Version 6.7 U2 或更高版本与 OpenShift Container Platform 集群搭配使用。vSphere 6.7U2 包括：

- 支持 VMware NSX-T

- 支持 vSAN 、VMFS 和 NFS（使用树状 VCP）

尽管支持 vSphere 6.5（硬件版本 13），但 OpenShift Container Platform 集群会受以下限制：

- 不支持 NSX-T SDN。
- 您必须使用 OpenShift Container Platform 支持的其他 SDN 或存储提供程序。

如果您使用 vSphere 版本 6.5 实例，请考虑升级到 6.7U2 后再安装 OpenShift Container Platform。

重要

您必须确保在安装 OpenShift Container Platform 前同步 ESXi 主机上的时间。请参阅 VMware 文档中的编辑主机时间配置。

1.1.3. 具有用户置备基础架构的集群的机器要求

对于含有用户置备的基础架构的集群，您必须部署所有所需的机器。

1.1.3.1. 所需的机器

最小的 OpenShift Container Platform 集群需要下列主机：

- 一个临时 bootstrap 机器
- 三台 control plane 或 master 机器
- 至少两台计算机器，也称为 worker 机器

注意

集群要求 bootstrap 机器在三台 control plane 机器上部署 OpenShift Container Platform 集群。您可在安装集群后删除 bootstrap 机器。

重要

要保持集群的高可用性，请将独立的物理主机用于这些集群机器。

bootstrap、control plane 以及计算（compute）机器必须使用 Red Hat Enterprise Linux CoreOS (RHCOS) 作为操作系统。

请注意，RHCOS 基于 Red Hat Enterprise Linux 8，并继承其所有硬件认证和要求。请查看Red Hat Enterprise Linux 技术功能及限制。

1.1.3.2. 网络连接要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置文件。在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

1.1.3.3. 最低资源要求

每台集群机器都必须满足以下最低要求：

机器	操作系统	vCPU	虚拟内存	存储
bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS 或 RHEL 7.6	2	8 GB	120 GB

1.1.3.4. 证书签名请求管理

在使用您置备的基础架构时，集群只能有限地访问自动机器管理，因此您必须提供一种在安装后批准集群证书签名请求 (CSR) 的机制。**kube-controller-manager** 只能批准 kubelet 客户端 CSR。 **machine-approver** 无法保证使用 kubelet 凭证请求的提供证书的有效性，因为它不能确认是正确的机器发出了该请求。您必须决定并实施一种方法，以验证 kubelet 提供证书请求的有效性并进行批准。

1.1.4. 创建用户置备的基础架构

在部署采用用户置备的基础架构的 OpenShift Container Platform 集群前，您必须创建底层基础架构。

先决条件

- 在为集群创建支持基础架构之前，请参阅OpenShift Container Platform 4.x Tested Integrations页。

流程

1. 配置 DHCP。
2. 提供所需的负载均衡器。
3. 配置机器的端口。
4. 配置 DNS。
5. 确保网络可以正常工作。

1.1.4.1. 用户置备的基础架构对网络的要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置。

在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

建议您使用 DHCP 服务器为集群进行长期机器管理。确保 DHCP 服务器已配置为向集群机器提供持久 IP 地址和主机名。

成功安装集群后，在每个 master 节点上运行的 Kubernetes API 服务器必须能够解析集群机器的节点名称。如果 API 服务器和 worker 节点位于不同的区域中，您可以配置默认 DNS 搜索区域，以便 API 服务器能够解析节点名称。另一种支持的方法是始终在节点对象和所有 DNS 请求中使用完全限定域名来指代主机。

您必须配置机器间的网络连接，以便集群组件进行通信。每台机器都必须能够解析集群中所有其他机器的主机名。

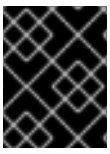
表 1.1. 所有机器到所有机器

协议	端口	描述
ICMP	N/A	网络可访问性测试
TCP	**9000**-**9999**	主机级别的服务，包括端口 **9100**-**9101** 上的节点导出器和端口 **9099** 上的 Cluster Version Operator。
	10250-**10259**	Kubernetes 保留的默认端口
	10256	openshift-sdn
UDP	**4789**	VXLAN 和 GENEVE
	6081	VXLAN 和 GENEVE
	9000-**9999**	主机级别的服务，包括端口 **9100**-**9101** 上的节点导出器。
TCP/UDP	**30000**-**32767**	Kubernetes NodePort

表 1.2. 要通过控制平面的所有机器

协议	端口	描述
TCP	**2379**-**2380**	etcd 服务器、对等和指标端口
	6443	Kubernetes API

网络拓扑要求
您为集群置备的基础架构必须满足下列网络拓扑要求。



重要

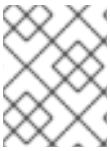
OpenShift Container Platform 要求所有节点都能访问互联网，以便为平台容器提取镜像并向红帽提供遥测数据。

负载**均衡器**

在安装 OpenShift Container Platform 之前，您必须置备两个 L4 负载均衡器。API 需要一个负载均衡器，默认的 Ingress Controller 需要第二个负载均衡器来为应用程序提供入口网络。

端口	机器	内部	外部	描述

端口	机器	内部	外部	描述
6443	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x	x	Kubernetes API 服务器
22623	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x		机器配置服务器
443	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTPS 流量
80	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTP 流量



注意

OpenShift Container Platform 集群需要正确配置入口路由器。control plane 初始化后，您必须配置入口路由器。

以太网适配器硬件地址要求
当为集群置备虚拟机时，为每个虚拟机配置的以太网接口必须使用 VMware 机构唯一识别符 (OUI) 分配范围中的 MAC 地址：

- **00:05:69:00:00:00** 到 **00:05:69:FF:FF:FF**
- **00:0c:29:00:00:00** 到 **00:0c:29:FF:FF:FF**
- **00:1c:14:00:00:00** 到 **00:1c:14:FF:FF:FF**
- **00:50:56:00:00:00** 到 **00:50:56:FF:FF:FF**

如果使用 VMware OUI 以外的 MAC 地址，集群安装将无法成功。

1.1.4.2. 用户置备 DNS 要求

采用用户置备的基础架构的 OpenShift Container Platform 集群需要以下 DNS 记录。在每一记录中，**<cluster_name>** 是集群名称，**<base_domain>** 则是您在 **install-config.yaml** 文件中指定的集群基域。完整的 DNS 记录采用如下格式: **<component>.<cluster_name>.<base_domain>.**。

表 1.3. 所需的 DNS 记录

组件	记录	描述
Kubernetes API	**api.<cluster_name>.<base_domain>.**	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。

组件	记录	描述
	api-int.<cluster_name>.<base_domain>.	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须可从集群内的所有节点解析。  **重要** API 服务器必须能够根据在 Kubernetes 中记录的主机名解析 worker 节点。如果无法解析节点名称，代理的 API 调用会失败，而且您无法从 Pod 检索日志。
Routes	***.apps.<cluster_name>.<base_domain>.**	通配符 DNS A/AAAA 或 CNAME 记录，指向以运行入口路由器 Pod 的机器（默认为 worker 节点）为目标的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。
etcd	**etcd-<index>.<cluster_name>.<base_domain>.**	OpenShift Container Platform 要求每个 etcd 实例的 DNS A/AAAA 记录指向托管实例的 control plane 机器。etcd 实例通过 **<index>** 值来区分，值的范围为 **0** 到 **n-1**，其中 **n** 是集群中 control plane 机器的数量。DNS 记录必须解析到 control plane 机器的单播 IPv4 地址，并且这些记录必须可以从集群中的所有节点解析。

组件	记录	描述
	_etcd-server-ssl._tcp.<cluster_name>.<base_domain>.	对于每台 control plane 机器，OpenShift Container Platform 还需要该机器上具有 etcd 服务器的 SRV DNS 记录，其优先级为 **0**，权重为 **10**，端口则为 **2380**。使用三台 control plane 机器的集群需要以下记录： # _service._proto.name. TTL class SRV priority weight port target. _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-0.<cluster_name>.<base_domain> _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-1.<cluster_name>.<base_domain> _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-2.<cluster_name>.<base_domain>

1.1.5. 生成 SSH 私钥并将其添加到代理中

如果要在集群上执行安装调试或灾难恢复，则必须为 **ssh-agent** 和安装程序提供 SSH 密钥。

注意

在生产环境中，您需要进行灾难恢复和调试。

您可以使用此密钥以 **core** 用户身份通过 SSH 连接到 master 节点。在部署集群时，此密钥会添加到 **core** 用户的 **~/.ssh/authorized_keys** 列表中。

注意

您必须使用一个本地密钥，而不要使用在特定平台上配置的密钥，如 AWS 密钥对。

流程

1. 如果还没有为计算机上免密码身份验证而配置的 SSH 密钥，请创建一个。例如，在使用 Linux 操作系统的计算机上运行以下命令：

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 指定 SSH 密钥的路径和文件名，如 **~/.ssh/id_rsa**。

运行此命令会在指定的位置生成不需要密码的 SSH 密钥。

2. 作为后台任务启动 **ssh-agent** 进程：

```
$ eval "$(ssh-agent -s)"

Agent pid 31874
```

3. 将 SSH 私钥添加到 **ssh-agent**：

```
$ ssh-add <path>/<file_name> 1

Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

1 指定 SSH 私钥的路径和文件名，如 **~/.ssh/id_rsa**

后续步骤

- 在安装 OpenShift Container Platform 时，为安装程序提供 SSH 公钥。如果在您置备的基础架构上安装集群，您必须将此密钥提供给集群的机器。

1.1.6. 获取安装程序

在安装 OpenShift Container Platform 之前，将安装文件下载到

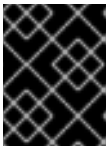
本地计算机。

先决条件

- 必须从使用 Linux 或 macOS 的计算机安装集群。
- 需要 500 MB 本地磁盘空间来下载安装程序。

流程

1. 访问 Red Hat OpenShift Cluster Manager 网站的 Infrastructure Provider 页面。如果您有红帽帐号，请使用自己的凭证登录。如果没有，请创建一个帐户。
2. 进入适用于您的安装类型的页面，下载您的操作系统的安装程序，并将文件放在要保存安装配置文件的目录中。。



重要

安装程序会在用来安装集群的计算机上创建若干文件。在完成集群安装后，您必须保留安装程序和安装程序所创建的文件。

3. 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 <installation_program>.tar.gz
```

4. 在 Red Hat OpenShift Cluster Manager 站点的 Pull Secret 页面中，下载您的安装 pull secret 的 **.txt** 文件。通过此 pull secret，您可以进行所含授权机构提供的服务的身份验证，这些服务包括为 OpenShift Container Platform 组件提供容器镜像的 Quay.io。

1.1.7. 手动创建安装配置文件

对于使用用户置备的基础架构的 OpenShift Container Platform 安装，您必须手动生成安装配置文件。

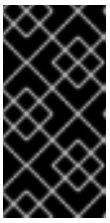
先决条件

- 获取 OpenShift Container Platform 安装程序和集群的访问令牌。

流程

1. 创建用来存储您所需的安装资产的安装目录：

```
$ mkdir <installation_directory>
```



重要

您必须创建目录。一些安装信息，如 bootstrap X.509 证书，有较短的过期间隔，因此不要重复使用安装目录。如果要重复使用另一个集群安装中的个别文件，可以将其复制到您的目录中。但是，一些安装数据的文件名可能会在发行版本之间有所改变。从 OpenShift Container Platform 老版本中复制安装文件时要格外小心。

2. 自定义以下 **install-config.yaml** 文件模板，并将它保存到 **<installation_directory>** 中。

注意

此配置文件必须命名为 **install-config.yaml**。

3. 备份 **install-config.yaml** 文件，以便用于安装多个集群。

重要

install-config.yaml 文件会在安装过程的下一步骤中消耗掉。现在必须备份它。

1.1.7.1. VMware vSphere install-config.yaml 文件示例

您可以自定义 **install-config.yaml** 文件，以指定有关 OpenShift Container Platform 集群平台的更多信息，或修改所需参数的值。

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
```

```
  hyperthreading: Enabled 5 6
  name: master
  replicas: 3 7
metadata:
  name: test 8
platform:
  vsphere:
    vcenter: your.vcenter.server 9
    username: username 10
    password: password 11
    datacenter: datacenter 12
    defaultDatastore: datastore 13
pullSecret: '{"auths": ...}' 14
sshKey: 'ssh-ed25519 AAAA...' 15
```

1 集群的基域。所有 DNS 记录都必须是这个基域的子域，并包含集群名称。

2 5 **controlPlane** 部分是一个单映射，但 compute 部分是一系列映射。为满足不同数据结构的要求，**compute** 部分的第一行必须以连字符 **-** 开头，**controlPlane** 部分的第一行则不可以连字符开头。虽然这两个部分目前都定义单个机器池，但未来的 OpenShift Container Platform 版本可能会支持在安装过程中定义多个计算池。只使用一个 control plane 池。

3 6 是否要启用或禁用并发多线程或**超线程**。默认情况下，启用并发多线程以提高机器内核的性能。您可以通过将参数值设为 **Disabled** 来禁用。如果您在某些集群机器上禁用并发多线程，则必须在所有集群机器上禁用。

重要

如果禁用并发多线程，请确保在容量规划时考虑到机器性能可能会显著降低的问题。如果您禁用并发多线程，则计算机必须至少使用 8 个 CPU 和 32GB RAM。

4 **replicas** 参数的值必须设置为 **0**。此参数控制集群为您创建和管理的 worker 数量，使用用户置备的基础架构时集群不会执行这些功能。在完成 OpenShift Container Platform 安装前，您必须手动为集群部署 worker 机器。

7 您添加到集群的 control plane 机器数量。由于集群将这个值用作集群中 etcd 端点的数量，因此该值必须与您部署的 control plane 机器数量匹配。

8 您在 DNS 记录中指定的集群名称。

9 vCenter 服务器的完全限定主机名或 IP 地址。

10 用于访问服务器的用户名。此用户必须至少在 vSphere 中有静态或动态持久性卷部署所需的角色和权限。

11 与 vSphere 用户关联的密码。

12 vSphere 数据中心。

13 要使用的默认 vSphere 数据存储。

14 从 Red Hat OpenShift Cluster Manager 站点的 Pull Secret 页面中获取的 pull secret。通过此 pull secret，您可以进行所含授权机构提供的服务的身份验证，这些服务包括为 OpenShift Container Platform 组件提供容器镜像的 Quay.io。

15 Red Hat Enterprise Linux CoreOS (RHCOS) 中 **core** 用户的默认 SSH 密钥的公钥部分。

注意

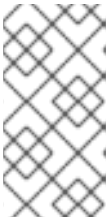
对于您要在其上执行安装调试或灾难恢复的生产环境 OpenShift Container Platform 集群，请指定 **ssh-agent** 进程使用的 SSH 密钥。

1.1.7.2. 在安装过程中配置集群范围代理

生产环境可能会拒绝直接访问互联网，而是提供 HTTP 或 HTTPS 代理。您可以通过在 **install-config.yaml** 文件中配置代理设置，将新的 OpenShift Container Platform 集群配置为使用代理。

先决条件

- 现有的 **install-config.yaml** 文件。
- 查看集群需要访问的站点，并决定是否需要绕过代理。默认情况下代理所有集群出口流量，包括对托管云供应商 API 的调用。若有需要，将站点添加到 Proxy 对象的 **spec.noProxy** 字段来绕过代理服务器。



注意

Proxy 对象的 **status.noProxy** 字段默认填充实例元数据端点 (**169.254.169.254**)，以及您的安装配置中 **networking.machineCIDR**、**networking.clusterNetwork.cidr** 和 **networking.serviceNetwork** 字段的值。

流程

1. 编辑 **install-config.yaml** 文件并添加代理设置。例如：

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: http://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
...
```

1 用于创建集群外 HTTP 连接的代理 URL。URL 必须是 **http**。

2 用于创建集群外 HTTPS 连接的代理 URL。如果未指定此字段，**httpProxy** 会同时用于 HTTP 和 HTTPS 连接。URL 方案必须是 **http**；目前不支持 **https**。

3 要排除代理的目标域名、域、IP 地址或其他网络 CIDR 的逗号分隔列表。域之前加上前缀 **.** 可包含该域的所有子域。使用 * 可对所有目的地绕过所有代理。您必须包含 vCenter 的 IP 地址以及用于其机器的 IP 范围。

4 如果提供，安装程序会在 **openshift-config** 命名空间中生成名为 **user-ca-bundle** 的 ConfigMap，其包含代理 HTTPS 连接所需的一个或多个额外 CA 证书。然后，Cluster

注意

安装程序不支持代理的 **readinessEndpoints** 字段。

2. 保存该文件，并在安装 OpenShift Container Platform 时引用。

安装程序会创建一个名为 **cluster** 的集群范围代理，该代理使用提供的 **install-config.yaml** 文件中的代理设置。如果没有提供代理设置，仍然会创建 **cluster** Proxy 对象，但它会有一个零 **spec**。

注意

只支持名为 **cluster** 的 Proxy 对象，且无法创建额外的代理。

1.1.8. 创建 Kubernetes 清单和 Ignition 配置文件

由于您必须修改一些集群定义文件并要手动启动集群机器，因此您必须生成 Kubernetes 清单和 Ignition 配置文件，集群需要这两项来创建其机器。

重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须完成集群安装，并使集群以非降级状态运行 24 小时，以确保完成第一次证书轮转。

先决条件

- 获取 OpenShift Container Platform 安装程序。
- 创建 **install-config.yaml** 安装配置文件。

流程

1. 为集群生成 Kubernetes 清单：

```
$ ./openshift-install create manifests --dir=<installation_directory> 1

WARNING There are no compute nodes specified. The cluster will not fully initialize without
compute nodes.
INFO Consuming "Install Config" from target directory
```

1 对于 **<installation_directory>**，请指定含有您创建的 **install-config.yaml** 文件的安装目录。

由于您稍后会在安装过程中自行创建计算机器，因此可以忽略这个警告。

2. 修改 **manifests/cluster-scheduler-02-config.yml** Kubernetes 清单文件，以防止在 control plane 机器上调度 Pod：

   a. 打开 **manifests/cluster-scheduler-02-config.yml** 文件。

b. 找到 **mastersSchedulable** 参数，并将其值设为 **False**。

c. 保存并退出文件。

注意

目前，由于 Kubernetes 限制，入口负载均衡器将无法访问在 control plane 机器上运行的路由器 Pod。以后的 OpenShift Container Platform 次要版本中可能不需要这一步骤。

3. 获取 Ignition 配置文件：

```
$ ./openshift-install create ignition-configs --dir=<installation_directory> 1
```

1 对于 **<installation_directory>**，请指定相同的安装目录。

该目录中将生成以下文件：

```
.
├── auth
│   ├── kubeadmin-password
│   └── kubeconfig
├── bootstrap.ign
├── master.ign
├── metadata.json
└── worker.ign
```

1.1.9. 在 vSphere 中创建 Red Hat Enterprise Linux CoreOS (RHCOS) 机器

在 VMware vSphere 上安装包含用户置备基础架构的集群前，您必须在 vSphere 主机上创建 RHCOS 机器供其使用。

先决条件

- 获取集群的 Ignition 配置文件。
- 具有 HTTP 服务器的访问权限，以便您可从计算机进行访问，并且您创建的机器也可访问此服务器。
- 创建 vSphere 集群。

流程

1. 将名为 **<installation_directory>/bootstrap.ign** 的 bootstrap Ignition 配置文件上传到 HTTP 服务器，该配置文件是由安装程序创建的。记下此文件的 URL。
   您必须托管 bootstrap Ignition 配置文件，因为它太大而无法放在 vApp 属性中。
2. 将 bootstrap 节点的以下辅助 Ignition 配置文件保存到计算机中，存为 **<installation_directory>/append-bootstrap.ign**。

```
{
  "ignition": {
    "config": {
```

```
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

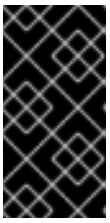
1 指定您托管的 bootstrap Ignition 配置文件的 URL。

为 bootstrap 机器创建虚拟机 (VM) 时，您要使用此 Ignition 配置文件。

3. 将 master、worker 和辅助 Bootstrap Ignition 配置文件转换为 Base64 编码。例如，如果您使用 Linux 操作系统，可以使用 **base64** 命令来编码这些文件。

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. 从红帽客户门户上的产品下载页面或 RHCOS 镜像页面，获取 RHCOS OVA 镜像。



重要

RHCOS 镜像可能不会随着 OpenShift Container Platform 的每一发行版本都有改变。您必须下载一个最高版本的镜像，其版本号应小于或等于您安装的 OpenShift Container Platform 版本。如果可用，请使用与 OpenShift Container Platform 版本匹配的镜像版本。

文件名包含 OpenShift Container Platform 版本号，格式为 **rhcos-<version>-<architecture>-vmware.ova**。

5. 在 vSphere 客户端中，在数据中心中创建一个文件夹来存储您的虚拟机。

   a. 点击 **VMs and Templates** 视图。

   b. 右键点击您的数据中心名称。

   c. 点击 **New Folder → New VM and Template Folder**。

   d. 在显示的窗口中输入文件夹名称。文件夹名称必须与您在 **install-config.yaml** 文件中指定的集群名称匹配。

6. 在 vSphere 客户端中，为 OVA 镜像创建一个模板。

注意

在以下步骤中，您要将同一模板用于所有集群机器，并针对您在置备虚拟机时指定的机器类型提供 Ignition 配置文件的位置。

a. 在 **Hosts and Clusters** 选项卡中，右键点击集群名称并点击 **Deploy OVF Template**。

b. 在 **Select an OVF** 选项卡中，指定您下载的 RHCOS OVA 文件的名称。

c. 在 **Select a name and folder**选项卡中，设置 **Virtual machine name**，如 RHCOS，再点击 vSphere 集群的名称，然后选择您在上一步中创建的文件夹。

d. 在 **Select a compute resource** 选项卡中，点击您的 vSphere 集群名称。

e. 在 **Select storage** 选项卡中，配置虚拟机的存储选项。

- 选择 **Thin Provision**。
- 选择您在 **install-config.yaml** 文件中指定的数据存储。

f. 在 **Select network** 选项卡中，指定您为集群配置的网络（如果可用）。

g. 如果您计划将同一模板用于所有集群机器类型，请不要在 **Customize template** 选项卡中指定值。

7. 部署模板后，为集群中的机器部署虚拟机。

a. 右键点击模板的名称，再点击 **Clone → Clone to Virtual Machine**。

b. 在 **Select a name and folder**选项卡中，指定虚拟机的名称。名称中可以包括机器类型，如 **control-plane-0** 或 **compute-1**。

c. 在 **Select a name and folder**选项卡中，选择您为集群创建的文件夹名称。

d. 在 **Select a compute resource** 选项卡中，选择数据中心中的主机名称。

e. 可选：在 **Select storage** 选项卡中，自定义存储选项。

f. 在 **Select clone options** 中，选择 **Customize this virtual machine's hardware**。

g. 在 **Customize hardware** 选项卡中，点击 **VM Options → Advanced**。

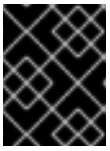
- 可选：在出现集群性能问题时，从 **Latency Sensitivity** 列表中选择 **High**。
- 点击 **Edit Configuration**，然后在 **Configuration Parameters** 窗口中点击 **Add Configuration Params**。定义以下参数名称和值：
  - **guestinfo.ignition.config.data**：粘贴此机器类型的 base64 编码 Ignition 配置文件的内容。
  - **guestinfo.ignition.config.data.encoding**：指定 **base64**。
  - **disk.EnableUUID**：指定 **TRUE**。
- 另外，也在打开虚拟机电源前通过 vApp 属性添加：
  - 导航到 vCenter 服务器清单中的某一虚拟机。

- 在 **Configure** 选项卡中，展开 **Settings** 并选择 **vApp options**。
- 向下滚动，并在 **Properties** 下应用上述配置。

h. 在 **Customize hardware** 选项卡的 **Virtual Hardware** 面板中，根据需要修改指定的值。确保 RAM、CPU 和磁盘存储的数量满足机器类型的最低要求。

i. 完成配置并打开虚拟机电源。

8. 对于每台机器，按照前面的步骤为集群创建其余的机器。



重要

此刻您必须创建 bootstrap 和 control plane 机器。由于计算机器上已默认部署了一些 Pod，因此在安装集群前，还要创建至少两台计算机器。

1.1.10. 安装 CLI

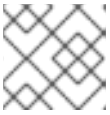
为了可以使用命令行界面与 OpenShift Container Platform 进行交互，您需要安装 CLI。

重要

如果安装了旧版本的 **oc**，则无法使用 OpenShift Container Platform 4.2 中的所有命令。下载并安装新版本的 **oc**。

流程

1. 在 Red Hat OpenShift Cluster Manager 站点的 Infrastructure Provider 页面中导航至您的安装类型页面，并点击 **Download Command-line Tools**。
2. 点您的操作系统和系统架构的文件夹，然后点压缩文件。



注意

您可在 Linux、Windows 或 macOS 上安装 **oc**。

3. 将文件保存到文件系统。
4. 展开压缩文件。
5. 把它放到 **PATH** 中的一个目录下。

安装 CLI 后，就可以使用**oc**命令：

```
$ oc <command>
```

1.1.11. 创建集群

要创建 OpenShift Container Platform 集群，请等待您通过安装程序生成的 Ignition 配置文件所置备的机器上完成 bootstrap 过程。

先决条件

- 为集群创建所需的基础架构。
- 已获得安装程序并为集群生成了 Ignition 配置文件。
- 已使用 Ignition 配置文件为集群创建 RHCOS 机器。
- 您的机器能够直接访问互联网。

流程

1. 监控 bootstrap 过程：

```
$ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
    --log-level=info 2
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.14.6+c4799753c up
INFO Waiting up to 30m0s for the bootstrap-complete event...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2 要查看不同的安装详情，请指定 **warn**、**debug** 或 **error**，而不要指定 **info**。

Kubernetes API 服务器提示已在 control plane 机器上完成 bootstrap 时，命令运行成功。

2. bootstrap 过程完成后，请从负载均衡器中删除 bootstrap 机器。

重要

此时您必须从负载均衡器中删除 bootstrap 机器。您还可以删除或重新格式化机器本身。

1.1.12. 登录集群

您可以通过导出集群 **kubeconfig** 文件，以默认系统用户身份登录集群。**kubeconfig** 文件包含关于集群的信息，供 CLI 用于将客户端连接到正确集群和 API 服务器。该文件特只适用于一个特定的集群，在 OpenShift Container Platform 安装过程中创建。

先决条件

- 部署 OpenShift Container Platform 集群。
- 安装 **oc** CLI。

流程

1. 导出 **kubeadmin** 凭证：

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2. 使用导出的配置，验证能否成功运行 **oc** 命令：

```
$ oc whoami
system:admin
```

1.1.13. 批准机器的 CSR

将机器添加到集群时，系统会为添加的每台机器生成两个待处理证书签名请求 (CSR)。您必须确认这些 CSR 已获得批准，或根据需要自行批准。

先决条件

- 您已将机器添加到集群中。

流程

1. 确认集群可以识别这些机器：

```
$ oc get nodes

NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.14.6+c4799753c
master-1  Ready     master  63m  v1.14.6+c4799753c
master-2  Ready     master  64m  v1.14.6+c4799753c
worker-0  NotReady  worker  76s  v1.14.6+c4799753c
worker-1  NotReady  worker  70s  v1.14.6+c4799753c
```

   输出将列出您创建的所有机器。

2. 检查待处理证书签名请求 (CSR)，并确保您添加到集群中的每一机器都有状态为 **Pending** 或 **Approved** 的客户端和服务器请求：

```
$ oc get csr

NAME        AGE     REQUESTOR                                                   CONDITION
csr-8b2br   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending (1)
csr-8vnps   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending
csr-bfd72   5m26s   system:node:ip-10-0-50-126.us-east-2.compute.internal
Pending (2)
csr-c57lv   5m26s   system:node:ip-10-0-95-157.us-east-2.compute.internal
Pending
...
```

   (1) 客户端请求 CSR。

   (2) 服务器请求 CSR。

   在本例中，两台机器加入了集群。您可能会在列表中看到更多已批准的 CSR。

3. 如果 CSR 没有获得批准，请在所添加机器的所有待处理 CSR 都处于 **Pending** 状态后，为您的集群机器批准这些 CSR：

注意

由于 CSR 会自动轮转，因此请在将机器添加到集群后一小时内批准您的 CSR。如果没有在一小时内批准，证书将会轮转，每个节点将会存在多个证书。您必须批准所有这些证书。批准初始 CSR 后，集群的 **kube-controller-manager** 会自动批准后续的节点客户端 CSR。您必须实施一个方法来自动批准 kubelet 提供的证书请求。

- 若要单独批准，请对每个有效的 CSR 运行以下命令：

```
$ oc adm certificate approve <csr_name> 1
```

1 **<csr_name>** 是当前 CSR 列表中 CSR 的名称。

- 要批准所有待处理的 CSR，请运行以下命令：

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}
{{end}}{{end}}' | xargs oc adm certificate approve
```

1.1.14. 初始 Operator 配置

在 control plane 初始化后，您必须立即配置一些 Operator 以便它们都可用。

先决条件

- 您的 control plane 已初始化。

流程

1. 观察集群组件上线：

```
$ watch -n5 oc get clusteroperators

NAME                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication             4.2.0     True        False         False     69s
cloud-credential           4.2.0     True        False         False     12m
cluster-autoscaler         4.2.0     True        False         False     11m
console                    4.2.0     True        False         False     46s
dns                        4.2.0     True        False         False     11m
image-registry             4.2.0     False       True          False     5m26s
ingress                    4.2.0     True        False         False     5m36s
kube-apiserver             4.2.0     True        False         False     8m53s
kube-controller-manager    4.2.0     True        False         False     7m24s
kube-scheduler             4.2.0     True        False         False     12m
machine-api                4.2.0     True        False         False     12m
machine-config             4.2.0     True        False         False     7m36s
marketplace                4.2.0     True        False         False     7m54m
monitoring                 4.2.0     True        False         False     7h54s
network                    4.2.0     True        False         False     5m9s
node-tuning                4.2.0     True        False         False     11m
openshift-apiserver        4.2.0     True        False         False     11m
openshift-controller-manager   4.2.0     True        False         False     5m943s
```

```
openshift-samples                  4.2.0     True        False         False      3m55s
operator-lifecycle-manager         4.2.0     True        False         False      11m
operator-lifecycle-manager-catalog 4.2.0     True        False         False      11m
service-ca                         4.2.0     True        False         False      11m
service-catalog-apiserver          4.2.0     True        False         False      5m26s
service-catalog-controller-manager 4.2.0     True        False         False      5m25s
storage                            4.2.0     True        False         False      5m30s
```

2. 配置不可用的 Operator。

1.1.14.1. 镜像 registry 存储配置

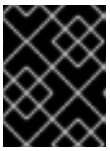
如果 **image-registry** Operator 不可用，您必须为其配置存储。提供了配置 PersistentVolume 的说明，这是生产集群所需要的；也提供了将空目录配置为存储位置的说明，这仅适用于非生产集群。

1.1.14.1.1. 为 VMware vSphere 配置 registry 存储

作为集群管理员，在安装后需要配置 registry 来使用存储。

先决条件

- 具有 Cluster Administrator 权限
- VMware vSphere上有一个集群。
- 具有**ReadWriteMany**访问模式的持久性存储卷（PV），例如**NFS** 。



重要

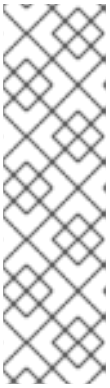
vSphere卷不支持**ReadWriteMany**访问模式。您必须使用其他存储后端（例如**NFS** ）来配置 registry 注存储。

- 必须有“ 100Gi”容量。

流程

1. 为了配置 registry 使用存储，需要修改 **configs.imageregistry/cluster** 资源中的 **spec.storage.pvc** 。
2. 验证您没有 registry Pod：

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



注意

如果存储类型为 **emptyDIR**，则副本数不能超过 **1**。如果存储类型为**NFS** ，并且希望通过设置 **replica>1**来扩展 registry Pod，则必须启用**no_wdelay**挂载选项。例如：

```
# cat /etc/exports
/mnt/data *(rw,sync,no_wdelay,no_root_squash,insecure,fsid=0)
sh-4.3# exportfs -rv
exporting *:/mnt/data
```

3. 检查 registry 配置：

```
$ oc edit configs.imageregistry.operator.openshift.io

storage:
  pvc:
    claim:
```

将 **claim** 字段留空以允许自动创建一个 **image-registry-storage** PVC。

4. 可选：在 PV 中添加新存储类：

   a. 创建 PV：

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    ReadWriteMany
    capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

   b. 创建 PVC：

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

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

最后，添加 PVC 的名称：

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc 1
```

1 通过创建自定义 PVC，您可以将 **claim** 字段留空以用于默认自动创建 **image-registry-storage** PVC。

5. 检查 **clusteroperator** 的状态：

```
$ oc get clusteroperator image-registry
```

1.1.14.1.2. 在非生产集群中配置镜像 registry 存储

您必须为 Image Registry Operator 配置存储。对于非生产集群，您可以将镜像 registry 设置为空目录。如果您这样做，重启 registry 后会丢失所有镜像。

流程

- 将镜像 registry 存储设置为空目录：

```
$ oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

警告

仅可为非生产集群配置这个选项。

如果在 Image Registry Operator 初始化其组件前运行此命令，**oc patch** 命令会失败并显示以下错误：

```
Error from server (NotFound): configs.imageregistry.operator.openshift.io "cluster" not found
```

等待几分钟，然后再次运行该命令。

1.1.15. 在用户置备的基础架构上完成安装

完成 Operator 配置后，可以在您提供的基础架构上完成集群安装。

先决条件

- 您的 control plane 已初始化。
- 已完成初始 Operator 配置。

流程

1. 确认所有集群组件都已上线：

```
$ watch -n5 oc get clusteroperators

NAME                                 VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                       4.2.0     True        False         False     10m
cloud-credential                     4.2.0     True        False         False     22m
cluster-autoscaler                   4.2.0     True        False         False     21m
console                              4.2.0     True        False         False     10m
dns                                  4.2.0     True        False         False     21m
image-registry                       4.2.0     True        False         False     16m
ingress                              4.2.0     True        False         False     16m
kube-apiserver                       4.2.0     True        False         False     19m
kube-controller-manager              4.2.0     True        False         False     18m
kube-scheduler                       4.2.0     True        False         False     22m
machine-api                          4.2.0     True        False         False     22m
machine-config                       4.2.0     True        False         False     18m
marketplace                          4.2.0     True        False         False     18m
monitoring                           4.2.0     True        False         False     18m
network                              4.2.0     True        False         False     16m
node-tuning                          4.2.0     True        False         False     21m
openshift-apiserver                  4.2.0     True        False         False     21m
openshift-controller-manager         4.2.0     True        False         False     17m
openshift-samples                    4.2.0     True        False         False     14m
operator-lifecycle-manager           4.2.0     True        False         False     21m
operator-lifecycle-manager-catalog   4.2.0     True        False         False     21m
service-ca                           4.2.0     True        False         False     21m
service-catalog-apiserver            4.2.0     True        False         False     16m
service-catalog-controller-manager   4.2.0     True        False         False     16m
storage                              4.2.0     True        False         False     16m
```

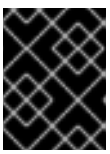
当所有集群 Operator 状态都是 **AVAILABLE** 时，您可以完成安装。

2. 监控集群完成：

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
INFO Waiting up to 30m0s for the cluster to initialize...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

Cluster Version Operator 完成从 Kubernetes API 服务器部署 OpenShift Container Platform 集群时，命令运行成功。



重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须以非降级状态持续运行集群 24 小时，以确保完成第一次证书轮转。

3. 确认 Kubernetes API 服务器正在与 Pod 通信。

   a. 要查看所有 Pod 的列表，请使用以下命令：

```
$ oc get pods --all-namespaces

NAMESPACE                  NAME                                        READY   STATUS
RESTARTS   AGE
openshift-apiserver-operator    openshift-apiserver-operator-85cb746d55-zqhs8   1/1
Running    1          9m
openshift-apiserver        apiserver-67b9g                             1/1     Running    0
3m
openshift-apiserver        apiserver-ljcmx                             1/1     Running    0
1m
openshift-apiserver        apiserver-z25h4                             1/1     Running    0
2m
openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8       1/1
Running    0          5m
...
```

b. 使用以下命令，查看上一命令的输出中所列 Pod 的日志：

```
$ oc logs <pod_name> -n <namespace> 1
```

1 指定 Pod 名称和命名空间，如上一命令的输出中所示。

如果 Pod 日志显示，Kubernetes API 服务器可以与集群机器通信。

后续步骤

- 自定义集群。
- 若有需要，您可以选择不使用远程健康报告。

1.2. 使用网络自定义在 VSPHERE 上安装集群

在 OpenShift Container Platform 版本 4.2 中，您可以使用自定义的网络配置选项在 VMware vSphere 环境中安装集群。通过自定义网络配置，您的集群可以与环境中现有的 IP 地址分配共存，并与现有的 MTU 和 VXLAN 配置集成。

大部分网络配置参数必须在安装过程中设置，只有 **kubeProxy** 配置参数可以在运行的集群中修改。

先决条件

- 为集群置备持久性存储。若要部署私有镜像 registry，您的存储必须提供 ReadWriteMany 访问模式。
- 查看有关 OpenShift Container Platform 安装和更新 流程的详细信息。
- 如果使用防火墙，必须将其配置为可以访问 Red Hat Insights。

1.2.1. OpenShift Container Platform 对互联网和 Telemetry 的访问

在 OpenShift Container Platform 4.2 中，您需要访问互联网来安装集群。默认运行的 Telemetry 服务提供有关集群健康状况和成功更新的指标，这也需要访问互联网。如果您的集群连接到互联网，Telemetry 会自动运行，而且集群会注册到 Red Hat OpenShift Cluster Manager （OCM）。

确认 Red Hat OpenShift Cluster Manager 清单正确后，可以由 Telemetry 自动维护，也可以使用 OCM 手动维护，使用订阅监控 来跟踪帐户或多集群级别的 OpenShift Container Platform 订阅。

您必须具有以下互联网访问权限：

- 访问 Red Hat OpenShift Cluster Manager 页面，以下载安装程序并执行订阅管理。如果集群可以访问互联网，并且没有禁用 Telemetry，该服务会自动授权您的集群。
- 访问 Quay.io，以获取安装集群所需的软件包。
- 获取执行集群更新所需的软件包。

重要

如果您的集群无法直接访问互联网，则可以在置备的某些类基础架构上执行受限网络安装。在此过程中，您要下载所需的内容，并使用它在镜像 registry（mirror registry） 中填充安装集群并生成安装程序所需的软件包。对于某些安装类型，集群要安装到的环境不需要访问互联网。在更新集群之前，要更新 registry 镜像系统中的内容。

1.2.2. VMware vSphere 基础架构要求

您必须在 VMware vSphere 版本 6.5 或 6.7U2 或更高版本的实例上安装 OpenShift Container Platform 集群。

VMware 建议您将 Sphere Version 6.7 U2 或更高版本与 OpenShift Container Platform 集群搭配使用。vSphere 6.7U2 包括：

- 支持 VMware NSX-T
- 支持 vSAN 、VMFS 和 NFS（使用树状 VCP）

尽管支持 vSphere 6.5（硬件版本 13），但 OpenShift Container Platform 集群会受以下限制：

- 不支持 NSX-T SDN。
- 您必须使用 OpenShift Container Platform 支持的其他 SDN 或存储提供程序。

如果您使用 vSphere 版本 6.5 实例，请考虑升级到 6.7U2 后再安装 OpenShift Container Platform。

重要

您必须确保在安装 OpenShift Container Platform 前同步 ESXi 主机上的时间。请参阅 VMware 文档中的编辑主机时间配置。

1.2.3. 具有用户置备基础架构的集群的机器要求

对于含有用户置备的基础架构的集群，您必须部署所有所需的机器。

1.2.3.1. 所需的机器

最小的 OpenShift Container Platform 集群需要下列主机：

- 一个临时 bootstrap 机器
- 三台 control plane 或 master 机器

- 至少两台计算机器，也称为 worker 机器

注意

集群要求 bootstrap 机器在三台 control plane 机器上部署 OpenShift Container Platform 集群。您可在安装集群后删除 bootstrap 机器。

重要

要保持集群的高可用性，请将独立的物理主机用于这些集群机器。

bootstrap、control plane 以及计算（compute）机器必须使用 Red Hat Enterprise Linux CoreOS (RHCOS) 作为操作系统。

请注意，RHCOS 基于 Red Hat Enterprise Linux 8，并继承其所有硬件认证和要求。请查看Red Hat Enterprise Linux 技术功能及限制。

1.2.3.2. 网络连接要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置文件。在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

1.2.3.3. 最低资源要求

每台集群机器都必须满足以下最低要求：

机器	操作系统	vCPU	虚拟内存	存储
bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS 或 RHEL 7.6	2	8 GB	120 GB

1.2.3.4. 证书签名请求管理

在使用您置备的基础架构时，集群只能有限地访问自动机器管理，因此您必须提供一种在安装后批准集群证书签名请求 (CSR) 的机制。**kube-controller-manager** 只能批准 kubelet 客户端 CSR。 **machine-approver** 无法保证使用 kubelet 凭证请求的提供证书的有效性，因为它不能确认是正确的机器发出了该请求。您必须决定并实施一种方法，以验证 kubelet 提供证书请求的有效性并进行批准。

1.2.4. 创建用户置备的基础架构

在部署采用用户置备的基础架构的 OpenShift Container Platform 集群前，您必须创建底层基础架构。

先决条件

- 在为集群创建支持基础架构之前，请参阅OpenShift Container Platform 4.x Tested Integrations页。

流程

1. 配置 DHCP。
2. 提供所需的负载均衡器。
3. 配置机器的端口。
4. 配置 DNS。
5. 确保网络可以正常工作。

1.2.4.1. 用户置备的基础架构对网络的要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置。

在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

建议您使用 DHCP 服务器为集群进行长期机器管理。确保 DHCP 服务器已配置为向集群机器提供持久 IP 地址和主机名。

成功安装集群后，在每个 master 节点上运行的 Kubernetes API 服务器必须能够解析集群机器的节点名称。如果 API 服务器和 worker 节点位于不同的区域中，您可以配置默认 DNS 搜索区域，以便 API 服务器能够解析节点名称。另一种支持的方法是始终在节点对象和所有 DNS 请求中使用完全限定域名来指代主机。

您必须配置机器间的网络连接，以便集群组件进行通信。每台机器都必须能够解析集群中所有其他机器的主机名。

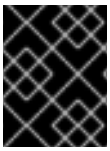
表 1.4. 所有机器到所有机器

协议	端口	描述
ICMP	N/A	网络可访问性测试
TCP	**9000**-**9999**	主机级别的服务，包括端口 **9100**-**9101** 上的节点导出器和端口 **9099** 上的 Cluster Version Operator。
	10250-**10259**	Kubernetes 保留的默认端口
	10256	openshift-sdn
UDP	**4789**	VXLAN 和 GENEVE
	6081	VXLAN 和 GENEVE
	9000-**9999**	主机级别的服务，包括端口 **9100**-**9101** 上的节点导出器。
TCP/UDP	**30000**-**32767**	Kubernetes NodePort

表 1.5. 要通过控制平面的所有机器

协议	端口	描述
TCP	**2379-2380**	etcd 服务器、对等和指标端口
	6443	Kubernetes API

网络拓扑要求
您为集群置备的基础架构必须满足下列网络拓扑要求。



重要

OpenShift Container Platform 要求所有节点都能访问互联网，以便为平台容器提取镜像并向红帽提供遥测数据。

负载**均衡器**

在安装 OpenShift Container Platform 之前，您必须置备两个 L4 负载均衡器。API 需要一个负载均衡器，默认的 Ingress Controller 需要第二个负载均衡器来为应用程序提供入口网络。

端口	机器	内部	外部	描述
6443	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x	x	Kubernetes API 服务器
22623	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x		机器配置服务器
443	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTPS 流量
80	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTP 流量

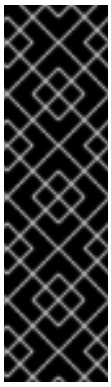
注意

OpenShift Container Platform 集群需要正确配置入口路由器。control plane 初始化后，您必须配置入口路由器。

1.2.4.2. 用户置备 DNS 要求

采用用户置备的基础架构的 OpenShift Container Platform 集群需要以下 DNS 记录。在每一记录中，**<cluster_name>** 是集群名称，**<base_domain>** 则是您在 **install-config.yaml** 文件中指定的集群基域。完整的 DNS 记录采用如下格式: **<component>.<cluster_name>.<base_domain>.**。

表 1.6. 所需的 DNS 记录

组件	记录	描述
Kubernetes API	**api.<cluster_name>.<base_domain>.**	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。
	api-int.<cluster_name>.<base_domain>.	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须可从集群内的所有节点解析。  **重要** API 服务器必须能够根据在 Kubernetes 中记录的主机名解析 worker 节点。如果无法解析节点名称，代理的 API 调用会失败，而且您无法从 Pod 检索日志。
Routes	***.apps.<cluster_name>.<base_domain>.**	通配符 DNS A/AAAA 或 CNAME 记录，指向以运行入口路由器 Pod 的机器（默认为 worker 节点）为目标的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。
etcd	**etcd-<index>.<cluster_name>.<base_domain>.**	OpenShift Container Platform 要求每个 etcd 实例的 DNS A/AAAA 记录指向托管实例的 control plane 机器。etcd 实例通过 **<index>** 值来区分，值的范围为 **0** 到 **n-1**，其中 **n** 是集群中 control plane 机器的数量。DNS 记录必须解析到 control plane 机器的单播 IPv4 地址，并且这些记录必须可以从集群中的所有节点解析。

组件	记录	描述
	_etcd-server-ssl._tcp.<cluster_name>.<base_domain>.	对于每台 control plane 机器，OpenShift Container Platform 还需要该机器上具有 etcd 服务器的 SRV DNS 记录，其优先级为 **0**，权重为 **10**，端口则为 **2380**。使用三台 control plane 机器的集群需要以下记录： # _service._proto.name. TTL class SRV priority weight port target. _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-0.<cluster_name>.<base_domain> _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-1.<cluster_name>.<base_domain> _etcd-server-ssl._tcp.<cluster_name>.<base_domain>. 86400 IN SRV 0 10 2380 etcd-2.<cluster_name>.<base_domain>

1.2.5. 生成 SSH 私钥并将其添加到代理中

如果要在集群上执行安装调试或灾难恢复，则必须为 **ssh-agent** 和安装程序提供 SSH 密钥。

注意

在生产环境中，您需要进行灾难恢复和调试。

您可以使用此密钥以 **core** 用户身份通过 SSH 连接到 master 节点。在部署集群时，此密钥会添加到 **core** 用户的 **~/.ssh/authorized_keys** 列表中。

注意

您必须使用一个本地密钥，而不要使用在特定平台上配置的密钥，如 AWS 密钥对。

流程

1. 如果还没有为计算机上免密码身份验证而配置的 SSH 密钥，请创建一个。例如，在使用 Linux 操作系统的计算机上运行以下命令：

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 指定 SSH 密钥的路径和文件名，如 **~/.ssh/id_rsa**。

运行此命令会在指定的位置生成不需要密码的 SSH 密钥。

2. 作为后台任务启动 **ssh-agent** 进程：

```
$ eval "$(ssh-agent -s)"

Agent pid 31874
```

3. 将 SSH 私钥添加到 **ssh-agent**：

```
$ ssh-add <path>/<file_name> 1

Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

1 指定 SSH 私钥的路径和文件名，如 **~/.ssh/id_rsa**

后续步骤

- 在安装 OpenShift Container Platform 时，为安装程序提供 SSH 公钥。

1.2.6. 获取安装程序

在安装 OpenShift Container Platform 之前，将安装文件下载到

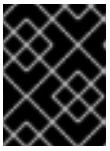
本地计算机。

先决条件

- 必须从使用 Linux 或 macOS 的计算机安装集群。
- 需要 500 MB 本地磁盘空间来下载安装程序。

流程

1. 访问 Red Hat OpenShift Cluster Manager 网站的 Infrastructure Provider 页面。如果您有红帽帐号，请使用自己的凭证登录。如果没有，请创建一个帐户。
2. 进入适用于您的安装类型的页面，下载您的操作系统的安装程序，并将文件放在要保存安装配置文件的目录中。。



重要

安装程序会在用来安装集群的计算机上创建若干文件。在完成集群安装后，您必须保留安装程序和安装程序所创建的文件。

3. 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 <installation_program>.tar.gz
```

4. 在 Red Hat OpenShift Cluster Manager 站点的 Pull Secret 页面中，下载您的安装 pull secret 的 **.txt** 文件。通过此 pull secret，您可以进行所含授权机构提供的服务的身份验证，这些服务包括为 OpenShift Container Platform 组件提供容器镜像的 Quay.io。

1.2.7. 手动创建安装配置文件

对于使用用户置备的基础架构的 OpenShift Container Platform 安装，您必须手动生成安装配置文件。

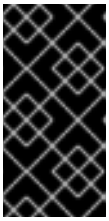
先决条件

- 获取 OpenShift Container Platform 安装程序和集群的访问令牌。

流程

1. 创建用来存储您所需的安装资产的安装目录：

```
$ mkdir <installation_directory>
```



重要

您必须创建目录。一些安装信息，如 bootstrap X.509 证书，有较短的过期间隔，因此不要重复使用安装目录。如果要重复使用另一个集群安装中的个别文件，可以将其复制到您的目录中。但是，一些安装数据的文件名可能会在发行版本之间有所改变。从 OpenShift Container Platform 老版本中复制安装文件时要格外小心。

2. 自定义以下 **install-config.yaml** 文件模板，并将它保存到 **<installation_directory>** 中。

注意

此配置文件必须命名为 **install-config.yaml**。

3. 备份 **install-config.yaml** 文件，以便用于安装多个集群。

重要

install-config.yaml 文件会在安装过程的下一步骤中消耗掉。现在必须备份它。

1.2.7.1. VMware vSphere install-config.yaml 文件示例

您可以自定义 **install-config.yaml** 文件，以指定有关 OpenShift Container Platform 集群平台的更多信息，或修改所需参数的值。

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
```

```
  hyperthreading: Enabled 5 6
  name: master
  replicas: 3 7
metadata:
  name: test 8
platform:
  vsphere:
    vcenter: your.vcenter.server 9
    username: username 10
    password: password 11
    datacenter: datacenter 12
    defaultDatastore: datastore 13
pullSecret: '{"auths": ...}' 14
sshKey: 'ssh-ed25519 AAAA...' 15
```

1 集群的基域。所有 DNS 记录都必须是这个基域的子域，并包含集群名称。

2 5 **controlPlane** 部分是一个单映射，但 compute 部分是一系列映射。为满足不同数据结构的要求，**compute** 部分的第一行必须以连字符 **-** 开头，**controlPlane** 部分的第一行则不可以连字符开头。虽然这两个部分目前都定义单个机器池，但未来的 OpenShift Container Platform 版本可能会支持在安装过程中定义多个计算池。只使用一个 control plane 池。

3 6 是否要启用或禁用并发多线程或**超线程**。默认情况下，启用并发多线程以提高机器内核的性能。您可以通过将参数值设为 **Disabled** 来禁用。如果您在某些集群机器上禁用并发多线程，则必须在所有集群机器上禁用。

重要

如果禁用并发多线程，请确保在容量规划时考虑到机器性能可能会显著降低的问题。如果您禁用并发多线程，则计算机必须至少使用 8 个 CPU 和 32GB RAM。

4 **replicas** 参数的值必须设置为 **0**。此参数控制集群为您创建和管理的 worker 数量，使用用户置备的基础架构时集群不会执行这些功能。在完成 OpenShift Container Platform 安装前，您必须手动为集群部署 worker 机器。

7 您添加到集群的 control plane 机器数量。由于集群将这个值用作集群中 etcd 端点的数量，因此该值必须与您部署的 control plane 机器数量匹配。

8 您在 DNS 记录中指定的集群名称。

9 vCenter 服务器的完全限定主机名或 IP 地址。

10 用于访问服务器的用户名。此用户必须至少在 vSphere 中有静态或动态持久性卷部署所需的角色和权限。

11 与 vSphere 用户关联的密码。

12 vSphere 数据中心。

13 要使用的默认 vSphere 数据存储。

14 从 Red Hat OpenShift Cluster Manager 站点的 Pull Secret 页面中获取的 pull secret。通过此 pull secret，您可以进行所含授权机构提供的服务的身份验证，这些服务包括为 OpenShift Container Platform 组件提供容器镜像的 Quay.io。

15 Red Hat Enterprise Linux CoreOS (RHCOS) 中 **core** 用户的默认 SSH 密钥的公钥部分。

注意

对于您要在其上执行安装调试或灾难恢复的生产环境 OpenShift Container Platform 集群，请指定 **ssh-agent** 进程使用的 SSH 密钥。

1.2.7.2. 网络配置参数

您可以在 **install-config.yaml** 配置文件中修改集群网络配置参数。下表描述了这些参数。

注意

安装之后，您无法修改 **install-config.yaml** 文件中的这些参数。

表 1.7. 所需的网络参数

参数	描述	值
networking.networkType	要部署的网络插件。**OpenShiftSDN** 插件是 OpenShift Container Platform 4.2 中唯一支持的插件。	默认值为 **OpenShiftSDN**。
networking.clusterNetwork.cidr	从中分配 Pod IP 地址的 IP 地址块。**OpenShiftSDN** 网络插件支持多个集群网络。多个集群网络的地址块不得互相重叠。请选择足够大的地址池，以适配预期的工作负载。	CIDR 格式的 IP 地址分配。默认值为 **10.128.0.0/14**。
networking.clusterNetwork.hostPrefix	分配给每个单独节点的子网前缀长度。例如，如果 **hostPrefix** 设为 **23**，则每个节点从所给的 **cidr** 中分配一个 **/23** 子网，这样就能有 510 (2^(32 - 23) - 2) 个 Pod IP 地址。	子网前缀。默认值为 **23**。
networking.serviceNetwork	服务的 IP 地址块。**OpenShiftSDN** 只允许一个 **serviceNetwork** 块。该地址块不得与任何其他网络块重叠。	CIDR 格式的 IP 地址分配。默认值为 **172.30.0.0/16**。
networking.machineCIDR	OpenShift Container Platform 安装程序在安装集群时使用的 IP 地址块。该地址块不得与任何其他网络块重叠。	CIDR 格式的 IP 地址分配。默认值为 **10.0.0.0/16**。

1.2.8. 修改高级网络配置参数

您只能在安装集群前修改高级网络配置参数。通过自定义高级配置，您可以指定 MTU 或 VXLAN 端口，允许自定义 kube-proxy 设置，以及为 **openshiftSDNConfig** 参数指定不同的 **mode**，从而将集群整合到现有的网络环境中。

重要

不支持直接修改 OpenShift Container Platform 清单文件。

先决条件

- 创建 **install-config.yaml** 文件并完成对其所做的任何修改。
- 为集群生成 Ignition 配置文件。

流程

1. 使用以下命令来创建清单：

```
$ ./openshift-install create manifests --dir=<installation_directory> 1
```

1 对于 **<installation_directory>**，请指定含有集群的 **install-config.yaml** 文件的目录的名称。

2. 修改 **<installation_directory>/manifests/cluster-scheduler-02-config.yml** Kubernetes 清单文件，以防止在 control plane 机器上调度 Pod：
   a. 打开 **manifests/cluster-scheduler-02-config.yml** 文件。
   b. 找到 **mastersSchedulable** 参数，并将其值设为 **False**。
   c. 保存并退出文件。

注意

目前，由于 Kubernetes 限制，入口负载均衡器将无法访问在 control plane 机器上运行的路由器 Pod。

3. 在 **<installation_directory>/manifests/** 目录下，创建一个名为 **cluster-network-03-config.yml** 的文件：

```
$ touch <installation_directory>/manifests/cluster-network-03-config.yml 1
```

1 对于 **<installation_directory>**，请指定包含集群的 **manifests/** 目录的目录名称。

创建该文件后，**manifests/** 目录中会包含多个网络配置文件，如下所示：

```
$ ls <installation_directory>/manifests/cluster-network-*
cluster-network-01-crd.yml
cluster-network-02-config.yml
cluster-network-03-config.yml
```

4. 在编辑器中打开 **cluster-network-03-config.yml** 文件，然后输入描述您想要的 Operator 配置的 CR：

```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec: 1
  clusterNetwork:
```

```
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork:
  - 172.30.0.0/16
  defaultNetwork:
    type: OpenShiftSDN
    openshiftSDNConfig:
      mode: NetworkPolicy
      mtu: 1450
      vxlanPort: 4789
```

1 **spec** 参数的参数仅作示例。在 CR 中为 Cluster Network Operator 指定配置。

CNO 为 CR 中的参数提供默认值，因此您必须只指定要更改的参数。

5. 保存 **cluster-network-03-config.yml** 文件，再退出文本编辑器。

6. 可选：备份 **manifests/cluster-network-03-config.yml** 文件。创建集群时，安装程序会删除 **manifests/** 目录。

1.2.9. Cluster Network Operator 自定义资源 (CR)

Network.operator.openshift.io 自定义资源 (CR) 中的集群网络配置存储 Cluster Network Operator (CNO) 的配置设置。Operator 管理集群网络。

您可以通过在 CNO CR 中设置 **defaultNetwork** 参数的参数，为 OpenShift Container Platform 集群指定集群网络配置。以下 CR 显示了 CNO 的默认配置，并且说明了您可以配置的参数和有效的参数值：

Cluster Network Operator CR

```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec:
  clusterNetwork: 1
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork: 2
  - 172.30.0.0/16
  defaultNetwork: 3
    ...
  kubeProxyConfig: 4
    iptablesSyncPeriod: 30s 5
    proxyArguments:
      iptables-min-sync-period: 6
      - 30s
```

1 2 在 **install-config.yaml** 文件中指定。

3 为集群网络配置软件定义型网络 (SDN)。

4 此对象的参数指定 **kube-proxy** 配置。如果没有指定参数值，Network Operator 会使用显示的默认参数值。

5 **iptables** 规则的刷新周期。默认值为 **30s**。有效的后缀包括 **s**、**m**和 **h**，具体参见 Go 时间包文档。

6 刷新 **iptables** 规则前的最短时长。此参数确保刷新的频率不会过于频繁。有效的后缀包括 **s**、**m**和 **h**，具体参见 Go 时间包。

1.2.9.1. OpenShift SDN 的配置参数

以下 YAML 对象描述了 OpenShift SDN 的配置参数：

```
defaultNetwork:
  type: OpenShiftSDN 1
  openshiftSDNConfig: 2
    mode: NetworkPolicy 3
    mtu: 1450 4
    vxlanPort: 4789 5
```

1 在 **install-config.yaml** 文件中指定。

2 只有您要覆盖部分 OpenShift SDN 配置时才需要指定。

3 配置 **OpenShiftSDN** 的网络隔离模式。允许的值有 **Multitenant**、**Subnet** 或 **NetworkPolicy**。默认值为 **NetworkPolicy**。

4 用于 VXLAN 覆盖网络的 MTU。这个值通常是自动配置的；但是，如果集群中的节点没有全部使用相同的 MTU，那么您必须将此值明确设置为比最小节点 MTU 的值小 50。

5 用于所有 VXLAN 数据包的端口。默认值为 **4789**。如果您在虚拟环境中运行，并且现有节点是另一个 VXLAN 网络的一部分，那么可能需要更改此值。例如，当在 VMware NSX-T 上运行 OpenShift SDN 覆盖时，您必须为 VXLAN 选择一个备用端口，因为两个 SDN 都使用相同的默认 VXLAN 端口号。

在 Amazon Web Services (AWS) 上，您可以在端口 **9000** 和端口 **9999** 之间为 VXLAN 选择一个备用端口。

1.2.9.2. Cluster Network Operator 示例 CR

下例中显示了 CNO 的完整 CR：

Cluster Network Operator 示例 CR

```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork:
  - 172.30.0.0/16
  defaultNetwork:
    type: OpenShiftSDN
    openshiftSDNConfig:
```

```
    mode: NetworkPolicy
    mtu: 1450
    vxlanPort: 4789
kubeProxyConfig:
  iptablesSyncPeriod: 30s
  proxyArguments:
    iptables-min-sync-period:
    - 30s
```

1.2.10. 创建 Ignition 配置文件

由于需要手工启动集群机器，因此您必须生成 Ignition 配置文件，集群需要它来创建其机器。

重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须完成集群安装，并使集群以非降级状态运行 24 小时，以确保完成第一次证书轮转。

先决条件

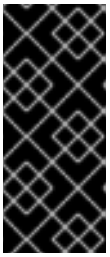
- 获取 OpenShift Container Platform 安装程序以及集群的 pull secret。

流程

1. 获取 Ignition 配置文件：

```
$ ./openshift-install create ignition-configs --dir=<installation_directory> 1
```

1 对于 **<installation_directory>**，请指定用于保存安装程序所创建的文件的目录名称。



重要

如果您创建了 **install-config.yaml** 文件，请指定包含该文件的目录。否则，指定一个空目录。一些安装信息，如 bootstrap X.509 证书，有较短的过期间隔，因此不要重复使用安装目录。如果要重复使用另一个集群安装中的个别文件，可以将其复制到您的目录中。但是，一些安装数据的文件名可能会在发行版本之间有所改变。从 OpenShift Container Platform 老版本中复制安装文件时要格外小心。

该目录中将生成以下文件：

```
.
├── auth
│   ├── kubeadmin-password
│   └── kubeconfig
├── bootstrap.ign
├── master.ign
├── metadata.json
└── worker.ign
```

1.2.11. 在 vSphere 中创建 Red Hat Enterprise Linux CoreOS (RHCOS) 机器

在 VMware vSphere 上安装包含用户置备基础架构的集群前，您必须在 vSphere 主机上创建 RHCOS 机器供其使用。

先决条件

- 获取集群的 Ignition 配置文件。
- 具有 HTTP 服务器的访问权限，以便您可从计算机进行访问，并且您创建的机器也可访问此服务器。
- 创建 vSphere 集群。

流程

1. 将名为 **<installation_directory>/bootstrap.ign** 的 bootstrap Ignition 配置文件上传到 HTTP 服务器，该配置文件是由安装程序创建的。记下此文件的 URL。
   您必须托管 bootstrap Ignition 配置文件，因为它太大而无法放在 vApp 属性中。
2. 将 bootstrap 节点的以下辅助 Ignition 配置文件保存到计算机中，存为 **<installation_directory>/append-bootstrap.ign**。

```
{
  "ignition": {
    "config": {
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

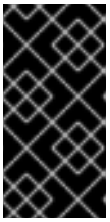
1 指定您托管的 bootstrap Ignition 配置文件的 URL。

   为 bootstrap 机器创建虚拟机 (VM) 时，您要使用此 Ignition 配置文件。

3. 将 master、worker 和辅助 Bootstrap Ignition 配置文件转换为 Base64 编码。
   例如，如果您使用 Linux 操作系统，可以使用 **base64** 命令来编码这些文件。

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. 从红帽客户门户上的产品下载页面或 RHCOS 镜像页面，获取 RHCOS OVA 镜像。



重要

RHCOS 镜像可能不会随着 OpenShift Container Platform 的每一发行版本都有改变。您必须下载一个最高版本的镜像，其版本号应小于或等于您安装的 OpenShift Container Platform 版本。如果可用，请使用与 OpenShift Container Platform 版本匹配的镜像版本。

文件名包含 OpenShift Container Platform 版本号，格式为 **rhcos-<version>-<architecture>-vmware.ova**。

5. 在 vSphere 客户端中，在数据中心中创建一个文件夹来存储您的虚拟机。
   a. 点击 **VMs and Templates** 视图。
   b. 右键点击您的数据中心名称。
   c. 点击 **New Folder → New VM and Template Folder**。
   d. 在显示的窗口中输入文件夹名称。文件夹名称必须与您在 **install-config.yaml** 文件中指定的集群名称匹配。
6. 在 vSphere 客户端中，为 OVA 镜像创建一个模板。

注意

在以下步骤中，您要将同一模板用于所有集群机器，并针对您在置备虚拟机时指定的机器类型提供 Ignition 配置文件的位置。

   a. 在 **Hosts and Clusters** 选项卡中，右键点击集群名称并点击 **Deploy OVF Template**。
   b. 在 **Select an OVF** 选项卡中，指定您下载的 RHCOS OVA 文件的名称。
   c. 在 **Select a name and folder**选项卡中，设置 **Virtual machine name**，如 RHCOS，再点击 vSphere 集群的名称，然后选择您在上一步中创建的文件夹。
   d. 在 **Select a compute resource** 选项卡中，点击您的 vSphere 集群名称。
   e. 在 **Select storage** 选项卡中，配置虚拟机的存储选项。
      - 选择 **Thin Provision**。
      - 选择您在 **install-config.yaml** 文件中指定的数据存储。
   f. 在 **Select network** 选项卡中，指定您为集群配置的网络（如果可用）。
   g. 如果您计划将同一模板用于所有集群机器类型，请不要在 **Customize template** 选项卡中指定值。
7. 部署模板后，为集群中的机器部署虚拟机。
   a. 右键点击模板的名称，再点击 **Clone → Clone to Virtual Machine**。
   b. 在 **Select a name and folder**选项卡中，指定虚拟机的名称。名称中可以包括机器类型，如 **control-plane-0** 或 **compute-1**。
   c. 在 **Select a name and folder**选项卡中，选择您为集群创建的文件夹名称。

d. 在 **Select a compute resource** 选项卡中，选择数据中心中的主机名称。

e. 可选：在 **Select storage** 选项卡中，自定义存储选项。

f. 在 **Select clone options** 中，选择 **Customize this virtual machine's hardware**。

g. 在 **Customize hardware** 选项卡中，点击 **VM Options → Advanced**。

- 可选：在出现集群性能问题时，从 **Latency Sensitivity** 列表中选择 **High**。
- 点击 **Edit Configuration**，然后在 **Configuration Parameters** 窗口中点击 **Add Configuration Params**。定义以下参数名称和值：
  - **guestinfo.ignition.config.data**：粘贴此机器类型的 base64 编码 Ignition 配置文件的内容。
  - **guestinfo.ignition.config.data.encoding**：指定 **base64**。
  - **disk.EnableUUID**：指定 **TRUE**。
- 另外，也在打开虚拟机电源前通过 vApp 属性添加：
  - 导航到 vCenter 服务器清单中的某一虚拟机。
  - 在 **Configure** 选项卡中，展开 **Settings** 并选择 **vApp options**。
  - 向下滚动，并在 **Properties** 下应用上述配置。

h. 在 **Customize hardware** 选项卡的 **Virtual Hardware** 面板中，根据需要修改指定的值。确保 RAM、CPU 和磁盘存储的数量满足机器类型的最低要求。

i. 完成配置并打开虚拟机电源。

8. 对于每台机器，按照前面的步骤为集群创建其余的机器。

重要

此刻您必须创建 bootstrap 和 control plane 机器。由于计算机器上已默认部署了一些 Pod，因此在安装集群前，还要创建至少两台计算机器。

1.2.12. 安装 CLI

为了可以使用命令行界面与 OpenShift Container Platform 进行交互，您需要安装 CLI。

重要

如果安装了旧版本的 **oc**，则无法使用 OpenShift Container Platform 4.2 中的所有命令。下载并安装新版本的 **oc**。

流程

1. 在 Red Hat OpenShift Cluster Manager 站点的 Infrastructure Provider 页面中导航至您的安装类型页面，并点击 **Download Command-line Tools**。
2. 点您的操作系统和系统架构的文件夹，然后点压缩文件。

注意

您可在 Linux、Windows 或 macOS 上安装 **oc**。

3. 将文件保存到文件系统。
4. 展开压缩文件。
5. 把它放到 **PATH** 中的一个目录下。

安装 CLI 后，就可以使用**oc**命令：

```
$ oc <command>
```

1.2.13. 创建集群

要创建 OpenShift Container Platform 集群，请等待您通过安装程序生成的 Ignition 配置文件所置备的机器上完成 bootstrap 过程。

先决条件

- 为集群创建所需的基础架构。
- 已获得安装程序并为集群生成了 Ignition 配置文件。
- 已使用 Ignition 配置文件为集群创建 RHCOS 机器。
- 您的机器能够直接访问互联网。

流程

1. 监控 bootstrap 过程：

```
$ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
    --log-level=info 2
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.14.6+c4799753c up
INFO Waiting up to 30m0s for the bootstrap-complete event...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2 要查看不同的安装详情，请指定 **warn**、**debug** 或 **error**，而不要指定 **info**。

Kubernetes API 服务器提示已在 control plane 机器上完成 bootstrap 时，命令运行成功。

2. bootstrap 过程完成后，请从负载均衡器中删除 bootstrap 机器。

重要

此时您必须从负载均衡器中删除 bootstrap 机器。您还可以删除或重新格式化机器本身。

1.2.14. 登录集群

您可以通过导出集群 **kubeconfig** 文件，以默认系统用户身份登录集群。**kubeconfig** 文件包含关于集群的信息，供 CLI 用于将客户端连接到正确集群和 API 服务器。该文件特只适用于一个特定的集群，在 OpenShift Container Platform 安装过程中创建。

先决条件

- 部署 OpenShift Container Platform 集群。
- 安装 **oc** CLI。

流程

1. 导出 **kubeadmin** 凭证：

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2. 使用导出的配置，验证能否成功运行 **oc** 命令：

```
$ oc whoami
system:admin
```

1.2.15. 批准机器的 CSR

将机器添加到集群时，系统会为添加的每台机器生成两个待处理证书签名请求 (CSR)。您必须确认这些 CSR 已获得批准，或根据需要自行批准。

先决条件

- 您已将机器添加到集群中。

流程

1. 确认集群可以识别这些机器：

```
$ oc get nodes

NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.14.6+c4799753c
master-1  Ready     master  63m  v1.14.6+c4799753c
master-2  Ready     master  64m  v1.14.6+c4799753c
worker-0  NotReady  worker  76s  v1.14.6+c4799753c
worker-1  NotReady  worker  70s  v1.14.6+c4799753c
```

输出将列出您创建的所有机器。

2. 检查待处理证书签名请求 (CSR)，并确保您添加到集群中的每一机器都有状态为 **Pending** 或 **Approved** 的客户端和服务器请求：

```
$ oc get csr

NAME        AGE     REQUESTOR                                      CONDITION
```

```
csr-8b2br   15m    system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Pending (1)
csr-8vnps   15m    system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Pending
csr-bfd72   5m26s  system:node:ip-10-0-50-126.us-east-2.compute.internal                        Pending (2)
csr-c57lv   5m26s  system:node:ip-10-0-95-157.us-east-2.compute.internal                        Pending
...
```

1. 客户端请求 CSR。
2. 服务器请求 CSR。

在本例中，两台机器加入了集群。您可能会在列表中看到更多已批准的 CSR。

3. 如果 CSR 没有获得批准，请在所添加机器的所有待处理 CSR 都处于 **Pending** 状态后，为您的集群机器批准这些 CSR：

注意

由于 CSR 会自动轮转，因此请在将机器添加到集群后一小时内批准您的 CSR。如果没有在一小时内批准，证书将会轮转，每个节点将会存在多个证书。您必须批准所有这些证书。批准初始 CSR 后，集群的 **kube-controller-manager** 会自动批准后续的节点客户端 CSR。您必须实施一个方法来自动批准 kubelet 提供的证书请求。

- 若要单独批准，请对每个有效的 CSR 运行以下命令：

```
$ oc adm certificate approve <csr_name> (1)
```

1. **<csr_name>** 是当前 CSR 列表中 CSR 的名称。

- 要批准所有待处理的 CSR，请运行以下命令：

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}{{end}}{{end}}' | xargs oc adm certificate approve
```

1.2.16. 初始 Operator 配置

在 control plane 初始化后，您必须立即配置一些 Operator 以便它们都可用。

先决条件

- 您的 control plane 已初始化。

流程

1. 观察集群组件上线：

```
$ watch -n5 oc get clusteroperators
```

```
NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                             4.2.0     True        False         False      69s
cloud-credential                           4.2.0     True        False         False      12m
cluster-autoscaler                         4.2.0     True        False         False      11m
console                                    4.2.0     True        False         False      46s
dns                                        4.2.0     True        False         False      11m
image-registry                             4.2.0     False       True          False      5m26s
ingress                                    4.2.0     True        False         False      5m36s
kube-apiserver                             4.2.0     True        False         False      8m53s
kube-controller-manager                    4.2.0     True        False         False      7m24s
kube-scheduler                             4.2.0     True        False         False      12m
machine-api                                4.2.0     True        False         False      12m
machine-config                             4.2.0     True        False         False      7m36s
marketplace                                4.2.0     True        False         False      7m54m
monitoring                                 4.2.0     True        False         False      7h54s
network                                    4.2.0     True        False         False      5m9s
node-tuning                                4.2.0     True        False         False      11m
openshift-apiserver                        4.2.0     True        False         False      11m
openshift-controller-manager               4.2.0     True        False         False      5m943s
openshift-samples                          4.2.0     True        False         False      3m55s
operator-lifecycle-manager                 4.2.0     True        False         False      11m
operator-lifecycle-manager-catalog         4.2.0     True        False         False      11m
service-ca                                 4.2.0     True        False         False      11m
service-catalog-apiserver                  4.2.0     True        False         False      5m26s
service-catalog-controller-manager         4.2.0     True        False         False      5m25s
storage                                    4.2.0     True        False         False      5m30s
```

2. 配置不可用的 Operator。

1.2.16.1. 镜像 registry 存储配置

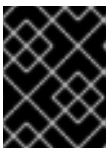
如果 **image-registry** Operator 不可用，您必须为其配置存储。提供了配置 PersistentVolume 的说明，这是生产集群所需要的；也提供了将空目录配置为存储位置的说明，这仅适用于非生产集群。

1.2.16.1.1. 为 VMware vSphere 配置 registry 存储

作为集群管理员，在安装后需要配置 registry 来使用存储。

先决条件

- 具有 Cluster Administrator 权限
- VMware vSphere上有一个集群。
- 具有**ReadWriteMany**访问模式的持久性存储卷（PV），例如**NFS** 。



重要

vSphere卷不支持**ReadWriteMany**访问模式。您必须使用其他存储后端（例如**NFS** ）来配置 registry 注存储。

- 必须有" 100Gi"容量。

流程

1. 为了配置 registry 使用存储，需要修改 **configs.imageregistry/cluster** 资源中的 **spec.storage.pvc** 。

2. 验证您没有 registry Pod：

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

注意

如果存储类型为 **emptyDIR**，则副本数不能超过 **1**。如果存储类型为**NFS** ，并且希望通过设置 **replica>1**来扩展 registry Pod，则必须启用**no_wdelay**挂载选项。例如：

```
# cat /etc/exports
/mnt/data *(rw,sync,no_wdelay,no_root_squash,insecure,fsid=0)
sh-4.3# exportfs -rv
exporting *:/mnt/data
```

3. 检查 registry 配置：

```
$ oc edit configs.imageregistry.operator.openshift.io

storage:
  pvc:
    claim:
```

将 **claim** 字段留空以允许自动创建一个 **image-registry-storage** PVC。

4. 可选：在 PV 中添加新存储类：

   a. 创建 PV：

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    ReadWriteMany
    capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

b. 创建 PVC：

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

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

最后，添加 PVC 的名称：

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc (1)
```

(1) 通过创建自定义 PVC，您可以将 **claim** 字段留空以用于默认自动创建 **image-registry-storage** PVC。

5. 检查 **clusteroperator** 的状态：

```
$ oc get clusteroperator image-registry
```

1.2.16.1.2. 在非生产集群中配置镜像 registry 存储

您必须为 Image Registry Operator 配置存储。对于非生产集群，您可以将镜像 registry 设置为空目录。如果您这样做，重启 registry 后会丢失所有镜像。

流程

- 将镜像 registry 存储设置为空目录：

```
$ oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

警告

仅可为非生产集群配置这个选项。

如果在 Image Registry Operator 初始化其组件前运行此命令，**oc patch** 命令会失败并显示以下错误：

```
Error from server (NotFound): configs.imageregistry.operator.openshift.io "cluster" not found
```

等待几分钟，然后再次运行该命令。

1.2.17. 在用户置备的基础架构上完成安装

完成 Operator 配置后，可以在您提供的基础架构上完成集群安装。

先决条件

- 您的 control plane 已初始化。
- 已完成初始 Operator 配置。

流程

1. 确认所有集群组件都已上线：

```
$ watch -n5 oc get clusteroperators

NAME                                 VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                       4.2.0     True        False         False      10m
cloud-credential                     4.2.0     True        False         False      22m
cluster-autoscaler                   4.2.0     True        False         False      21m
console                              4.2.0     True        False         False      10m
dns                                  4.2.0     True        False         False      21m
image-registry                       4.2.0     True        False         False      16m
ingress                              4.2.0     True        False         False      16m
kube-apiserver                       4.2.0     True        False         False      19m
kube-controller-manager              4.2.0     True        False         False      18m
kube-scheduler                       4.2.0     True        False         False      22m
machine-api                          4.2.0     True        False         False      22m
machine-config                       4.2.0     True        False         False      18m
marketplace                          4.2.0     True        False         False      18m
monitoring                           4.2.0     True        False         False      18m
network                              4.2.0     True        False         False      16m
node-tuning                          4.2.0     True        False         False      21m
openshift-apiserver                  4.2.0     True        False         False      21m
openshift-controller-manager         4.2.0     True        False         False      17m
openshift-samples                    4.2.0     True        False         False      14m
operator-lifecycle-manager           4.2.0     True        False         False      21m
operator-lifecycle-manager-catalog   4.2.0     True        False         False      21m
```

```
service-ca                                 4.2.0     True        False         False      21m
service-catalog-apiserver                  4.2.0     True        False         False      16m
service-catalog-controller-manager         4.2.0     True        False         False      16m
storage                                    4.2.0     True        False         False      16m
```

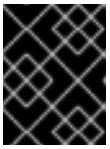
当所有集群 Operator 状态都是 **AVAILABLE** 时，您可以完成安装。

2. 监控集群完成：

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
INFO Waiting up to 30m0s for the cluster to initialize...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

Cluster Version Operator 完成从 Kubernetes API 服务器部署 OpenShift Container Platform 集群时，命令运行成功。



重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须以非降级状态持续运行集群 24 小时，以确保完成第一次证书轮转。

3. 确认 Kubernetes API 服务器正在与 Pod 通信。

   a. 要查看所有 Pod 的列表，请使用以下命令：

```
$ oc get pods --all-namespaces

NAMESPACE                       NAME                                         READY   STATUS
RESTARTS   AGE
openshift-apiserver-operator    openshift-apiserver-operator-85cb746d55-zqhs8   1/1
Running    1          9m
openshift-apiserver             apiserver-67b9g                               1/1     Running     0
3m
openshift-apiserver             apiserver-ljcmx                               1/1     Running     0
1m
openshift-apiserver             apiserver-z25h4                               1/1     Running     0
2m
openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8       1/1
Running     0          5m
...
```

   b. 使用以下命令，查看上一命令的输出中所列 Pod 的日志：

```
$ oc logs <pod_name> -n <namespace> 1
```

1 指定 Pod 名称和命名空间，如上一命令的输出中所示。

如果 Pod 日志显示，Kubernetes API 服务器可以与集群机器通信。

后续步骤

- 自定义集群。

- 若有需要，您可以选择不使用远程健康报告。

1.3. 在受限网络中的 VSPHERE 上安装集群

在 OpenShift Container Platform 版本 4.2 中，可以在受限网络中置备的 VMware vSphere 基础架构上安装集群。

先决条件

- 在堡垒主机上创建镜像 registry，并获取您的 OpenShift Container Platform 版本的 **imageContentSources** 数据。

重要

由于安装介质位于堡垒主机上，因此请使用该计算机完成所有安装步骤。

- 为集群置备持久性存储。若要部署私有镜像 registry，您的存储必须提供 ReadWriteMany 访问模式。
- 查看有关 OpenShift Container Platform 安装和更新流程的详细信息。
- 如果使用防火墙并计划使用遥测（telemetry），您必须将其配置为允许集群需要访问的站点。

注意

如果您要配置代理，请务必也要查看此站点列表。

1.3.1. 关于在受限网络中安装

在 OpenShift Container Platform 4.2 中，可以执行不需要互联网活跃连接也能获取软件组件的安装。受限网络安装只能在您置备的基础架构上完成，不能在安装程序置备的基础架构上完成，因此您的平台选择会受到限制。

如果选择在云平台中执行受限网络安装，仍然需要访问其云 API。有些云功能，比如 Amazon Web Service 的 IAM 服务，需要访问互联网，因此您可能仍需要连入互联网。根据您的网络，在裸机硬件或 VMware vSphere 上安装时可能需要较少的互联网访问。

要完成受限网络安装，您必须创建一个 registry，镜像 OpenShift Container Platform registry 的内容并包含其安装介质。您可以在堡垒主机上创建此镜像，该主机可同时访问互联网和您的封闭网络，也可以使用满足您的限制条件的其他方法。

重要

受限网络安装始终使用用户置备的基础架构。由于用户置备安装配置的复杂性，在尝试受限网络安装前，请考虑完成标准用户置备基础架构安装。通过完成此测试安装，您可以更轻松地隔离和排查您在受限网络中安装时可能出现的问题。

1.3.1.1. 其他限制

受限网络中的集群还有以下额外限制：

- ClusterVersion 状态包含 **Unable to retrieve available updates** 错误。

- 默认情况下，您无法使用 Developer Catalog 的内容，因为您无法访问所需的 ImageStreamTag。

1.3.2. OpenShift Container Platform 对互联网和 Telemetry 的访问

在 OpenShift Container Platform 4.2 中，您需要访问互联网来获得用来安装集群的镜像。默认运行的 Telemetry 服务提供有关集群健康状况和成功更新的指标，这也需要访问互联网。如果您的集群连接到互联网，Telemetry 会自动运行，而且集群会注册到 Red Hat OpenShift Cluster Manager （OCM）。

确认 Red Hat OpenShift Cluster Manager 清单正确后，可以由 Telemetry 自动维护，也可以使用 OCM 手动维护，使用订阅监控 来跟踪帐户或多集群级别的 OpenShift Container Platform 订阅。

您必须具有以下互联网访问权限：

- 访问 Red Hat OpenShift Cluster Manager 页面，以下载安装程序并执行订阅管理。如果集群可以访问互联网，并且没有禁用 Telemetry，该服务会自动授权您的集群。
- 访问 Quay.io，以获取安装集群所需的软件包。
- 获取执行集群更新所需的软件包。

重要

如果您的集群无法直接访问互联网，则可以在置备的某些类基础架构上执行受限网络安装。在此过程中，您要下载所需的内容，并使用它在镜像 registry（mirror registry） 中填充安装集群并生成安装程序所需的软件包。对于某些安装类型，集群要安装到的环境不需要访问互联网。在更新集群之前，要更新 registry 镜像系统中的内容。

1.3.3. VMware vSphere 基础架构要求

您必须在 VMware vSphere 版本 6.5 或 6.7U2 或更高版本的实例上安装 OpenShift Container Platform 集群。

VMware 建议您将 Sphere Version 6.7 U2 或更高版本与 OpenShift Container Platform 集群搭配使用。vSphere 6.7U2 包括：

- 支持 VMware NSX-T
- 支持 vSAN 、VMFS 和 NFS（使用树状 VCP）

尽管支持 vSphere 6.5（硬件版本 13），但 OpenShift Container Platform 集群会受以下限制：

- 不支持 NSX-T SDN。
- 您必须使用 OpenShift Container Platform 支持的其他 SDN 或存储提供程序。

如果您使用 vSphere 版本 6.5 实例，请考虑升级到 6.7U2 后再安装 OpenShift Container Platform。

重要

您必须确保在安装 OpenShift Container Platform 前同步 ESXi 主机上的时间。请参阅 VMware 文档中的编辑主机时间配置。

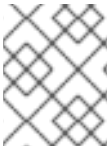
1.3.4. 具有用户置备基础架构的集群的机器要求

对于含有用户置备的基础架构的集群，您必须部署所有所需的机器。

1.3.4.1. 所需的机器

最小的 OpenShift Container Platform 集群需要下列主机：

- 一个临时 bootstrap 机器
- 三台 control plane 或 master 机器
- 至少两台计算机器，也称为 worker 机器



注意

集群要求 bootstrap 机器在三台 control plane 机器上部署 OpenShift Container Platform 集群。您可在安装集群后删除 bootstrap 机器。

重要

要保持集群的高可用性，请将独立的物理主机用于这些集群机器。

bootstrap、control plane 以及计算（compute）机器必须使用 Red Hat Enterprise Linux CoreOS (RHCOS) 作为操作系统。

请注意，RHCOS 基于 Red Hat Enterprise Linux 8，并继承其所有硬件认证和要求。请查看Red Hat Enterprise Linux 技术功能及限制。

1.3.4.2. 网络连接要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置文件。在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

1.3.4.3. 最低资源要求

每台集群机器都必须满足以下最低要求：

机器	操作系统	vCPU	虚拟内存	存储
bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS 或 RHEL 7.6	2	8 GB	120 GB

1.3.4.4. 证书签名请求管理

在使用您置备的基础架构时，集群只能有限地访问自动机器管理，因此您必须提供一种在安装后批准集群证书签名请求 (CSR) 的机制。**kube-controller-manager** 只能批准 kubelet 客户端 CSR。 **machine-approver** 无法保证使用 kubelet 凭证请求的提供证书的有效性，因为它不能确认是正确的机器发出了该请求。您必须决定并实施一种方法，以验证 kubelet 提供证书请求的有效性并进行批准。

1.3.5. 创建用户置备的基础架构

在部署采用用户置备的基础架构的 OpenShift Container Platform 集群前，您必须创建底层基础架构。

先决条件

- 在为集群创建支持基础架构之前，请参阅OpenShift Container Platform 4.x Tested Integrations页。

流程

1. 配置 DHCP。
2. 提供所需的负载均衡器。
3. 配置机器的端口。
4. 配置 DNS。
5. 确保网络可以正常工作。

1.3.5.1. 用户置备的基础架构对网络的要求

所有 Red Hat Enterprise Linux CoreOS (RHCOS) 机器在启动过程中需要 **initramfs** 中的网络从 Machine Config Server 获取 Ignition 配置。

在初次启动过程中，机器需要 DHCP 服务器来建立网络连接，以下载其 Ignition 配置文件。

建议您使用 DHCP 服务器为集群进行长期机器管理。确保 DHCP 服务器已配置为向集群机器提供持久 IP 地址和主机名。

成功安装集群后，在每个 master 节点上运行的 Kubernetes API 服务器必须能够解析集群机器的节点名称。如果 API 服务器和 worker 节点位于不同的区域中，您可以配置默认 DNS 搜索区域，以便 API 服务器能够解析节点名称。另一种支持的方法是始终在节点对象和所有 DNS 请求中使用完全限定域名来指代主机。

您必须配置机器间的网络连接，以便集群组件进行通信。每台机器都必须能够解析集群中所有其他机器的主机名。

表 1.8. 所有机器到所有机器

协议	端口	描述
ICMP	N/A	网络可访问性测试
TCP	**9000**-**9999**	主机级别的服务，包括端口 **9100**-**9101** 上的节点导出器和端口 **9099** 上的 Cluster Version Operator。
	10250-**10259**	Kubernetes 保留的默认端口
	10256	openshift-sdn
UDP	**4789**	VXLAN 和 GENEVE

协议	端口	描述
	6081	VXLAN 和 GENEVE
	9000-9999	主机级别的服务，包括端口 **9100-9101** 上的节点导出器。
TCP/UDP	**30000-32767**	Kubernetes NodePort

表 1.9. 要通过控制平面的所有机器

协议	端口	描述
TCP	**2379-2380**	etcd 服务器、对等和指标端口
	6443	Kubernetes API

网络拓扑要求
您为集群置备的基础架构必须满足下列网络拓扑要求。

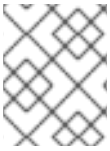
重要

OpenShift Container Platform 要求所有节点都能访问互联网，以便为平台容器提取镜像并向红帽提供遥测数据。

负载**均衡器**

在安装 OpenShift Container Platform 之前，您必须置备两个 L4 负载均衡器。API 需要一个负载均衡器，默认的 Ingress Controller 需要第二个负载均衡器来为应用程序提供入口网络。

端口	机器	内部	外部	描述
6443	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x	x	Kubernetes API 服务器
22623	Bootstrap 和 control plane.bootstrap 机器初始化集群 control plane 后，您要从负载均衡器中删除 bootstrap 机器。	x		机器配置服务器
443	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTPS 流量
80	默认运行入口路由器 Pod、计算或 worker 的机器。	x	x	HTTP 流量



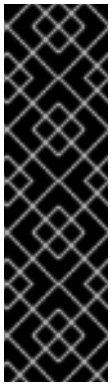
注意

OpenShift Container Platform 集群需要正确配置入口路由器。control plane 初始化后，您必须配置入口路由器。

1.3.5.2. 用户置备 DNS 要求

采用用户置备的基础架构的 OpenShift Container Platform 集群需要以下 DNS 记录。在每一记录中，**<cluster_name>** 是集群名称，**<base_domain>** 则是您在 **install-config.yaml** 文件中指定的集群基域。完整的 DNS 记录采用如下格式: **<component>.<cluster_name>.<base_domain>.**。

表 1.10. 所需的 DNS 记录

组件	记录	描述
Kubernetes API	**api.<cluster_name>.<base_domain>.**	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。
	api-int.<cluster_name>.<base_domain>.	此 DNS A/AAAA 或 CNAME 记录必须指向 control plane 机器的负载均衡器。此记录必须可从集群内的所有节点解析。  **重要** API 服务器必须能够根据在 Kubernetes 中记录的主机名解析 worker 节点。如果无法解析节点名称，代理的 API 调用会失败，而且您无法从 Pod 检索日志。
Routes	***.apps.<cluster_name>.<base_domain>.**	通配符 DNS A/AAAA 或 CNAME 记录，指向以运行入口路由器 Pod 的机器（默认为 worker 节点）为目标的负载均衡器。此记录必须能由集群外的客户端和集群内的所有节点解析。

组件	记录	描述
etcd	**etcd-<index>.<cluster_name>.<base_domain>.**	OpenShift Container Platform 要求每个 etcd 实例的 DNS A/AAAA 记录指向托管实例的 control plane 机器。etcd 实例通过 **<index>** 值来区分，值的范围为 **0** 到 **n-1**，其中 **n** 是集群中 control plane 机器的数量。DNS 记录必须解析到 control plane 机器的单播 IPv4 地址，并且这些记录必须可以从集群中的所有节点解析。
	_etcd-server-ssl._tcp.<cluster_name>.<base_domain>.	对于每台 control plane 机器，OpenShift Container Platform 还需要该机器上具有 etcd 服务器的 SRV DNS 记录，其优先级为 **0**，权重为 **10**，端口则为 **2380**。使用三台 control plane 机器的集群需要以下记录： `# _service._proto.name. TTL class SRV priority weight port target. _etcd-server-ssl._tcp. <cluster_name>. <base_domain>. 86400 IN SRV 0 10 2380 etcd-0.<cluster_name>. <base_domain> _etcd-server-ssl._tcp. <cluster_name>. <base_domain>. 86400 IN SRV 0 10 2380 etcd-1.<cluster_name>. <base_domain> _etcd-server-ssl._tcp. <cluster_name>. <base_domain>. 86400 IN SRV 0 10 2380 etcd-2.<cluster_name>. <base_domain>`

1.3.6. 生成 SSH 私钥并将其添加到代理中

如果要在集群上执行安装调试或灾难恢复，则必须为 **ssh-agent** 和安装程序提供 SSH 密钥。

注意

在生产环境中，您需要进行灾难恢复和调试。

您可以使用此密钥以 **core** 用户身份通过 SSH 连接到 master 节点。在部署集群时，此密钥会添加到 **core** 用户的 **~/.ssh/authorized_keys** 列表中。

注意

您必须使用一个本地密钥，而不要使用在特定平台上配置的密钥，如 AWS 密钥对。

流程

1. 如果还没有为计算机上免密码身份验证而配置的 SSH 密钥，请创建一个。例如，在使用 Linux 操作系统的计算机上运行以下命令：

```
$ ssh-keygen -t rsa -b 4096 -N '' \
  -f <path>/<file_name> 1
```

1 指定 SSH 密钥的路径和文件名，如 **~/.ssh/id_rsa**。

运行此命令会在指定的位置生成不需要密码的 SSH 密钥。

2. 作为后台任务启动 **ssh-agent** 进程：

```
$ eval "$(ssh-agent -s)"

Agent pid 31874
```

3. 将 SSH 私钥添加到 **ssh-agent**：

```
$ ssh-add <path>/<file_name> 1

Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

1 指定 SSH 私钥的路径和文件名，如 **~/.ssh/id_rsa**

后续步骤

- 在安装 OpenShift Container Platform 时，为安装程序提供 SSH 公钥。如果在您置备的基础架构上安装集群，您必须将此密钥提供给集群的机器。

1.3.7. 手动创建安装配置文件

对于使用用户置备的基础架构的 OpenShift Container Platform 安装，您必须手动生成安装配置文件。

先决条件

- 获取 OpenShift Container Platform 安装程序和集群的访问令牌。
- 获取命令输出中的 **imageContentSources** 部分来镜像存储库。
- 获取您的镜像 registry 的证书内容。

流程

1. 创建用来存储您所需的安装资产的安装目录：

```
$ mkdir <installation_directory>
```

重要

您必须创建目录。一些安装信息，如 bootstrap X.509 证书，有较短的过期间隔，因此不要重复使用安装目录。如果要重复使用另一个集群安装中的个别文件，可以将其复制到您的目录中。但是，一些安装数据的文件名可能会在发行版本之间有所改变。从 OpenShift Container Platform 老版本中复制安装文件时要格外小心。

2. 自定义以下 **install-config.yaml** 文件模板，并将它保存到 **<installation_directory>** 中。

注意

此配置文件必须命名为 **install-config.yaml**。

- 除非使用 RHCOS 默认信任的 registry，如 **docker.io**，否则必须在 **additionalTrustBundle** 部分中提供镜像存储库的证书内容。在大多数情况下，必须为您的镜像提供证书。
- 您必须包含命令输出中的 **imageContentSources** 部分，才能镜像存储库。

3. 备份 **install-config.yaml** 文件，以便用于安装多个集群。

重要

install-config.yaml 文件会在安装过程的下一步骤中消耗掉。现在必须备份它。

1.3.7.1. VMware vSphere install-config.yaml 文件示例

您可以自定义 **install-config.yaml** 文件，以指定有关 OpenShift Container Platform 集群平台的更多信息，或修改所需参数的值。

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
  hyperthreading: Enabled (5) (6)
  name: master
  replicas: 3 (7)
metadata:
  name: test (8)
platform:
  vsphere:
    vcenter: your.vcenter.server (9)
    username: username (10)
    password: password (11)
    datacenter: datacenter (12)
    defaultDatastore: datastore (13)
```

```
pullSecret: '{"auths":{"<bastion_host_name>:5000": {"auth": "<credentials>","email":
"you@example.com"}}}' 14
sshKey: 'ssh-ed25519 AAAA...' 15
additionalTrustBundle: | 16
  -----BEGIN CERTIFICATE-----
  ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
  -----END CERTIFICATE-----
imageContentSources: 17
- mirrors:
  - <bastion_host_name>:5000/<repo_name>/release
  source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - <bastion_host_name>:5000/<repo_name>/release
  source: registry.svc.ci.openshift.org/ocp/release
```

1 集群的基域。所有 DNS 记录都必须是这个基域的子域，并包含集群名称。

2 5 **controlPlane** 部分是一个单映射，但 compute 部分是一系列映射。为满足不同数据结构的要求，**compute** 部分的第一行必须以连字符 **-** 开头，**controlPlane** 部分的第一行则不可以连字符开头。虽然这两个部分目前都定义单个机器池，但未来的 OpenShift Container Platform 版本可能会支持在安装过程中定义多个计算池。只使用一个 control plane 池。

3 6 是否要启用或禁用并发多线程或**超线程**。默认情况下，启用并发多线程以提高机器内核的性能。您可以通过将参数值设为 **Disabled** 来禁用。如果您在某些集群机器上禁用并发多线程，则必须在所有集群机器上禁用。

重要

如果禁用并发多线程，请确保在容量规划时考虑到机器性能可能会显著降低的问题。如果您禁用并发多线程，则计算机必须至少使用 8 个 CPU 和 32GB RAM。

4 **replicas** 参数的值必须设置为 **0**。此参数控制集群为您创建和管理的 worker 数量，使用用户置备的基础架构时集群不会执行这些功能。在完成 OpenShift Container Platform 安装前，您必须手动为集群部署 worker 机器。

7 您添加到集群的 control plane 机器数量。由于集群将这个值用作集群中 etcd 端点的数量，因此该值必须与您部署的 control plane 机器数量匹配。

8 您在 DNS 记录中指定的集群名称。

9 vCenter 服务器的完全限定主机名或 IP 地址。

10 用于访问服务器的用户名。此用户必须至少在 vSphere 中有静态或动态持久性卷部署所需的角色和权限。

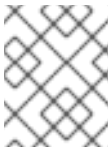
11 与 vSphere 用户关联的密码。

12 vSphere 数据中心。

13 要使用的默认 vSphere 数据存储。

14 对于 **bastion_host_name**，请指定您在镜像 registry 证书中指定的 registry 域名；对于 **<credentials>**，请指定您的镜像 registry 的 base64 编码用户名和密码。

15 Red Hat Enterprise Linux CoreOS (RHCOS) 中 **core** 用户的默认 SSH 密钥的公钥部分。



注意

对于您要在其上执行安装调试或灾难恢复的生产环境 OpenShift Container Platform 集群，请指定 **ssh-agent** 进程使用的 SSH 密钥。

16 提供用于镜像 registry 的证书文件内容。

17 提供命令输出中的 **imageContentSources** 部分来镜像存储库。

1.3.7.2. 在安装过程中配置集群范围代理

生产环境可能会拒绝直接访问互联网，而是提供 HTTP 或 HTTPS 代理。您可以通过在 **install-config.yaml** 文件中配置代理设置，将新的 OpenShift Container Platform 集群配置为使用代理。

先决条件

- 现有的 **install-config.yaml** 文件。
- 查看集群需要访问的站点，并决定是否需要绕过代理。默认情况下代理所有集群出口流量，包括对托管云供应商 API 的调用。若有需要，将站点添加到 Proxy 对象的 **spec.noProxy** 字段来绕过代理服务器。

注意

Proxy 对象的 **status.noProxy** 字段默认填充实例元数据端点 (**169.254.169.254**)，以及您的安装配置中 **networking.machineCIDR**、**networking.clusterNetwork.cidr** 和 **networking.serviceNetwork** 字段的值。

流程

1. 编辑 **install-config.yaml** 文件并添加代理设置。例如：

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: http://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
...
```

1 用于创建集群外 HTTP 连接的代理 URL。URL 必须是 **http**。

2 用于创建集群外 HTTPS 连接的代理 URL。如果未指定此字段，**httpProxy** 会同时用于 HTTP 和 HTTPS 连接。URL 方案必须是 **http**；目前不支持 **https**。

3 要排除代理的目标域名、域、IP 地址或其他网络 CIDR 的逗号分隔列表。域之前加上前缀 **.** 可包含该域的所有子域。使用 * 可对所有目的地绕过所有代理。

4 如果提供，安装程序会在 **openshift-config** 命名空间中生成名为 **user-ca-bundle** 的 ConfigMap，其包含代理 HTTPS 连接所需的一个或多个额外 CA 证书。然后，Cluster

注意

安装程序不支持代理的 **readinessEndpoints** 字段。

2. 保存该文件，并在安装 OpenShift Container Platform 时引用。

安装程序会创建一个名为 **cluster** 的集群范围代理，该代理使用提供的 **install-config.yaml** 文件中的代理设置。如果没有提供代理设置，仍然会创建 **cluster** Proxy 对象，但它会有一个零 **spec**。

注意

只支持名为 **cluster** 的 Proxy 对象，且无法创建额外的代理。

1.3.8. 创建 Kubernetes 清单和 Ignition 配置文件

由于您必须修改一些集群定义文件并要手动启动集群机器，因此您必须生成 Kubernetes 清单和 Ignition 配置文件，集群需要这两项来创建其机器。

重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须完成集群安装，并使集群以非降级状态运行 24 小时，以确保完成第一次证书轮转。

先决条件

- 获取 OpenShift Container Platform 安装程序。对于受限网络安装，这些文件位于您的堡垒主机上。
- 创建 **install-config.yaml** 安装配置文件。

流程

1. 为集群生成 Kubernetes 清单：

```
$ ./openshift-install create manifests --dir=<installation_directory> 1

WARNING There are no compute nodes specified. The cluster will not fully initialize without
compute nodes.
INFO Consuming "Install Config" from target directory
```

1 对于 **<installation_directory>**，请指定含有您创建的 **install-config.yaml** 文件的安装目录。

由于您稍后会在安装过程中自行创建计算机器，因此可以忽略这个警告。

2. 修改 **manifests/cluster-scheduler-02-config.yml** Kubernetes 清单文件，以防止在 control plane 机器上调度 Pod：

   a. 打开 **manifests/cluster-scheduler-02-config.yml** 文件。

b. 找到 **mastersSchedulable** 参数，并将其值设为 **False**。

c. 保存并退出文件。

注意

目前，由于 Kubernetes 限制，入口负载均衡器将无法访问在 control plane 机器上运行的路由器 Pod。以后的 OpenShift Container Platform 次要版本中可能不需要这一步骤。

3. 获取 Ignition 配置文件：

```
$ ./openshift-install create ignition-configs --dir=<installation_directory> 1
```

1 对于 **<installation_directory>**，请指定相同的安装目录。

该目录中将生成以下文件：

```
.
├── auth
│   ├── kubeadmin-password
│   └── kubeconfig
├── bootstrap.ign
├── master.ign
├── metadata.json
└── worker.ign
```

1.3.9. 在 vSphere 中创建 Red Hat Enterprise Linux CoreOS (RHCOS) 机器

在 VMware vSphere 上安装包含用户置备基础架构的集群前，您必须在 vSphere 主机上创建 RHCOS 机器供其使用。

先决条件

- 获取集群的 Ignition 配置文件。
- 具有 HTTP 服务器的访问权限，以便您可从计算机进行访问，并且您创建的机器也可访问此服务器。
- 创建 vSphere 集群。

流程

1. 将名为 **<installation_directory>/bootstrap.ign** 的 bootstrap Ignition 配置文件上传到 HTTP 服务器，该配置文件是由安装程序创建的。记下此文件的 URL。
   您必须托管 bootstrap Ignition 配置文件，因为它太大而无法放在 vApp 属性中。
2. 将 bootstrap 节点的以下辅助 Ignition 配置文件保存到计算机中，存为 **<installation_directory>/append-bootstrap.ign**。

```
{
  "ignition": {
    "config": {
```

```
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

1 指定您托管的 bootstrap Ignition 配置文件的 URL。

为 bootstrap 机器创建虚拟机 (VM) 时，您要使用此 Ignition 配置文件。

3. 将 master、worker 和辅助 Bootstrap Ignition 配置文件转换为 Base64 编码。例如，如果您使用 Linux 操作系统，可以使用 **base64** 命令来编码这些文件。

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. 从红帽客户门户上的产品下载页面或 RHCOS 镜像页面，获取 RHCOS OVA 镜像。

重要

RHCOS 镜像可能不会随着 OpenShift Container Platform 的每一发行版本都有改变。您必须下载一个最高版本的镜像，其版本号应小于或等于您安装的 OpenShift Container Platform 版本。如果可用，请使用与 OpenShift Container Platform 版本匹配的镜像版本。

文件名包含 OpenShift Container Platform 版本号，格式为 **rhcos-<version>-<architecture>-vmware.ova**。

5. 在 vSphere 客户端中，在数据中心中创建一个文件夹来存储您的虚拟机。

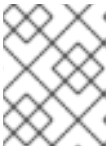
   a. 点击 **VMs and Templates** 视图。

   b. 右键点击您的数据中心名称。

   c. 点击 **New Folder → New VM and Template Folder**。

   d. 在显示的窗口中输入文件夹名称。文件夹名称必须与您在 **install-config.yaml** 文件中指定的集群名称匹配。

6. 在 vSphere 客户端中，为 OVA 镜像创建一个模板。



注意

在以下步骤中，您要将同一模板用于所有集群机器，并针对您在置备虚拟机时指定的机器类型提供 Ignition 配置文件的位置。

a. 在 **Hosts and Clusters** 选项卡中，右键点击集群名称并点击 **Deploy OVF Template**。

b. 在 **Select an OVF** 选项卡中，指定您下载的 RHCOS OVA 文件的名称。

c. 在 **Select a name and folder**选项卡中，设置 **Virtual machine name**，如 RHCOS，再点击 vSphere 集群的名称，然后选择您在上一步中创建的文件夹。

d. 在 **Select a compute resource** 选项卡中，点击您的 vSphere 集群名称。

e. 在 **Select storage** 选项卡中，配置虚拟机的存储选项。

- 选择 **Thin Provision**。
- 选择您在 **install-config.yaml** 文件中指定的数据存储。

f. 在 **Select network** 选项卡中，指定您为集群配置的网络（如果可用）。

g. 如果您计划将同一模板用于所有集群机器类型，请不要在 **Customize template** 选项卡中指定值。

7. 部署模板后，为集群中的机器部署虚拟机。

a. 右键点击模板的名称，再点击 **Clone → Clone to Virtual Machine**。

b. 在 **Select a name and folder**选项卡中，指定虚拟机的名称。名称中可以包括机器类型，如 **control-plane-0** 或 **compute-1**。

c. 在 **Select a name and folder**选项卡中，选择您为集群创建的文件夹名称。

d. 在 **Select a compute resource** 选项卡中，选择数据中心中的主机名称。

e. 可选：在 **Select storage** 选项卡中，自定义存储选项。

f. 在 **Select clone options** 中，选择 **Customize this virtual machine's hardware**。

g. 在 **Customize hardware** 选项卡中，点击 **VM Options → Advanced**。

- 可选：在出现集群性能问题时，从 **Latency Sensitivity** 列表中选择 **High**。
- 点击 **Edit Configuration**，然后在 **Configuration Parameters** 窗口中点击 **Add Configuration Params**。定义以下参数名称和值：
  - **guestinfo.ignition.config.data**：粘贴此机器类型的 base64 编码 Ignition 配置文件的内容。
  - **guestinfo.ignition.config.data.encoding**：指定 **base64**。
  - **disk.EnableUUID**：指定 **TRUE**。
- 另外，也在打开虚拟机电源前通过 vApp 属性添加：
  - 导航到 vCenter 服务器清单中的某一虚拟机。

- 在 **Configure** 选项卡中，展开 **Settings** 并选择 **vApp options**。
- 向下滚动，并在 **Properties** 下应用上述配置。

h. 在 **Customize hardware** 选项卡的 **Virtual Hardware** 面板中，根据需要修改指定的值。确保 RAM、CPU 和磁盘存储的数量满足机器类型的最低要求。

i. 完成配置并打开虚拟机电源。

8. 对于每台机器，按照前面的步骤为集群创建其余的机器。

重要

此刻您必须创建 bootstrap 和 control plane 机器。由于计算机器上已默认部署了一些 Pod，因此在安装集群前，还要创建至少两台计算机器。

1.3.10. 创建集群

要创建 OpenShift Container Platform 集群，请等待您通过安装程序生成的 Ignition 配置文件所置备的机器上完成 bootstrap 过程。

先决条件

- 为集群创建所需的基础架构。
- 已获得安装程序并为集群生成了 Ignition 配置文件。
- 已使用 Ignition 配置文件为集群创建 RHCOS 机器。

流程

1. 监控 bootstrap 过程：

```
$ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
    --log-level=info 2
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.14.6+c4799753c up
INFO Waiting up to 30m0s for the bootstrap-complete event...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2 要查看不同的安装详情，请指定 **warn**、**debug** 或 **error**，而不要指定 **info**。

Kubernetes API 服务器提示已在 control plane 机器上完成 bootstrap 时，命令运行成功。

2. bootstrap 过程完成后，请从负载均衡器中删除 bootstrap 机器。

重要

此时您必须从负载均衡器中删除 bootstrap 机器。您还可以删除或重新格式化机器本身。

1.3.11. 登录集群

您可以通过导出集群 **kubeconfig** 文件，以默认系统用户身份登录集群。**kubeconfig** 文件包含关于集群的信息，供 CLI 用于将客户端连接到正确集群和 API 服务器。该文件特只适用于一个特定的集群，在 OpenShift Container Platform 安装过程中创建。

先决条件

- 部署 OpenShift Container Platform 集群。
- 安装 **oc** CLI。

流程

1. 导出 **kubeadmin** 凭证：

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

2. 使用导出的配置，验证能否成功运行 **oc** 命令：

```
$ oc whoami
system:admin
```

1.3.12. 批准机器的 CSR

将机器添加到集群时，系统会为添加的每台机器生成两个待处理证书签名请求 (CSR)。您必须确认这些 CSR 已获得批准，或根据需要自行批准。

先决条件

- 您已将机器添加到集群中。

流程

1. 确认集群可以识别这些机器：

```
$ oc get nodes

NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.14.6+c4799753c
master-1  Ready     master  63m  v1.14.6+c4799753c
master-2  Ready     master  64m  v1.14.6+c4799753c
worker-0  NotReady  worker  76s  v1.14.6+c4799753c
worker-1  NotReady  worker  70s  v1.14.6+c4799753c
```

输出将列出您创建的所有机器。

2. 检查待处理证书签名请求 (CSR)，并确保您添加到集群中的每一机器都有状态为 **Pending** 或 **Approved** 的客户端和服务器请求：

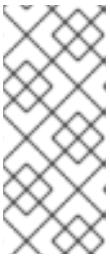
```
$ oc get csr

NAME        AGE     REQUESTOR                                                   CONDITION
```

```
csr-8b2br   15m    system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending (1)
csr-8vnps   15m    system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending
csr-bfd72   5m26s   system:node:ip-10-0-50-126.us-east-2.compute.internal
Pending (2)
csr-c57lv   5m26s   system:node:ip-10-0-95-157.us-east-2.compute.internal
Pending
...
```

1. 客户端请求 CSR。
2. 服务器请求 CSR。

在本例中，两台机器加入了集群。您可能会在列表中看到更多已批准的 CSR。

3. 如果 CSR 没有获得批准，请在所添加机器的所有待处理 CSR 都处于 **Pending** 状态后，为您的集群机器批准这些 CSR：



注意

由于 CSR 会自动轮转，因此请在将机器添加到集群后一小时内批准您的 CSR。如果没有在一小时内批准，证书将会轮转，每个节点将会存在多个证书。您必须批准所有这些证书。批准初始 CSR 后，集群的 **kube-controller-manager** 会自动批准后续的节点客户端 CSR。您必须实施一个方法来自动批准 kubelet 提供的证书请求。

- 若要单独批准，请对每个有效的 CSR 运行以下命令：

```
$ oc adm certificate approve <csr_name> (1)
```

1. **<csr_name>** 是当前 CSR 列表中 CSR 的名称。

- 要批准所有待处理的 CSR，请运行以下命令：

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}
{{end}}{{end}}' | xargs oc adm certificate approve
```

1.3.13. 初始 Operator 配置

在 control plane 初始化后，您必须立即配置一些 Operator 以便它们都可用。

先决条件

- 您的 control plane 已初始化。

流程

1. 观察集群组件上线：

```
$ watch -n5 oc get clusteroperators
```

```
NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                             4.2.0     True        False         False     69s
cloud-credential                           4.2.0     True        False         False     12m
cluster-autoscaler                         4.2.0     True        False         False     11m
console                                    4.2.0     True        False         False     46s
dns                                        4.2.0     True        False         False     11m
image-registry                             4.2.0     False       True          False     5m26s
ingress                                    4.2.0     True        False         False     5m36s
kube-apiserver                             4.2.0     True        False         False     8m53s
kube-controller-manager                    4.2.0     True        False         False     7m24s
kube-scheduler                             4.2.0     True        False         False     12m
machine-api                                4.2.0     True        False         False     12m
machine-config                             4.2.0     True        False         False     7m36s
marketplace                                4.2.0     True        False         False     7m54m
monitoring                                 4.2.0     True        False         False     7h54s
network                                    4.2.0     True        False         False     5m9s
node-tuning                                4.2.0     True        False         False     11m
openshift-apiserver                        4.2.0     True        False         False     11m
openshift-controller-manager               4.2.0     True        False         False     5m943s
openshift-samples                          4.2.0     True        False         False     3m55s
operator-lifecycle-manager                 4.2.0     True        False         False     11m
operator-lifecycle-manager-catalog         4.2.0     True        False         False     11m
service-ca                                 4.2.0     True        False         False     11m
service-catalog-apiserver                  4.2.0     True        False         False     5m26s
service-catalog-controller-manager         4.2.0     True        False         False     5m25s
storage                                    4.2.0     True        False         False     5m30s
```

2. 配置不可用的 Operator。

1.3.13.1. 镜像 registry 存储配置

如果 **image-registry** Operator 不可用，您必须为其配置存储。提供了配置 PersistentVolume 的说明，这是生产集群所需要的；也提供了将空目录配置为存储位置的说明，这仅适用于非生产集群。

1.3.13.1.1. 为 VMware vSphere 配置 registry 存储

作为集群管理员，在安装后需要配置 registry 来使用存储。

先决条件

- 具有 Cluster Administrator 权限
- VMware vSphere上有一个集群。
- 具有**ReadWriteMany**访问模式的持久性存储卷（PV），例如**NFS** 。

重要

vSphere卷不支持**ReadWriteMany**访问模式。您必须使用其他存储后端（例如**NFS** ）来配置 registry 注存储。

- 必须有" 100Gi"容量。

流程

1. 为了配置 registry 使用存储，需要修改 **configs.imageregistry/cluster** 资源中的 **spec.storage.pvc** 。
2. 验证您没有 registry Pod：

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

注意

如果存储类型为 **emptyDIR**，则副本数不能超过 **1**。如果存储类型为**NFS** ，并且希望通过设置 **replica>1**来扩展 registry Pod，则必须启用**no_wdelay**挂载选项。例如：

```
# cat /etc/exports
/mnt/data *(rw,sync,no_wdelay,no_root_squash,insecure,fsid=0)
sh-4.3# exportfs -rv
exporting *:/mnt/data
```

3. 检查 registry 配置：

```
$ oc edit configs.imageregistry.operator.openshift.io

storage:
  pvc:
    claim:
```

将 **claim** 字段留空以允许自动创建一个 **image-registry-storage** PVC。

4. 可选：在 PV 中添加新存储类：
   a. 创建 PV：

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    ReadWriteMany
    capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

b. 创建 PVC：

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

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

最后，添加 PVC 的名称：

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc (1)
```

(1) 通过创建自定义 PVC，您可以将 **claim** 字段留空以用于默认自动创建 **image-registry-storage** PVC。

5. 检查 **clusteroperator** 的状态：

```
$ oc get clusteroperator image-registry
```

1.3.13.1.2. 在非生产集群中配置镜像 registry 存储

您必须为 Image Registry Operator 配置存储。对于非生产集群，您可以将镜像 registry 设置为空目录。如果您这样做，重启 registry 后会丢失所有镜像。

流程

- 将镜像 registry 存储设置为空目录：

```
$ oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

警告

仅可为非生产集群配置这个选项。

如果在 Image Registry Operator 初始化其组件前运行此命令，**oc patch** 命令会失败并显示以下错误：

```
Error from server (NotFound): configs.imageregistry.operator.openshift.io "cluster" not found
```

等待几分钟，然后再次运行该命令。

1.3.14. 在用户置备的基础架构上完成安装

完成 Operator 配置后，可以在您提供的基础架构上完成集群安装。

先决条件

- 您的 control plane 已初始化。
- 已完成初始 Operator 配置。

流程

1. 确认所有集群组件都已上线：

```
$ watch -n5 oc get clusteroperators

NAME                                 VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                       4.2.0     True        False         False      10m
cloud-credential                     4.2.0     True        False         False      22m
cluster-autoscaler                   4.2.0     True        False         False      21m
console                              4.2.0     True        False         False      10m
dns                                  4.2.0     True        False         False      21m
image-registry                       4.2.0     True        False         False      16m
ingress                              4.2.0     True        False         False      16m
kube-apiserver                       4.2.0     True        False         False      19m
kube-controller-manager              4.2.0     True        False         False      18m
kube-scheduler                       4.2.0     True        False         False      22m
machine-api                          4.2.0     True        False         False      22m
machine-config                       4.2.0     True        False         False      18m
marketplace                          4.2.0     True        False         False      18m
monitoring                           4.2.0     True        False         False      18m
network                              4.2.0     True        False         False      16m
node-tuning                          4.2.0     True        False         False      21m
openshift-apiserver                  4.2.0     True        False         False      21m
openshift-controller-manager         4.2.0     True        False         False      17m
openshift-samples                    4.2.0     True        False         False      14m
operator-lifecycle-manager           4.2.0     True        False         False      21m
operator-lifecycle-manager-catalog   4.2.0     True        False         False      21m
```

```
service-ca                                 4.2.0     True        False         False      21m
service-catalog-apiserver                  4.2.0     True        False         False      16m
service-catalog-controller-manager         4.2.0     True        False         False      16m
storage                                    4.2.0     True        False         False      16m
```

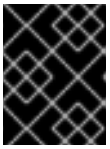
当所有集群 Operator 状态都是 **AVAILABLE** 时，您可以完成安装。

2. 监控集群完成：

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
INFO Waiting up to 30m0s for the cluster to initialize...
```

1 对于 **<installation_directory>**，请指定安装文件保存到的目录的路径。

Cluster Version Operator 完成从 Kubernetes API 服务器部署 OpenShift Container Platform 集群时，命令运行成功。



重要

安装程序生成的 Ignition 配置文件中所含的证书会在 24 小时后过期。您必须以非降级状态持续运行集群 24 小时，以确保完成第一次证书轮转。

3. 确认 Kubernetes API 服务器正在与 Pod 通信。

   a. 要查看所有 Pod 的列表，请使用以下命令：

```
$ oc get pods --all-namespaces

NAMESPACE                       NAME                                         READY   STATUS
RESTARTS   AGE
openshift-apiserver-operator    openshift-apiserver-operator-85cb746d55-zqhs8   1/1
Running    1          9m
openshift-apiserver             apiserver-67b9g                               1/1     Running     0
3m
openshift-apiserver             apiserver-ljcmx                               1/1     Running     0
1m
openshift-apiserver             apiserver-z25h4                               1/1     Running     0
2m
openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8       1/1
Running     0          5m
...
```

   b. 使用以下命令，查看上一命令的输出中所列 Pod 的日志：

```
$ oc logs <pod_name> -n <namespace> 1
```

1 指定 Pod 名称和命名空间，如上一命令的输出中所示。

如果 Pod 日志显示，Kubernetes API 服务器可以与集群机器通信。

4. 在 Cluster registration 页面注册您的集群。

后续步骤

- 自定义集群。
- 若有需要，您可以选择不使用远程健康报告。