



# OpenShift Container Platform 4.6

## RHV へのインストール

OpenShift Container Platform RHV クラスターのインストール



# OpenShift Container Platform 4.6 RHV へのインストール

---

## OpenShift Container Platform RHV クラスターのインストール

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Installing\_on\_RHV.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

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

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

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

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

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

本書では、Red Hat Virtualization に OpenShift Container Platform クラスターをインストールし、アンインストールする方法について説明します。

## 目次

<b>第1章 RHV へのインストール</b> .....	<b>4</b>
1.1. {RH-VIRTUALIZATION} へのクラスターのクイックインストール	4
1.1.1. 前提条件	4
1.1.2. OpenShift Container Platform のインターネットアクセス	5
1.1.3. RHV 環境の要件	5
1.1.4. RHV 環境の要件の確認	7
1.1.5. RHV でのネットワーク環境の準備	8
1.1.6. RHV 用の CA 証明書の設定	9
1.1.7. SSH プライベートキーの生成およびエージェントへの追加	10
1.1.8. インストールプログラムの取得	12
1.1.9. クラスターのデプロイ	12
1.1.10. バイナリーのダウンロードによる OpenShift CLI のインストール	15
1.1.10.1. Linux への OpenShift CLI のインストール	16
1.1.10.2. Windows への OpenShift CLI のインストール	16
1.1.10.3. macOS への OpenShift CLI のインストール	17
1.1.11. CLI の使用によるクラスターへのログイン	17
1.1.12. クラスターステータスの確認	18
1.1.13. RHV での OpenShift Container Platform Web コンソールへのアクセス	19
1.1.14. OpenShift Container Platform の Telemetry アクセス	19
1.1.15. Red Hat Virtualization (RHV) へのインストールに関するよくある問題のトラブルシューティング	19
1.1.15.1. CPU 負荷が増大し、ノードが Not Ready 状態になる	19
1.1.15.2. OpenShift Container Platform クラスター API に接続できない	20
1.1.16. インストール後のタスク	20
1.2. カスタマイズによる RHV へのクラスターのインストール	20
1.2.1. 前提条件	21
1.2.2. OpenShift Container Platform のインターネットアクセス	22
1.2.3. RHV 環境の要件	22
1.2.4. RHV 環境の要件の確認	24
1.2.5. RHV でのネットワーク環境の準備	25
1.2.6. RHV 用の CA 証明書の設定	26
1.2.7. SSH プライベートキーの生成およびエージェントへの追加	27
1.2.8. インストールプログラムの取得	29
1.2.9. インストール設定ファイルの作成	29
1.2.9.1. Red Hat Virtualization (RHV) のサンプル install-config.yaml ファイル	32
1.2.9.2. インストール設定パラメーター	34
1.2.9.2.1. 必須設定パラメーター	34
1.2.9.2.2. ネットワーク設定パラメーター	36
1.2.9.2.3. オプションの設定パラメーター	37
1.2.9.2.4. 追加の Red Hat Virtualization (RHV) 設定パラメーター	41
1.2.9.2.5. マシンプールの追加 RHV パラメーター	42
1.2.10. クラスターのデプロイ	43
1.2.11. バイナリーのダウンロードによる OpenShift CLI のインストール	45
1.2.11.1. Linux への OpenShift CLI のインストール	45
1.2.11.2. Windows への OpenShift CLI のインストール	46
1.2.11.3. macOS への OpenShift CLI のインストール	46
1.2.12. CLI の使用によるクラスターへのログイン	47
1.2.13. クラスターステータスの確認	47
1.2.14. RHV での OpenShift Container Platform Web コンソールへのアクセス	48
1.2.15. OpenShift Container Platform の Telemetry アクセス	49
1.2.16. Red Hat Virtualization (RHV) へのインストールに関するよくある問題のトラブルシューティング	49
1.2.16.1. CPU 負荷が増大し、ノードが Not Ready 状態になる	49

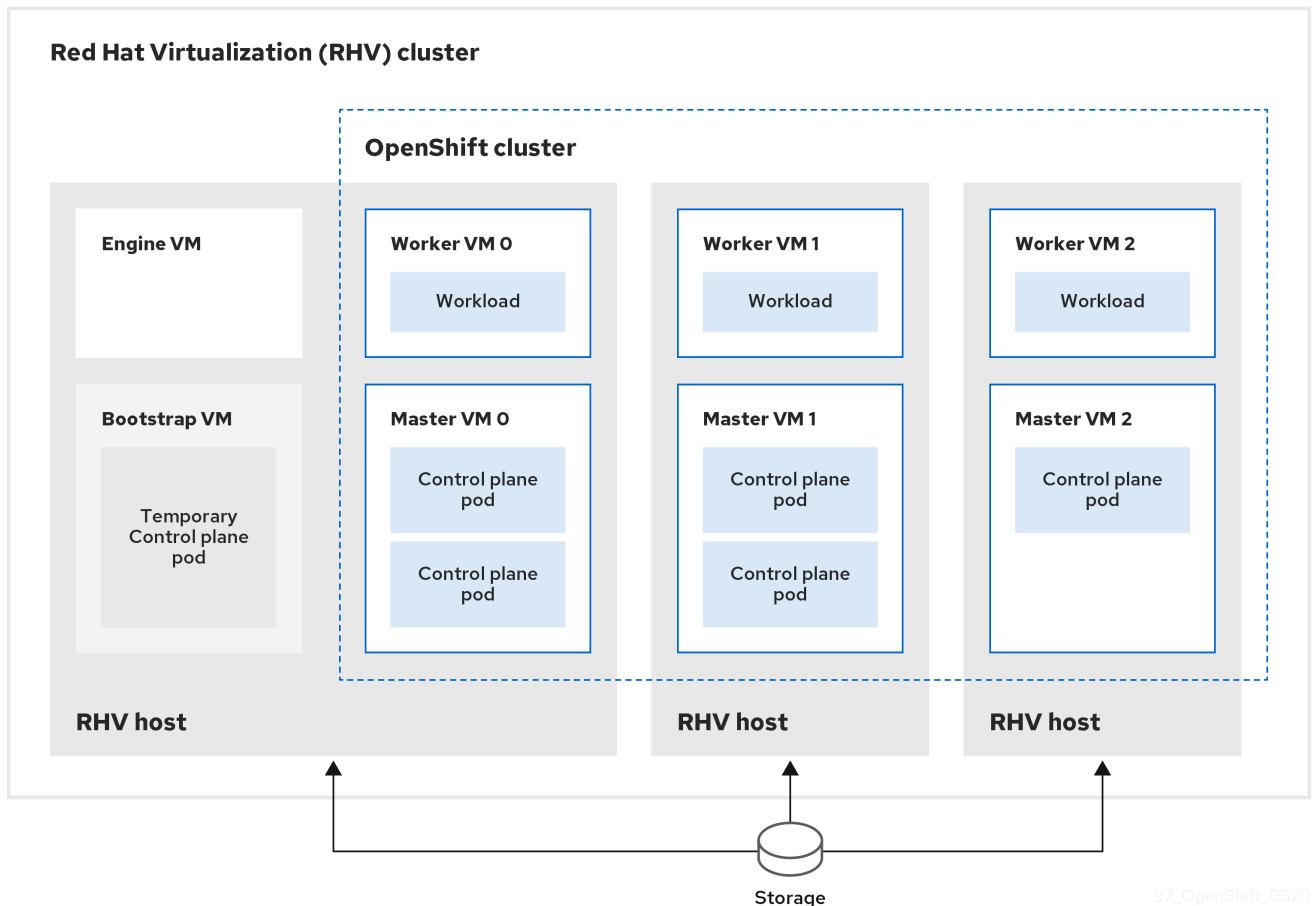
1.2.16.2. OpenShift Container Platform クラスター API に接続できない	49
1.2.17. インストール後のタスク	50
1.2.18. 次のステップ	50
1.3. ユーザーによってプロビジョニングされるインフラストラクチャーを使用した RHV へのクラスターのインストール	50
1.3.1. 前提条件	51
1.3.2. OpenShift Container Platform のインターネットアクセス	51
1.3.3. RHV 環境の要件	52
1.3.4. RHV 環境の要件の確認	53
1.3.5. ユーザーによってプロビジョニングされるインフラストラクチャーのネットワーク要件	55
ネットワークポロジータン要件	57
ロードバランサー	57
NTP 設定	59
1.3.6. インストールマシンの設定	59
1.3.7. RHV 用の CA 証明書の設定	60
1.3.8. SSH プライベートキーの生成およびエージェントへの追加	61
1.3.9. インストールプログラムの取得	62
1.3.10. Ansible Playbook のダウンロード	63
1.3.11. inventory.yml ファイル	64
1.3.12. RHCOS イメージ設定の指定	68
1.3.13. インストール設定ファイルの作成	68
1.3.14. install-config.yaml のカスタマイズ	69
1.3.15. マニフェストファイルの生成	70
1.3.16. コントロールプレーンノードのスケジュール対象外の設定	72
1.3.17. Ignition ファイルのビルド	72
1.3.18. テンプレートおよび仮想マシンの作成	73
1.3.19. ブートストラップマシンの作成	73
1.3.20. コントロールプレーンノードの作成	74
1.3.21. クラスターステータスの確認	75
1.3.22. ブートストラップマシンの削除	75
1.3.23. ワーカーノードの作成およびインストールの完了	76
1.3.24. OpenShift Container Platform の Telemetry アクセス	77
1.4. RHV でのクラスターのアンインストール	78
1.4.1. インストーラーでプロビジョニングされるインフラストラクチャーを使用するクラスターの削除	78
1.4.2. ユーザーによってプロビジョニングされるインフラストラクチャーを使用するクラスターの削除	79



## 第1章 RHV へのインストール

### 1.1. {RH-VIRTUALIZATION} へのクラスタのクイックインストール

以下の図に示されるように、デフォルトの、カスタマイズされていない OpenShift Container Platform クラスタを Red Hat Virtualization (RHV) クラスタにすばやくインストールできます。



92\_OpenShift\_0520

インストールプログラムは、インストーラーでプロビジョニングされるインフラストラクチャーを使用してクラスタの作成およびデプロイを自動化します。

デフォルトのクラスタをインストールするには、環境を準備し、インストールプログラムを実行してプロンプトに応答します。次に、インストールプログラムは OpenShift Container Platform クラスタを作成します。

デフォルトクラスタの代替インストール方法については、[カスタマイズによるクラスタのインストール](#) について参照してください。



#### 注記

このインストールプログラムは、Linux および macOS でのみ利用できます。

#### 1.1.1. 前提条件

- [OpenShift Container Platform のインストールおよび更新](#) プロセスについての詳細を確認します。



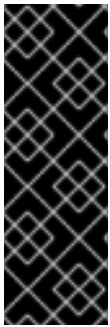
- [Support Matrix for OpenShift Container Platform on Red Hat Virtualization \(RHV\)](#) に記載のあるサポートされるバージョンの組み合わせを使用できる。
- ファイアウォールを使用する場合、クラスターがアクセスする必要のある [サイトを許可するようにファイアウォールを設定](#) します。

### 1.1.2. OpenShift Container Platform のインターネットアクセス

OpenShift Container Platform 4.6 では、クラスターをインストールするためにインターネットアクセスが必要になります。

インターネットへのアクセスは以下を実行するために必要です。

- [OpenShift Cluster Manager](#) にアクセスし、インストールプログラムをダウンロードし、サブスクリプション管理を実行します。クラスターにインターネットアクセスがあり、Telemetry を無効にしない場合、そのサービスは有効なサブスクリプションでクラスターを自動的に使用します。
- クラスターのインストールに必要なパッケージを取得するために [Quay.io](#) にアクセスします。
- クラスターの更新を実行するために必要なパッケージを取得します。



#### 重要

クラスターでインターネットに直接アクセスできない場合、プロビジョニングする一部のタイプのインフラストラクチャーでネットワークが制限されたインストールを実行できます。このプロセスで、必要なコンテンツをダウンロードし、これを使用してミラーレジストリーにクラスターのインストールおよびインストールプログラムの生成に必要なパッケージを設定します。インストールタイプによっては、クラスターのインストール環境でインターネットアクセスが不要となる場合があります。クラスターを更新する前に、ミラーレジストリーのコンテンツを更新します。

### 1.1.3. RHV 環境の要件

OpenShift Container Platform クラスターをインストールし、実行するには、RHV 環境が以下の要件を満たしている必要があります。

これらの要件を満たさないと、インストールまたはプロセスが失敗する可能性があります。さらに、これらの要件を満たしていないと、OpenShift Container Platform クラスターはインストールしてから数日または数週間後に失敗する可能性があります。

CPU、メモリー、ストレージリソースについての以下の要件は、インストールプログラムが作成する仮想マシンのデフォルト数で乗算した **デフォルト** 値に基づいています。これらのリソースは、RHV 環境が OpenShift Container Platform 以外の操作に使用するものに **加え**、利用可能でなければなりません。

デフォルトでは、インストールプログラムは7つの仮想マシンをインストールプロセスで作成します。まず、ブートストラップ仮想マシンを作成し、OpenShift Container Platform クラスターの残りの部分を作成する間に一時サービスとコントロールプレーンを提供します。インストールプログラムがクラスターの作成を終了すると、ブートストラップマシンが削除され、そのリソースが解放されます。

RHV 環境の仮想マシン数を増やす場合は、リソースを適宜増やす必要があります。

#### 要件

- RHV 環境に **Up** 状態のデータセンターが1つあること。

- RHV データセンターに RHV クラスターが含まれていること。
- RHV クラスターに OpenShift Container Platform クラスター専用の以下のリソースがあること。
  - 最小 28 vCPU: インストール時に作成される 7 仮想マシンのそれぞれに 4 vCPU。
  - 以下を含む 112 GiB 以上の RAM。
    - 一時的なコントロールプレーンを提供するブートストラップマシン用に 16 GiB 以上。
    - コントロールプレーンを提供する 3 つのコントロールプレーンマシンのそれぞれに 16 GiB 以上。
    - アプリケーションワークロードを実行する 3 つのコンピュータマシンのそれぞれに 16 GiB 以上。
- RHV ストレージドメインは、[これらの etcd バックエンドのパフォーマンス要件](#) を満たす必要があります。
- 実稼働環境では、各仮想マシンに 120 GiB 以上が必要です。そのため、ストレージドメインはデフォルトの OpenShift Container Platform クラスターに 840 GiB 以上を提供する必要があります。リソースに制約のある環境または非実稼働環境では、各仮想マシンに 32 GiB 以上を指定する必要があるため、ストレージドメインにはデフォルトの OpenShift Container Platform クラスター用に 230 GiB 以上が必要になります。
- インストールおよび更新中に Red Hat Ecosystem Catalog からイメージをダウンロードするには、RHV クラスターがインターネット接続にアクセスできる必要があります。また、サブスクリプションおよびエンタイトルメントプロセスを単純化するために Telemetry サービスにもインターネット接続が必要です。
- RHV クラスターには、RHV Manager の REST API にアクセスできる仮想ネットワークが必要です。インストーラーが作成する仮想マシンが DHCP を使用して IP アドレスを取得するため、DHCP がこのネットワークで有効にされていることを確認します。
- ターゲット RHV クラスターに OpenShift Container Platform クラスターをインストールし、管理するための以下の最小限の権限を持つユーザーアカウントおよびグループ。
  - **DiskOperator**
  - **DiskCreator**
  - **UserTemplateBasedVm**
  - **TemplateOwner**
  - **TemplateCreator**
  - ターゲットクラスターの **ClusterAdmin**

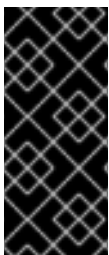


### 警告

最小権限の原則を適用します。インストールプロセスで RHV で **SuperUser** 権限を持つ管理者アカウントを使用することを避けます。インストールプログラムは、ユーザーが指定する認証情報を、危険にさらされる可能性のある一時的な **ovirt-config.yaml** ファイルに保存します。

## 1.1.4. RHV 環境の要件の確認

RHV 環境が OpenShift Container Platform クラスターをインストールし、実行するための要件を満たしていることを確認します。これらの要件を満たさないと、エラーが発生する可能性があります。



### 重要

これらの要件は、インストールプログラムがコントロールプレーンおよびコンピュータマシンの作成に使用するデフォルトのリソースに基づいています。これらのリソースには、vCPU、メモリー、およびストレージが含まれます。これらのリソースを変更するか、または OpenShift Container Platform マシンの数を増やす場合は、これらの要件を適宜調整します。

### 手順

1. RHV のバージョンを確認します。
  - a. RHV Administration Portal の右上にある ? ヘルプアイコンをクリックし、**About** を選択します。
  - b. 開かれるウィンドウで、**RHV ソフトウェアのバージョン** をメモします。
  - c. OpenShift Container Platform のバージョン 4.6 とメモした RHV のバージョンが、[Support Matrix for OpenShift Container Platform on RHV](#) でサポートされている組み合わせのいずれかであることを確認します。
2. データセンター、クラスター、およびストレージを検査します。
  - a. RHV 管理ポータルで、**Compute → Data Centers** をクリックします。
  - b. OpenShift Container Platform をインストールする予定のデータセンターにアクセスできることを確認します。
  - c. そのデータセンターの名前をクリックします。
  - d. データセンターの詳細の **Storage** タブで、OpenShift Container Platform をインストールする予定のストレージドメインが **Active** であることを確認します。
  - e. 後で使用できるように **ドメイン名** を記録します。
  - f. **空き領域** に 230 GiB 以上あることを確認します。
  - g. ストレージドメインが [これらの etcd バックエンドのパフォーマンス要件](#) を満たしていることを確認します。これは、[fio パフォーマンスベンチマークツール](#) を使用して測定できません。

- h. データセンターの詳細で、**Clusters** タブをクリックします。
  - i. OpenShift Container Platform をインストールする予定の RHV クラスタを見つけます。後で使用できるようにクラスター名を記録します。
3. RHV ホストリソースを確認します。
    - a. RHV 管理ポータルで、**Compute > Clusters** をクリックします。
    - b. OpenShift Container Platform をインストールする予定のクラスターをクリックします。
    - c. クラスタの詳細で、**Hosts** タブをクリックします。
    - d. ホストを検査し、それらに OpenShift Container Platform クラスタ **専用** として利用可能な **論理 CPU コア** の合計が 28 つ以上であることを確認します。
    - e. 後で使用できるように、利用可能な **論理 CPU コア** の数を記録します。
    - f. これらの CPU コアが分散され、インストール時に作成された 7 つの仮想マシンのそれぞれに 4 つのコアを持たせることができることを確認します。
    - g. ホストには、以下の OpenShift Container Platform マシンのそれぞれの要件を満たすように **新規仮想マシンをスケジュールするための最大空きメモリー** として 112 GiB があることを確認します。
      - ブートストラップマシンに 16 GiB が必要です。
      - 3 つのコントロールプレーンマシンのそれぞれに 16 GiB が必要です。
      - 3 つのコンピュートマシンのそれぞれに 16 GiB が必要です。
    - h. 後で使用できるように **新規仮想マシンをスケジュールするための最大空きメモリー** の量を記録します。
  4. OpenShift Container Platform をインストールするための仮想ネットワークが RHV Manager の REST API にアクセスできることを確認します。このネットワーク上の仮想マシンから、RHV Manager の REST API に到達するために curl を使用します。

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

**1** **<username>** については、RHV で OpenShift Container Platform クラスタを作成および管理する権限を持つ RHV アカウントのユーザー名を指定します。**<profile>** には、ログインプロファイルを指定します。ログインプロファイルは、RHV Administration Portal ログインページに移動し、**Profile** ドロップダウンリストで確認できます。**<password>** に、そのユーザー名のパスワードを指定します。

**2** **<engine-fqdn>** に、RHV 環境の完全修飾ドメイン名を指定します。

以下に例を示します。

```
$ curl -k -u ocpadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

### 1.1.5. RHV でのネットワーク環境の準備

OpenShift Container Platform クラスターの 2 つの静的 IP アドレスを設定し、これらのアドレスを使用して DNS エントリーを作成します。

## 手順

1. 2 つの静的 IP アドレスを予約します。
  - a. OpenShift Container Platform をインストールするネットワークで、DHCP リースプール外にある 2 つの静的 IP アドレスを特定します。
  - b. このネットワーク上のホストに接続し、それぞれの IP アドレスが使用されていないことを確認します。たとえば、Address Resolution Protocol (ARP) を使用して、IP アドレスのいずれにもエントリーがないことを確認します。

```
$ arp 10.35.1.19
```

## 出力例

```
10.35.1.19 (10.35.1.19) -- no entry
```

- c. ネットワーク環境の標準的な方法に従って、2 つの静的 IP アドレスを予約します。
  - d. 今後の参照用にこれらの IP アドレスを記録します。
2. 以下の形式を使用して、OpenShift Container Platform REST API およびアプリケーションドメイン名の DNS エントリーを作成します。

```
api.<cluster-name>.<base-domain> <ip-address> ①  
*.apps.<cluster-name>.<base-domain> <ip-address> ②
```

- ① **<cluster-name>**、**<base-domain>**、および **<ip-address>** には、クラスター名、ベースドメイン、および OpenShift Container Platform API の静的 IP アドレスを指定します。
- ② Ingress およびロードバランサー用に OpenShift Container Platform アプリケーションのクラスター名、ベースドメイン、および静的 IP アドレスを指定します。

以下に例を示します。

```
api.my-cluster.virtlab.example.com 10.35.1.19  
*.apps.my-cluster.virtlab.example.com 10.35.1.20
```

### 1.1.6. RHV 用の CA 証明書の設定

Red Hat Virtualization (RHV) Manager から CA 証明書をダウンロードし、インストールマシンにこれを設定します。

RHV Manager からの Web サイトまたは **curl** コマンドを使用して、証明書をダウンロードできます。

その後、インストールプログラムに証明書を提供します。

## 手順

1. 以下の 2 つの方法のいずれかを使用して CA 証明書をダウンロードします。

- Manager の Web ページ (<https://<engine-fqdn>/ovirt-engine/>) に移動します。次に、**Downloads** で **CA Certificate** のリンクをクリックします。
- 以下のコマンドを実行します。

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem 1
```

- 1 **<engine-fqdn>** には、RHV Manager の完全修飾ドメイン名 (例: **rhv-env.virtlab.example.com**) を指定します。

2. ルートレスユーザーに Manager へのアクセスを付与するように CA ファイルを設定します。CA ファイルのパーミッションを 8 進数の **0644** に設定します (シンボリック値: **-rw-r--r--**):

```
$ sudo chmod 0644 /tmp/ca.pem
```

3. Linux の場合は、サーバー証明書のディレクトリーに CA 証明書をコピーします。-p を使用してパーミッションを保存します。

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

4. オペレーティングシステム用の証明書マネージャーに証明書を追加します。

- MacOS の場合は、証明書ファイルをダブルクリックして、**Keychain Access** ユーティリティを使用してファイルを **System** キーチェーンに追加します。
- Linux の場合は、CA 信頼を更新します。

```
$ sudo update-ca-trust
```



### 注記

独自の認証局を使用する場合は、システムがこれを信頼することを確認します。

## 関連情報

- 詳細は、RHV ドキュメントの [Authentication and Security](#) を参照してください。

### 1.1.7. SSH プライベートキーの生成およびエージェントへの追加

クラスターでインストールのデバッグまたは障害復旧を実行する必要がある場合、**ssh-agent** とインストールプログラムの両方に SSH キーを指定する必要があります。このキーを使用してパブリッククラスターのブートストラップマシンにアクセスし、インストールの問題をトラブルシューティングできます。



### 注記

実稼働環境では、障害復旧およびデバッグが必要です。

このキーを使用して、ユーザー **core** としてマスターノードに対して SSH を実行できます。クラスターをデプロイする際に、キーは **core** ユーザーの **~/.ssh/authorized\_keys** 一覧に追加されます。

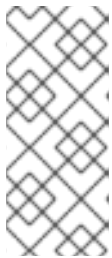
## 手順

1. パスワードなしの認証に設定されている SSH キーがコンピューター上にない場合は、これを作成します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ ssh-keygen -t ed25519 -N "" \
-f <path>/<file_name> ①
```

- ① `~/.ssh/id_rsa` などの、新規 SSH キーのパスおよびファイル名を指定します。既存のキーペアがある場合は、公開鍵が `~/.ssh` ディレクトリーにあることを確認します。

このコマンドを実行すると、指定した場所にパスワードを必要としない SSH キーが生成されます。



## 注記

FIPS で検証済み/進行中のモジュール (Modules in Process) 暗号ライブラリーを使用する OpenShift Container Platform クラスターを **x86\_64** アーキテクチャーにインストールする予定の場合は、**ed25519** アルゴリズムを使用するキーは作成しないでください。代わりに、**rsa** アルゴリズムまたは **ecdsa** アルゴリズムを使用するキーを作成します。

2. **ssh-agent** プロセスをバックグラウンドタスクとして開始します。

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

## 出力例

```
Agent pid 31874
```



## 注記

クラスターが FIPS モードにある場合は、FIPS 準拠のアルゴリズムのみを使用して SSH キーを生成します。鍵は RSA または ECDSA のいずれかである必要があります。

3. SSH プライベートキーを **ssh-agent** に追加します。

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

## 出力例

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

- ① `~/.ssh/id_rsa` などの、SSH プライベートキーのパスおよびファイル名を指定します。

## 次のステップ

- OpenShift Container Platform をインストールする際に、SSH パブリックキーをインストールプログラムに指定します。

### 1.1.8. インストールプログラムの取得

OpenShift Container Platform をインストールする前に、インストールファイルをローカルコンピューターにダウンロードします。

#### 前提条件

- 500 MB のローカルディスク領域がある Linux または macOS を実行するコンピューターが必要です。

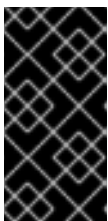
#### 手順

1. OpenShift Cluster Manager サイトの [インフラストラクチャプロバイダー](#) ページにアクセスします。Red Hat アカウントがある場合は、認証情報を使ってログインします。アカウントがない場合はこれを作成します。
2. インフラストラクチャプロバイダーを選択します。
3. 選択するインストールタイプのページに移動し、オペレーティングシステムのインストールプログラムをダウンロードし、ファイルをインストール設定ファイルを保存するディレクトリに配置します。



#### 重要

インストールプログラムは、クラスターのインストールに使用するコンピューターにいくつかのファイルを作成します。クラスターのインストール完了後は、インストールプログラムおよびインストールプログラムが作成するファイルを保持する必要があります。ファイルはいずれもクラスターを削除するために必要になります。



#### 重要

インストールプログラムで作成されたファイルを削除しても、クラスターがインストール時に失敗した場合でもクラスターは削除されません。クラスターを削除するには、特定のクラウドプロバイダー用の OpenShift Container Platform のアンインストール手順を実行します。

4. インストールプログラムを展開します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ tar xvf openshift-install-linux.tar.gz
```

5. [Red Hat OpenShift Cluster Manager](#) から [インストールプルシークレット](#) をダウンロードします。このプルシークレットを使用し、OpenShift Container Platform コンポーネントのコンテナイメージを提供する Quay.io など、組み込まれた各種の認証局によって提供されるサービスで認証できます。

### 1.1.9. クラスターのデプロイ

互換性のあるクラウドプラットフォームに OpenShift Container Platform をインストールできます。





## 重要

インストールプログラムの **create cluster** コマンドは、初期インストール時に1回だけ実行できます。

## 前提条件

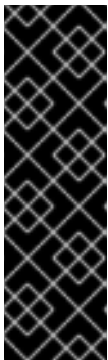
- インストーラーを実行するマシンから **ovirt-imageio** ポートを Manager へのポートを開放する。デフォルトでは、ポートは **54322** です。
- OpenShift Container Platform インストールプログラム、およびクラスターのプルシークレットを取得します。

## 手順

1. インストールプログラムが含まれるディレクトリーに切り替え、クラスターのデプロイメントを初期化します。

```
$ ./openshift-install create cluster --dir <installation_directory> \ 1
--log-level=info 2
```

- 1 **<installation\_directory>** の場合、インストールプログラムが作成するファイルを保存するためにディレクトリー名を指定します。
- 2 異なるインストールの詳細情報を表示するには、**info** ではなく、**warn**、**debug**、または **error** を指定します。



## 重要

空のディレクトリーを指定します。ブートストラップ X.509 証明書などの一部のインストールアセットの有効期限は短く設定されているため、インストールディレクトリーを再利用することができません。別のクラスターインストールの個別のファイルを再利用する必要がある場合は、それらをディレクトリーにコピーすることができます。ただし、インストールアセットのファイル名はリリース間で変更される可能性があります。インストールファイルを以前のバージョンの OpenShift Container Platform からコピーする場合は注意してコピーを行ってください。

インストールプログラムのプロンプトに対応します。

- a. オプション: **SSH Public Key** には、パスワードなしのパブリックキー (例: `~/.ssh/id_rsa.pub`) を選択します。このキーは、新規 OpenShift Container Platform クラスターとの接続を認証します。



## 注記

インストールのデバッグまたは障害復旧を実行する必要がある実稼働用の OpenShift Container Platform クラスターには、**ssh-agent** プロセスが使用する SSH キーを選択します。

- b. **Platform** には、**ovirt** を選択します。
- c. **Engine FQDN[:PORT]** に、RHV 環境の完全修飾ドメイン名 (FQDN) を入力します。

以下に例を示します。

```
rhv-env.virtlab.example.com:443
```

- d. インストーラーは CA 証明書を自動的に生成します。 **Would you like to use the above certificate to connect to the Manager?** では、 **y** または **N** のいずれかで回答します。 **N** と回答する場合は、 OpenShift Container Platform を非セキュアモードでインストールする必要があります。
- e. **Engine username** には、この形式を使用して RHV 管理者のユーザー名およびプロフィールを入力します。

```
<username>@<profile> 1
```

- 1 **<username>** に、RHV 管理者のユーザー名を指定します。 **<profile>** には、ログインプロフィールを指定します。ログインプロフィールは、RHV Administration Portal ログインページに移動し、 **Profile** ドロップダウンリストで確認できます。例:  
**admin@internal**

- f. **Engine password** に、RHV 管理者パスワードを入力します。
- g. **Cluster** には、OpenShift Container Platform をインストールするための RHV クラスタを選択します。
- h. **Storage domain** には、OpenShift Container Platform をインストールするためのストレージドメインを選択します。
- i. **Network** には、RHV Manager REST API へのアクセスのある仮想ネットワークを選択します。
- j. **Internal API Virtual IP** に、クラスタの REST API とは別の静的 IP アドレスを入力します。
- k. **Ingress virtual IP** に、ワイルドカードアプリドメイン用に予約した静的 IP アドレスを入力します。
- l. **Base Domain** に、OpenShift Container Platform クラスタのベースドメインを入力します。このクラスタが外部に公開される場合、これは DNS インフラストラクチャーが認識する有効なドメインである必要があります。たとえば、 **virtlab.example.com** を入力します。
- m. **Cluster Name** に、クラスタの名前を入力します。例: **my-cluster** OpenShift Container Platform REST API およびアプリケーションドメイン名向けに作成した外部登録/解決可能な DNS エントリーのクラスタ名を使用します。インストールプログラムは、この名前を RHV 環境のクラスタにも指定します。
- n. **Pull secret** には、先にダウンロードした **pull-secret.txt** ファイルからプルシークレットをコピーし、ここに貼り付けます。 [Red Hat OpenShift Cluster Manager から同じプルシークレット](#) のコピーを取得することもできます。



### 注記

ホストに設定した AWS アカウントにクラスタをデプロイするための十分なパーミッションがない場合、インストールプログラムは停止し、不足しているパーミッションが表示されます。

クラスターのデプロイメントが完了すると、Web コンソールへのリンクや **kubeadmin** ユーザーの認証情報を含む、クラスターにアクセスするための指示がターミナルに表示されます。

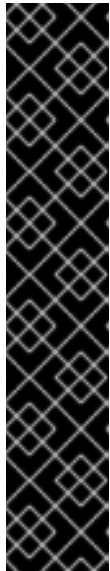
## 出力例

```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "4vYBz-Ee6gm-ymBZj-
Wt5AL"
INFO Time elapsed: 36m22s
```



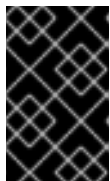
### 注記

クラスターアクセスおよび認証情報の情報は、インストールが正常に実行される際に `<installation_directory>/openshift_install.log` に出力されます。



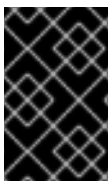
### 重要

- インストールプログラムが生成する Ignition 設定ファイルには、24 時間が経過すると期限切れになり、その後に更新される証明書が含まれます。証明書を更新する前にクラスターが停止し、24 時間経過した後にクラスターを再起動すると、クラスターは期限切れの証明書を自動的に復元します。例外として、kubelet 証明書を回復するために保留状態の **node-bootstrapper** 証明書署名要求 (CSR) を手動で承認する必要があります。詳細は、**コントロールプレーン証明書の期限切れの状態からのリカバリー** についてのドキュメントを参照してください。
- 24 時間証明書はクラスターのインストール後 16 時間から 22 時間にローテーションするため、Ignition 設定ファイルは、生成後 12 時間以内に使用することをお勧めします。12 時間以内に Ignition 設定ファイルを使用することにより、インストール中に証明書の更新が実行された場合のインストールの失敗を回避できます。



### 重要

インストールプログラム、またはインストールプログラムが作成するファイルを削除することはできません。これらはいずれもクラスターを削除するために必要になります。



### 重要

クラスターのインストールに必要な手順を完了している必要があります。残りの手順では、クラスターを検証し、インストールのトラブルシューティングを行う方法を説明します。

## 1.1.10. バイナリーのダウンロードによる OpenShift CLI のインストール

コマンドラインインターフェイスを使用して OpenShift Container Platform と対話するために CLI (**oc**) をインストールすることができます。**oc** は Linux、Windows、または macOS にインストールできます。



## 重要

以前のバージョンの **oc** をインストールしている場合、これを使用して OpenShift Container Platform 4.6 のすべてのコマンドを実行することはできません。新規バージョンの **oc** をダウンロードし、インストールします。

### 1.1.10.1. Linux への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを Linux にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 Linux Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。
4. アーカイブを展開します。

```
$ tar xvzf <file>
```

5. **oc** バイナリーを、**PATH** にあるディレクトリーに配置します。**PATH** を確認するには、以下のコマンドを実行します。

```
$ echo $PATH
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
$ oc <command>
```

### 1.1.10.2. Windows への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを Windows にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 Windows Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。
4. ZIP プログラムでアーカイブを解凍します。
5. **oc** バイナリーを、**PATH** にあるディレクトリーに移動します。**PATH** を確認するには、コマンドプロンプトを開いて以下のコマンドを実行します。

```
C:\> path
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
C:\> oc <command>
```

### 1.1.10.3. macOS への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを macOS にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 MacOSX Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。
4. アーカイブを展開し、解凍します。
5. **oc** バイナリーをパスにあるディレクトリーに移動します。  
**PATH** を確認するには、ターミナルを開き、以下のコマンドを実行します。

```
$ echo $PATH
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
$ oc <command>
```

詳細は、[Getting started with the OpenShift CLI](#) を参照してください。

### 1.1.11. CLI の使用によるクラスターへのログイン

クラスター **kubeconfig** ファイルをエクスポートし、デフォルトシステムユーザーとしてクラスターにログインできます。**kubeconfig** ファイルには、クライアントを正しいクラスターおよび API サーバーに接続するために CLI で使用されるクラスターについての情報が含まれます。このファイルはクラスターに固有のファイルであり、OpenShift Container Platform のインストール時に作成されます。

#### 前提条件

- OpenShift Container Platform クラスターをデプロイしていること。
- **oc** CLI をインストールしていること。

#### 手順

1. **kubeadmin** 認証情報をエクスポートします。

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

- 1** **<installation\_directory>** には、インストールファイルを保存したディレクトリーへのパスを指定します。

2. エクスポートされた設定を使用して、**oc** コマンドを正常に実行できることを確認します。

```
$ oc whoami
```

### 出力例

```
system:admin
```

### 関連情報

- OpenShift Container Platform へのアクセスおよびその詳細は、[Web コンソールへのアクセス](#) について参照してください。

### 1.1.12. クラスターステータスの確認

インストール時またはインストール後に OpenShift Container Platform クラスターのステータスを確認することができます。

#### 手順

1. クラスター環境で、管理者の kubeconfig ファイルをエクスポートします。

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

- 1** **<installation\_directory>** には、インストールファイルを保存したディレクトリーへのパスを指定します。

**kubeconfig** ファイルには、クライアントを正しいクラスターおよび API サーバーに接続するために CLI で使用されるクラスターについての情報が含まれます。

2. デプロイメント後に作成されたコントロールプレーンおよびコンピューターマシンを表示します。

```
$ oc get nodes
```

3. クラスターのバージョンを表示します。

```
$ oc get clusterversion
```

4. Operator のステータスを表示します。

```
$ oc get clusteroperator
```

5. クラスター内のすべての実行中の Pod を表示します。

```
$ oc get pods -A
```

### トラブルシューティング

インストールが失敗すると、インストールプログラムがタイムアウトし、エラーメッセージが表示されます。詳細は、[インストールに関する問題のトラブルシューティング](#) を参照してください。

### 1.1.13. RHV での OpenShift Container Platform Web コンソールへのアクセス

OpenShift Container Platform クラスターの初期化後に、OpenShift Container Platform Web コンソールにログインできます。

#### 手順

1. オプション: Red Hat Virtualization (RHV) Administration Portal で、**Compute** → **Cluster** を開きます。
2. インストールプログラムが仮想マシンを作成することを確認します。
3. インストールプログラムが実行されているコマンドラインに戻ります。インストールプログラムが完了すると、OpenShift Container Platform Web コンソールにログインするためのユーザー名およびパスワードの一時パスワードが表示されます。
4. ブラウザーから OpenShift Container Platform の Web コンソールの URL を開きます。URL は以下の形式を使用します。

```
console-openshift-console.apps.<clustername>.<basedomain> ①
```

① **<clustername>.<basedomain>** に、クラスター名およびベースドメインを指定します。

以下に例を示します。

```
console-openshift-console.apps.my-cluster.virtlab.example.com
```

### 1.1.14. OpenShift Container Platform の Telemetry アクセス

OpenShift Container Platform 4.6 では、クラスターのヘルスと更新の成功に関するメトリクスを提供するためにデフォルトで実行される Telemetry サービスには、インターネットアクセスが必要です。クラスターがインターネットに接続されている場合、Telemetry は自動的に実行され、クラスターは [OpenShift Cluster Manager](#) に登録されます。

[OpenShift Cluster Manager](#) インベントリが正常である (Telemetry によって自動的に維持、または OpenShift Cluster Manager を使用して手動で維持) ことを確認した後に、[subscription watch](#) を使用して、アカウントまたはマルチクラスターレベルで OpenShift Container Platform サブスクリプションを追跡します。

#### 関連情報

- Telemetry サービスの詳細は、[リモートヘルスマニターリング](#) について参照してください。

### 1.1.15. Red Hat Virtualization (RHV) へのインストールに関するよくある問題のトラブルシューティング

以下に、一般的な問題およびそれらについて考えられる原因および解決策を記載します。

#### 1.1.15.1. CPU 負荷が増大し、ノードが Not Ready 状態になる

- **現象:** CPU 負荷が大幅に増大し、ノードが **Not Ready** 状態に切り替わり始める。

- **原因:** 特にコントロールプレーンノード (別名マスターノード) の場合、ストレージドメインのレイテンシーが高すぎる可能性があります。
- **解決策:**  
Kubelet サービスを再起動して、ノードを再度 Ready 状態にします。

```
$ systemctl restart kubelet
```

OpenShift Container Platform メトリクスサービスを検査します。これは、etcd ディスクの同期期間などの有用なデータを収集し、これについて報告します。クラスターが機能している場合は、このデータを使用して、ストレージのレイテンシーまたはスループットが根本的な問題かどうかを判断します。その場合、レイテンシーが短く、スループットの高いストレージリソースの使用を検討してください。

未加工メトリクスを取得するには、kubeadmin または cluster-admin 権限を持つユーザーで以下のコマンドを実行します。

```
$ oc get --insecure-skip-tls-verify --server=https://localhost:<port> --raw=/metrics
```

詳細は、[Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x](#) を参照してください。

### 1.1.15.2. OpenShift Container Platform クラスター API に接続できない

- **現象:** インストールプログラムは完了するが、OpenShift Container Platform クラスター API は利用できない。ブートストラップの仮想マシンは、ブートストラッププロセスの完了後も起動した状態になります。以下のコマンドを入力すると、応答がタイムアウトします。

```
$ oc login -u kubeadmin -p *** <apiurl>
```

- **原因:** ブートストラップ仮想マシンがインストールプログラムによって削除されず、クラスターの API IP アドレスをリリースしない。
- **解決策:** **wait-for** サブコマンドを使用して、ブートストラッププロセスの完了時に通知を受信する。

```
$ ./openshift-install wait-for bootstrap-complete
```

ブートストラッププロセスが完了したら、ブートストラップ仮想マシンを削除します。

```
$ ./openshift-install destroy bootstrap
```

### 1.1.16. インストール後のタスク

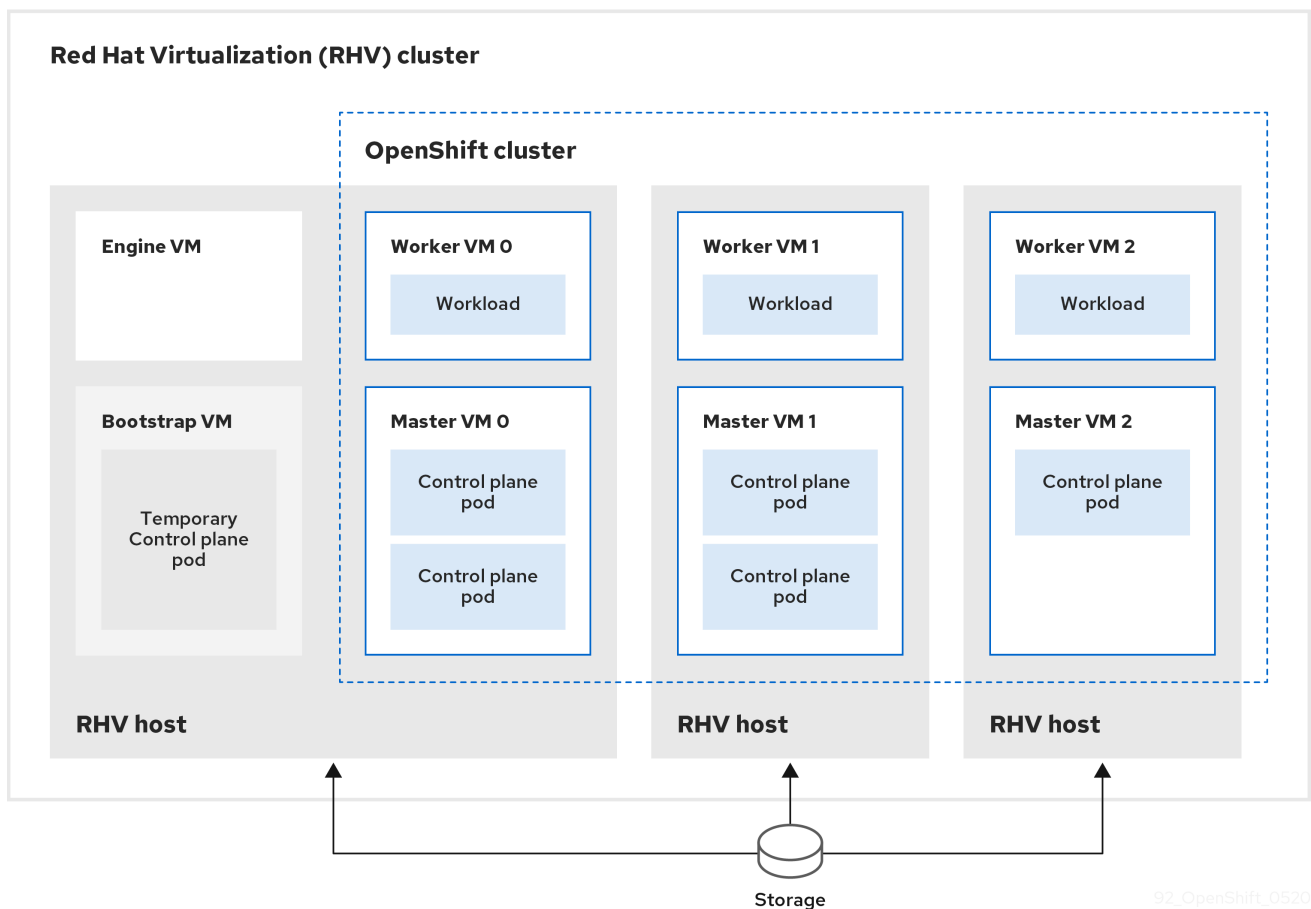
OpenShift Container Platform クラスターの初期化後に、以下のタスクを実行できます。

- オプション: デプロイメント後に、OpenShift Container Platform で Machine Config Operator (MCO) を使用して SSH キーを追加するか、または置き換えます。
- オプション: **kubeadmin** ユーザーを削除します。代わりに、認証プロバイダーを使用して cluster-admin 権限を持つユーザーを作成します。

## 1.2. カスタマイズによる RHV へのクラスターのインストール



以下の図に示されるように、OpenShift Container Platform クラスタを Red Hat Virtualization (RHV) でカスタマイズし、インストールすることができます。



92\_OpenShift\_0520

インストールプログラムは、インストーラーでプロビジョニングされるインフラストラクチャーを使用してクラスタの作成およびデプロイを自動化します。

カスタマイズされたクラスタをインストールするには、環境を準備し、以下の手順を実行します。

1. インストールプログラムを実行し、そのプロンプトに回答して、インストール設定ファイル **install-config.yaml** ファイルを作成します。
2. **install-config.yaml** ファイルでパラメーターを検査し、変更します。
3. **install-config.yaml** ファイルの作業用コピーを作成します。
4. **install-config.yaml** ファイルのコピーを使ってインストールプログラムを実行します。

次に、インストールプログラムは OpenShift Container Platform クラスタを作成します。

カスタマイズされたクラスタをインストールする代替方法については、[デフォルトのクラスタのインストール](#) を参照してください。



#### 注記

このインストールプログラムは、Linux および macOS でのみ利用できます。

### 1.2.1. 前提条件

- [OpenShift Container Platform のインストールおよび更新](#) プロセスについての詳細を確認します。
- [Support Matrix for OpenShift Container Platform on Red Hat Virtualization \(RHV\)](#) に記載のあるサポートされるバージョンの組み合わせを使用できる。
- ファイアウォールを使用する場合、クラスターがアクセスする必要のある [サイトを許可するようにファイアウォールを設定](#) します。

## 1.2.2. OpenShift Container Platform のインターネットアクセス

OpenShift Container Platform 4.6 では、クラスターをインストールするためにインターネットアクセスが必要になります。

インターネットへのアクセスは以下を実行するために必要です。

- [OpenShift Cluster Manager](#) にアクセスし、インストールプログラムをダウンロードし、サブスクリプション管理を実行します。クラスターにインターネットアクセスがあり、Telemetry を無効にしない場合、そのサービスは有効なサブスクリプションでクラスターを自動的に使用します。
- クラスターのインストールに必要なパッケージを取得するために [Quay.io](#) にアクセスします。
- クラスターの更新を実行するために必要なパッケージを取得します。



### 重要

クラスターでインターネットに直接アクセスできない場合、プロビジョニングする一部のタイプのインフラストラクチャーでネットワークが制限されたインストールを実行できます。このプロセスで、必要なコンテンツをダウンロードし、これを使用してミラーレジストリーにクラスターのインストールおよびインストールプログラムの生成に必要なパッケージを設定します。インストールタイプによっては、クラスターのインストール環境でインターネットアクセスが不要となる場合があります。クラスターを更新する前に、ミラーレジストリーのコンテンツを更新します。

## 1.2.3. RHV 環境の要件

OpenShift Container Platform クラスターをインストールし、実行するには、RHV 環境が以下の要件を満たしている必要があります。

これらの要件を満たさないと、インストールまたはプロセスが失敗する可能性があります。さらに、これらの要件を満たしていないと、OpenShift Container Platform クラスターはインストールしてから数日または数週間後に失敗する可能性があります。

CPU、メモリー、ストレージリソースについての以下の要件は、インストールプログラムが作成する仮想マシンのデフォルト数で乗算した **デフォルト** 値に基づいています。これらのリソースは、RHV 環境が OpenShift Container Platform 以外の操作に使用するものに **加え**、利用可能でなければなりません。

デフォルトでは、インストールプログラムは7つの仮想マシンをインストールプロセスで作成します。まず、ブートストラップ仮想マシンを作成し、OpenShift Container Platform クラスターの残りの部分を作成する間に一時サービスとコントロールプレーンを提供します。インストールプログラムがクラスターの作成を終了すると、ブートストラップマシンが削除され、そのリソースが解放されます。

RHV 環境の仮想マシン数を増やす場合は、リソースを適宜増やす必要があります。

### 要件

- RHV 環境に Up 状態のデータセンターが1つあること。
- RHV データセンターに RHV クラスターが含まれていること。
- RHV クラスターに OpenShift Container Platform クラスター専用の以下のリソースがあること。
  - 最小 28 vCPU: インストール時に作成される 7 仮想マシンのそれぞれに 4 vCPU。
  - 以下を含む 112 GiB 以上の RAM。
    - 一時的なコントロールプレーンを提供するブートストラップマシン用に 16 GiB 以上。
    - コントロールプレーンを提供する 3 つのコントロールプレーンマシンのそれぞれに 16 GiB 以上。
    - アプリケーションワークロードを実行する 3 つのコンピュータマシンのそれぞれに 16 GiB 以上。
- RHV ストレージドメインは、これらの [etcd バックエンドのパフォーマンス要件](#) を満たす必要があります。
- 実稼働環境では、各仮想マシンに 120 GiB 以上が必要です。そのため、ストレージドメインはデフォルトの OpenShift Container Platform クラスターに 840 GiB 以上を提供する必要があります。リソースに制約のある環境または非実稼働環境では、各仮想マシンに 32 GiB 以上を指定する必要があるため、ストレージドメインにはデフォルトの OpenShift Container Platform クラスター用に 230 GiB 以上が必要になります。
- インストールおよび更新中に Red Hat Ecosystem Catalog からイメージをダウンロードするには、RHV クラスターがインターネット接続にアクセスできる必要があります。また、サブスクリプションおよびエンタイトルメントプロセスを単純化するために Telemetry サービスにもインターネット接続が必要です。
- RHV クラスターには、RHV Manager の REST API にアクセスできる仮想ネットワークが必要です。インストーラーが作成する仮想マシンが DHCP を使用して IP アドレスを取得するため、DHCP がこのネットワークで有効にされていることを確認します。
- ターゲット RHV クラスターに OpenShift Container Platform クラスターをインストールし、管理するための以下の最小限の権限を持つユーザーアカウントおよびグループ。
  - **DiskOperator**
  - **DiskCreator**
  - **UserTemplateBasedVm**
  - **TemplateOwner**
  - **TemplateCreator**
  - ターゲットクラスターの **ClusterAdmin**



### 警告

最小権限の原則を適用します。インストールプロセスで RHV で **SuperUser** 権限を持つ管理者アカウントを使用することを避けます。インストールプログラムは、ユーザーが指定する認証情報を、危険にさらされる可能性のある一時的な **ovirt-config.yaml** ファイルに保存します。

## 1.2.4. RHV 環境の要件の確認

RHV 環境が OpenShift Container Platform クラスターをインストールし、実行するための要件を満たしていることを確認します。これらの要件を満たさないと、エラーが発生する可能性があります。



### 重要

これらの要件は、インストールプログラムがコントロールプレーンおよびコンピュータマシンの作成に使用するデフォルトのリソースに基づいています。これらのリソースには、vCPU、メモリー、およびストレージが含まれます。これらのリソースを変更するか、または OpenShift Container Platform マシンの数を増やす場合は、これらの要件を適宜調整します。

### 手順

1. RHV のバージョンを確認します。
  - a. RHV Administration Portal の右上にある ? ヘルプアイコンをクリックし、**About** を選択します。
  - b. 開かれるウィンドウで、**RHV ソフトウェアのバージョン** をメモします。
  - c. OpenShift Container Platform のバージョン 4.6 とメモした RHV のバージョンが、[Support Matrix for OpenShift Container Platform on RHV](#) でサポートされている組み合わせのいずれかであることを確認します。
2. データセンター、クラスター、およびストレージを検査します。
  - a. RHV 管理ポータルで、**Compute → Data Centers** をクリックします。
  - b. OpenShift Container Platform をインストールする予定のデータセンターにアクセスできることを確認します。
  - c. そのデータセンターの名前をクリックします。
  - d. データセンターの詳細の **Storage** タブで、OpenShift Container Platform をインストールする予定のストレージドメインが **Active** であることを確認します。
  - e. 後で使用できるように **ドメイン名** を記録します。
  - f. **空き領域** に 230 GiB 以上あることを確認します。
  - g. ストレージドメインが [これらの etcd バックエンドのパフォーマンス要件](#) を満たしていることを確認します。これは、[fio パフォーマンスベンチマークツール](#) を使用して測定できません。

- h. データセンターの詳細で、**Clusters** タブをクリックします。
  - i. OpenShift Container Platform をインストールする予定の RHV クラスタを見つけます。後で使用できるようにクラスタ名を記録します。
3. RHV ホストリソースを確認します。
    - a. RHV 管理ポータルで、**Compute > Clusters** をクリックします。
    - b. OpenShift Container Platform をインストールする予定のクラスタをクリックします。
    - c. クラスタの詳細で、**Hosts** タブをクリックします。
    - d. ホストを検査し、それらに OpenShift Container Platform クラスタ **専用** として利用可能な **論理 CPU コア** の合計が 28 つ以上であることを確認します。
    - e. 後で使用できるように、利用可能な **論理 CPU コア** の数を記録します。
    - f. これらの CPU コアが分散され、インストール時に作成された 7 つの仮想マシンのそれぞれに 4 つのコアを持たせることができることを確認します。
    - g. ホストには、以下の OpenShift Container Platform マシンのそれぞれの要件を満たすように **新規仮想マシンをスケジュールするための最大空きメモリー** として 112 GiB があることを確認します。
      - ブートストラップマシンに 16 GiB が必要です。
      - 3 つのコントロールプレーンマシンのそれぞれに 16 GiB が必要です。
      - 3 つのコンピュートマシンのそれぞれに 16 GiB が必要です。
    - h. 後で使用できるように **新規仮想マシンをスケジュールするための最大空きメモリー** の量を記録します。
  4. OpenShift Container Platform をインストールするための仮想ネットワークが RHV Manager の REST API にアクセスできることを確認します。このネットワーク上の仮想マシンから、RHV Manager の REST API に到達するために curl を使用します。

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

1 **<username>** については、RHV で OpenShift Container Platform クラスタを作成および管理する権限を持つ RHV アカウントのユーザー名を指定します。**<profile>** には、ログインプロファイルを指定します。ログインプロファイルは、RHV Administration Portal ログインページに移動し、**Profile** ドロップダウンリストで確認できます。**<password>** に、そのユーザー名のパスワードを指定します。

2 **<engine-fqdn>** に、RHV 環境の完全修飾ドメイン名を指定します。

以下に例を示します。

```
$ curl -k -u ocpadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

### 1.2.5. RHV でのネットワーク環境の準備

OpenShift Container Platform クラスターの 2 つの静的 IP アドレスを設定し、これらのアドレスを使用して DNS エントリーを作成します。

## 手順

1. 2 つの静的 IP アドレスを予約します。
  - a. OpenShift Container Platform をインストールするネットワークで、DHCP リースプール外にある 2 つの静的 IP アドレスを特定します。
  - b. このネットワーク上のホストに接続し、それぞれの IP アドレスが使用されていないことを確認します。たとえば、Address Resolution Protocol (ARP) を使用して、IP アドレスのいずれにもエントリーがないことを確認します。

```
$ arp 10.35.1.19
```

## 出力例

```
10.35.1.19 (10.35.1.19) -- no entry
```

- c. ネットワーク環境の標準的な方法に従って、2 つの静的 IP アドレスを予約します。
  - d. 今後の参照用にこれらの IP アドレスを記録します。
2. 以下の形式を使用して、OpenShift Container Platform REST API およびアプリケーションドメイン名の DNS エントリーを作成します。

```
api.<cluster-name>.<base-domain> <ip-address> ①  
*.apps.<cluster-name>.<base-domain> <ip-address> ②
```

- ① **<cluster-name>**、**<base-domain>**、および **<ip-address>** には、クラスター名、ベースドメイン、および OpenShift Container Platform API の静的 IP アドレスを指定します。
- ② Ingress およびロードバランサー用に OpenShift Container Platform アプリケーションのクラスター名、ベースドメイン、および静的 IP アドレスを指定します。

以下に例を示します。

```
api.my-cluster.virtlab.example.com 10.35.1.19  
*.apps.my-cluster.virtlab.example.com 10.35.1.20
```

## 1.2.6. RHV 用の CA 証明書の設定

Red Hat Virtualization (RHV) Manager から CA 証明書をダウンロードし、インストールマシンにこれを設定します。

RHV Manager からの Web サイトまたは **curl** コマンドを使用して、証明書をダウンロードできます。

その後、インストールプログラムに証明書を提供します。

## 手順

1. 以下の 2 つの方法のいずれかを使用して CA 証明書をダウンロードします。

- Manager の Web ページ (<https://<engine-fqdn>/ovirt-engine/>) に移動します。次に、**Downloads** で **CA Certificate** のリンクをクリックします。
- 以下のコマンドを実行します。

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem 1
```

- 1 **<engine-fqdn>** には、RHV Manager の完全修飾ドメイン名 (例: **rhv-env.virtlab.example.com**) を指定します。

2. ルートレスユーザーに Manager へのアクセスを付与するように CA ファイルを設定します。CA ファイルのパーミッションを 8 進数の **0644** に設定します (シンボリック値: **-rw-r--r--**):

```
$ sudo chmod 0644 /tmp/ca.pem
```

3. Linux の場合は、サーバー証明書のディレクトリーに CA 証明書をコピーします。-p を使用してパーミッションを保存します。

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

4. オペレーティングシステム用の証明書マネージャーに証明書を追加します。

- MacOS の場合は、証明書ファイルをダブルクリックして、**Keychain Access** ユーティリティを使用してファイルを **System** キーチェーンに追加します。
- Linux の場合は、CA 信頼を更新します。

```
$ sudo update-ca-trust
```



### 注記

独自の認証局を使用する場合は、システムがこれを信頼することを確認します。

## 関連情報

- 詳細は、RHV ドキュメントの [Authentication and Security](#) を参照してください。

### 1.2.7. SSH プライベートキーの生成およびエージェントへの追加

クラスターでインストールのデバッグまたは障害復旧を実行する必要がある場合、**ssh-agent** とインストールプログラムの両方に SSH キーを指定する必要があります。このキーを使用してパブリッククラスターのブートストラップマシンにアクセスし、インストールの問題をトラブルシューティングできます。



### 注記

実稼働環境では、障害復旧およびデバッグが必要です。

このキーを使用して、ユーザー **core** としてマスターノードに対して SSH を実行できます。クラスターをデプロイする際に、キーは **core** ユーザーの **~/.ssh/authorized\_keys** 一覧に追加されます。

## 手順

1. パスワードなしの認証に設定されている SSH キーがコンピューター上にない場合は、これを作成します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ ssh-keygen -t ed25519 -N "" \
-f <path>/<file_name> ①
```

- ① `~/.ssh/id_rsa` などの、新規 SSH キーのパスおよびファイル名を指定します。既存のキーペアがある場合は、公開鍵が `~/.ssh` ディレクトリーにあることを確認します。

このコマンドを実行すると、指定した場所にパスワードを必要としない SSH キーが生成されます。



### 注記

FIPS で検証済み/進行中のモジュール (Modules in Process) 暗号ライブラリーを使用する OpenShift Container Platform クラスタを **x86\_64** アーキテクチャーにインストールする予定の場合は、**ed25519** アルゴリズムを使用するキーは作成しないでください。代わりに、**rsa** アルゴリズムまたは **ecdsa** アルゴリズムを使用するキーを作成します。

2. **ssh-agent** プロセスをバックグラウンドタスクとして開始します。

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

### 出力例

```
Agent pid 31874
```



### 注記

クラスタが FIPS モードにある場合は、FIPS 準拠のアルゴリズムのみを使用して SSH キーを生成します。鍵は RSA または ECDSA のいずれかである必要があります。

3. SSH プライベートキーを **ssh-agent** に追加します。

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

### 出力例

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

- ① `~/.ssh/id_rsa` などの、SSH プライベートキーのパスおよびファイル名を指定します。

## 次のステップ



- OpenShift Container Platform をインストールする際に、SSH パブリックキーをインストールプログラムに指定します。

## 1.2.8. インストールプログラムの取得

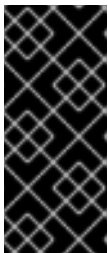
OpenShift Container Platform をインストールする前に、インストールファイルをローカルコンピューターにダウンロードします。

### 前提条件

- 500 MB のローカルディスク領域がある Linux または macOS を実行するコンピューターが必要です。

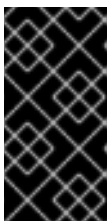
### 手順

1. OpenShift Cluster Manager サイトの [インフラストラクチャプロバイダー](#) ページにアクセスします。Red Hat アカウントがある場合は、認証情報を使ってログインします。アカウントがない場合はこれを作成します。
2. インフラストラクチャプロバイダーを選択します。
3. 選択するインストールタイプのページに移動し、オペレーティングシステムのインストールプログラムをダウンロードし、ファイルをインストール設定ファイルを保存するディレクトリに配置します。



#### 重要

インストールプログラムは、クラスターのインストールに使用するコンピューターにいくつかのファイルを作成します。クラスターのインストール完了後は、インストールプログラムおよびインストールプログラムが作成するファイルを保持する必要があります。ファイルはいずれもクラスターを削除するために必要になります。



#### 重要

インストールプログラムで作成されたファイルを削除しても、クラスターがインストール時に失敗した場合でもクラスターは削除されません。クラスターを削除するには、特定のクラウドプロバイダー用の OpenShift Container Platform のアンインストール手順を実行します。

4. インストールプログラムを展開します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ tar xvf openshift-install-linux.tar.gz
```

5. [Red Hat OpenShift Cluster Manager](#) から [インストールプルシークレット](#) をダウンロードします。このプルシークレットを使用し、OpenShift Container Platform コンポーネントのコンテナイメージを提供する Quay.io など、組み込まれた各種の認証局によって提供されるサービスで認証できます。

## 1.2.9. インストール設定ファイルの作成

Red Hat Virtualization (RHV) にインストールする OpenShift Container Platform クラスターをカスタマイズできます。

## 前提条件

- OpenShift Container Platform インストールプログラム、およびクラスターのプルシークレットを取得します。

## 手順

### 1. `install-config.yaml` ファイルを作成します。

- インストールプログラムが含まれるディレクトリーに切り替え、以下のコマンドを実行します。

```
$ ./openshift-install create install-config --dir <installation_directory> ❶
```

- ❶ `<installation_directory>` の場合、インストールプログラムが作成するファイルを保存するためにディレクトリー名を指定します。



### 重要

空のディレクトリーを指定します。ブートストラップ X.509 証明書などの一部のインストールアセットの有効期限は短く設定されているため、インストールディレクトリーを再利用することができません。別のクラスターインストールの個別のファイルを再利用する必要がある場合は、それらをディレクトリーにコピーすることができます。ただし、インストールアセットのファイル名はリリース間で変更される可能性があります。インストールファイルを以前のバージョンの OpenShift Container Platform からコピーする場合は注意してコピーを行ってください。

- インストールプログラムのプロンプトに対応します。
  - SSH Public Key** では、パスワードなしのパブリックキー (例: `~/ssh/id_rsa.pub`) を選択します。このキーは、新規 OpenShift Container Platform クラスターとの接続を認証します。



### 注記

インストールのデバッグまたは障害復旧を実行する必要がある実稼働用の OpenShift Container Platform クラスターには、**ssh-agent** プロセスが使用する SSH キーを選択します。

- Platform** には、**ovirt** を選択します。
- Enter oVirt's API endpoint URL** に、この形式を使用して RHV API の URL を入力します。

```
https://<engine-fqdn>/ovirt-engine/api ❶
```

- ❶ `<engine-fqdn>` に、RHV 環境の完全修飾ドメイン名を指定します。

以下に例を示します。

```
$ curl -k -u ocpadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

- iv. **Is the oVirt CA trusted?** には、CA 証明書がすでに設定されているため **Yes** を入力します。そうでない場合は、**No** と入力します。
- v. **oVirt's CA bundle** には、前の質問で **Yes** を入力している場合には、`/etc/pki/ca-trust/source/anchors/ca.pem` の内容をコピーし、ここに貼り付けます。その後、**Enter** を 2 回押します。そうでない場合、つまり、前の質問で **No** と入力している場合は、この質問は表示されません。
- vi. **oVirt engine username** には、この形式を使用して RHV 管理者のユーザー名およびプロファイルを入力します。

```
<username>@<profile> ①
```

- ① **<username>** に、RHV 管理者のユーザー名を指定します。**<profile>** には、ログインプロファイルを指定します。ログインプロファイルは、RHV Administration Portal ログインページに移動し、**Profile** ドロップダウンリストで確認できます。ユーザー名とプロファイルは以下のようになります。

```
ocpadmin@internal
```

- vii. **oVirt engine password** に、RHV 管理者パスワードを入力します。
- viii. **oVirt cluster** には、OpenShift Container Platform をインストールするためのクラスターを選択します。
- ix. **oVirt storage domain** には、OpenShift Container Platform をインストールするためのストレージドメインを選択します。
- x. **oVirt network** には、Manager REST API にアクセスできる仮想ネットワークを選択します。
- xi. **Internal API Virtual IP** に、クラスターの REST API とは別の静的 IP アドレスを入力します。
- xii. **Ingress virtual IP** に、ワイルドカードアプリドメイン用に予約した静的 IP アドレスを入力します。
- xiii. **Base Domain** に、OpenShift Container Platform クラスターのベースドメインを入力します。このクラスターが外部に公開される場合、これは DNS インフラストラクチャーが認識する有効なドメインである必要があります。たとえば、**virtlab.example.com** を入力します。
- xiv. **Cluster Name** に、クラスターの名前を入力します。例: **my-cluster** OpenShift Container Platform REST API およびアプリケーションドメイン名向けに作成した外部登録/解決可能な DNS エントリーのクラスター名を使用します。インストールプログラムは、この名前を RHV 環境のクラスターにも指定します。
- xv. **Pull secret** には、先にダウンロードした **pull-secret.txt** ファイルからプルシークレットをコピーし、ここに貼り付けます。[Red Hat OpenShift Cluster Manager](#) から [同じプルシークレット](#) のコピーを取得することもできます。

2. **install-config.yaml** ファイルを変更します。利用可能なパラメーターの詳細については、**インストール設定パラメーター**セクションを参照してください。
3. **install-config.yaml** ファイルをバックアップし、複数のクラスターをインストールするのに使用できるようにします。



### 重要

**install-config.yaml** ファイルはインストールプロセス時に使用されます。このファイルを再利用する必要がある場合は、この段階でこれをバックアップしてください。

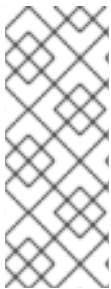
#### 1.2.9.1. Red Hat Virtualization (RHV) のサンプル **install-config.yaml** ファイル

**install-config.yaml** ファイルのパラメーターおよびパラメーター値を変更して、インストールプログラムが作成する OpenShift Container Platform クラスターをカスタマイズできます。

以下の例は、RHV への OpenShift Container Platform のインストールに固有の例です。

このファイルは、以下のコマンドを実行する際に指定する **<installation\_directory>** にあります。

```
$ ./openshift-install create install-config --dir <installation_directory>
```



### 注記

- これらのサンプルファイルは参照用のみ提供されます。インストールプログラムを使用して **install-config.yaml** ファイルを取得する必要があります。
- **install-config.yaml** ファイルを変更すると、クラスターに必要なリソースを増やすことができます。RHV 環境にそれらの追加リソースがあることを確認します。これらが無い場合は、インストールまたはクラスターが失敗します。

例: これはデフォルトの **install-config.yaml** ファイルです。

```
apiVersion: v1
baseDomain: example.com
compute:
- architecture: amd64
  hyperthreading: Enabled
  name: worker
  platform: {}
  replicas: 3
controlPlane:
  architecture: amd64
  hyperthreading: Enabled
  name: master
  platform: {}
  replicas: 3
metadata:
  creationTimestamp: null
  name: my-cluster
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
```

```

hostPrefix: 23
machineNetwork:
- cidr: 10.0.0.0/16
networkType: OpenShiftSDN
serviceNetwork:
- 172.30.0.0/16
platform:
ovirt:
  api_vip: 10.46.8.230
  ingress_vip: 192.168.1.5
  ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
  ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
  ovirt_network_name: ovirtmgmt
  vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
publish: External
pullSecret: '{"auths": ...}'
sshKey: ssh-ed12345 AAAA...

```

#### 例: 最小の install-config.yaml ファイル

```

apiVersion: v1
baseDomain: example.com
metadata:
  name: test-cluster
platform:
ovirt:
  api_vip: 10.46.8.230
  ingress_vip: 10.46.8.232
  ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
  ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
  ovirt_network_name: ovirtmgmt
  vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
pullSecret: '{"auths": ...}'
sshKey: ssh-ed12345 AAAA...

```

#### 例: install-config.yaml ファイルのカスタムマシンプール

```

apiVersion: v1
baseDomain: example.com
controlPlane:
  name: master
platform:
ovirt:
  cpu:
    cores: 4
    sockets: 2
  memoryMB: 65536
  osDisk:
    sizeGB: 100
  vmType: server
replicas: 3
compute:
- name: worker
  platform:
    ovirt:

```

```

cpu:
  cores: 4
  sockets: 4
memoryMB: 65536
osDisk:
  sizeGB: 200
vmType: server
replicas: 5
metadata:
  name: test-cluster
platform:
  ovirt:
    api_vip: 10.46.8.230
    ingress_vip: 10.46.8.232
    ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
    ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
    ovirt_network_name: ovirtmgmt
    vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
  pullSecret: '{"auths": ...}'
  sshKey: ssh-ed25519 AAAA...

```

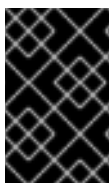
### 1.2.9.2. インストール設定パラメーター

OpenShift Container Platform クラスターをデプロイする前に、クラスターをホストするクラウドプラットフォームでアカウントを記述し、クラスターのプラットフォームをオプションでカスタマイズするためにパラメーターの値を指定します。**install-config.yaml** インストール設定ファイルを作成する際に、コマンドラインで必要なパラメーターの値を指定します。クラスターをカスタマイズする場合、**install-config.yaml** ファイルを変更して、プラットフォームについての詳細情報を指定できます。



#### 注記

インストール後は、これらのパラメーターを **install-config.yaml** ファイルで変更することはできません。



#### 重要

**openshift-install** コマンドは、パラメーターのフィールド名を検証しません。正しくない名前を指定すると、関連するファイルまたはオブジェクトは作成されず、エラーが報告されません。指定されたパラメーターのフィールド名が正しいことを確認します。

#### 1.2.9.2.1. 必須設定パラメーター

必須のインストール設定パラメーターは、以下の表で説明されています。

表1.1 必須パラメーター

パラメーター	説明	値
<b>apiVersion</b>	<b>install-config.yaml</b> コンテンツの API バージョン。現在のバージョンは <b>v1</b> です。インストーラーは、古い API バージョンをサポートすることもできます。	文字列

パラメーター	説明	値
--------	----	---

<b>baseDomain</b>	クラウドプロバイダーのベースドメイン。ベースドメインは、OpenShift Container Platform クラスターコンポーネントへのルートを作成するために使用されます。クラスターの完全な DNS 名は、 <b>baseDomain</b> と <b>&lt;metadata.name&gt;</b> 。 <b>&lt;baseDomain&gt;</b> 形式を使用する <b>metadata.name</b> パラメーターの値の組み合わせです。	<b>example.com</b> などの完全修飾ドメインまたはサブドメイン名。
<b>metadata</b>	Kubernetes リソース <b>ObjectMeta</b> 。ここからは <b>name</b> パラメーターのみが消費されます。	オブジェクト
<b>metadata.name</b>	クラスターの名前。クラスターの DNS レコードはすべて <b>{{.metadata.name}}</b> 。 <b>{{.baseDomain}}</b> のサブドメインです。	<b>dev</b> などの小文字、ハイフン (-)、およびピリオド (.) が含まれる文字列。
<b>platform</b>	インストールの実行に使用する特定プラットフォームの設定: <b>aws</b> 、 <b>baremetal</b> 、 <b>azure</b> 、 <b>openstack</b> 、 <b>ovirt</b> 、 <b>vsphere</b> 。 <b>platform.&lt;platform&gt;</b> パラメーターに関する追加情報は、以下の表で特定のプラットフォームについて参照してください。	オブジェクト

パラメーター	説明	値
<b>pullSecret</b>	Red Hat OpenShift Cluster Manager からプルシークレットを取得して、Quay.io などのサービスから OpenShift Container Platform コンポーネントのコンテナイメージをダウンロードすることを認証します。	<pre>{   "auths":{     "cloud.openshift.com":{       "auth":"b3Blb=",       "email":"you@example.com"     },     "quay.io":{       "auth":"b3Blb=",       "email":"you@example.com"     }   } }</pre>

### 1.2.9.2.2. ネットワーク設定パラメーター

既存のネットワークインフラストラクチャーの要件に基づいて、インストール設定をカスタマイズできます。たとえば、クラスターネットワークの IP アドレスブロックを拡張するか、デフォルトとは異なる IP アドレスブロックを指定できます。

IPv4 アドレスのみがサポートされます。

表1.2 ネットワークパラメーター

パラメーター	説明	値
<b>networking</b>	クラスターのネットワークの設定。	オブジェクト   <b>注記</b> インストール後に <b>networking</b> オブジェクトで指定したパラメーターを変更することはできません。
<b>networking.networkType</b>	インストールするクラスターネットワークプロバイダー Container Network Interface (CNI) プラグイン。	<b>OpenShiftSDN</b> または <b>OVNKubernetes</b> のいずれか。デフォルト値は <b>OpenShiftSDN</b> です。
<b>networking.clusterNetwork</b>	Pod の IP アドレスブロック。  デフォルト値は <b>10.128.0.0/14</b> で、ホストの接頭辞は <b>/23</b> です。  複数の IP アドレスブロックを指定する場合は、ブロックが重複しないようにしてください。	オブジェクトの配列。以下に例を示します。  <pre>networking:   clusterNetwork:     - cidr: 10.128.0.0/14       hostPrefix: 23</pre>




パラメーター	説明	値
<b>networking.clusterNetwork.cidr</b>	<b>networking.clusterNetwork</b> を使用する場合に必須です。IP アドレスブロック。  IPv4 ネットワーク	CIDR (Classless Inter-Domain Routing) 表記の IP アドレスブロック。IPv4 ブロックの接頭辞長は <b>0</b> から <b>32</b> の間になります。
<b>networking.clusterNetwork.hostPrefix</b>	それぞれの個別ノードに割り当てるサブネット接頭辞長。たとえば、 <b>hostPrefix</b> が <b>23</b> に設定される場合、各ノードに指定の <b>cidr</b> から <b>/23</b> サブネットが割り当てられます。 <b>hostPrefix</b> 値の <b>23</b> は、 $2^{(32-23)-2}$ Pod IP アドレスを提供します。	サブネット接頭辞。  デフォルト値は <b>23</b> です。
<b>networking.serviceNetwork</b>	サービスの IP アドレスブロック。デフォルト値は <b>172.30.0.0/16</b> です。  OpenShift SDN および OVN-Kubernetes ネットワークプロバイダーは、サービスネットワークの単一 IP アドレスブロックのみをサポートします。	CIDR 形式の IP アドレスブロックを持つ配列。以下に例を示します。  <pre>networking:   serviceNetwork:     - 172.30.0.0/16</pre>
<b>networking.machineNetwork</b>	マシンの IP アドレスブロック。  複数の IP アドレスブロックを指定する場合は、ブロックが重複しないようにしてください。	オブジェクトの配列。以下に例を示します。  <pre>networking:   machineNetwork:     - cidr: 10.0.0.0/16</pre>
<b>networking.machineNetwork.cidr</b>	<b>networking.machineNetwork</b> を使用する場合に必須です。IP アドレスブロック。libvirt 以外のすべてのプラットフォームでは、デフォルト値は <b>10.0.0.0/16</b> です。libvirt の場合、デフォルト値は <b>192.168.126.0/24</b> です。	CIDR 表記の IP ネットワークブロック。  例: <b>10.0.0.0/16</b>   <b>注記</b> 優先される NIC が置かれている CIDR に一致する <b>networking.machineNetwork</b> を設定します。




### 1.2.9.2.3. オプションの設定パラメーター

オプションのインストール設定パラメーターは、以下の表で説明されています。

表1.3 オプションのパラメーター

パラメーター	説明	値
<b>additionalTrustBundle</b>	ノードの信頼済み証明書ストアに追加される PEM でエンコードされた X.509 証明書バンドル。この信頼バンドルは、プロキシが設定される際にも使用できます。	文字列
<b>compute</b>	コンピュータノードを設定するマシンの設定。	machine-pool オブジェクトの配列。詳細は、以下の Machine-pool の表を参照してください。
<b>compute.architecture</b>	プール内のマシンの命令セットアーキテクチャーを決定します。現時点で異種クラスターはサポートされていないため、すべてのプールが同じアーキテクチャーを指定する必要があります。有効な値は <b>amd64</b> (デフォルト) です。	文字列
<b>compute.hyperthreading</b>	<p>コンピュータマシンで同時マルチスレッドまたは <b>hyperthreading</b> を有効/無効にするかどうか。デフォルトでは、同時スレッドはマシンのコアのパフォーマンスを上げるために有効にされます。</p> <div style="display: flex; align-items: center;">  <div> <p><b>重要</b></p> <p>同時スレッドを無効にする場合は、容量計画においてマシンパフォーマンスの大幅な低下が考慮に入れていることを確認します。</p> </div> </div>	<b>Enabled</b> または <b>Disabled</b>
<b>compute.name</b>	<b>compute</b> を使用する場合に必須です。マシンプールの名前。	<b>worker</b>
<b>compute.platform</b>	<b>compute</b> を使用する場合に必須です。このパラメーターを使用して、ワーカーマシンをホストするクラウドプロバイダーを指定します。このパラメーターの値は <b>controlPlane.platform</b> パラメーターの値に一致する必要があります。	<b>aws</b> 、 <b>azure</b> 、 <b>gcp</b> 、 <b>openstack</b> 、 <b>ovirt</b> 、 <b>vsphere</b> 、または <b>{}</b>

パラメーター	説明	値
<b>compute.replicas</b>	プロビジョニングするコンピュータマシン (ワーカーマシンとしても知られる) の数。	<b>2</b> 以上の正の整数。デフォルト値は <b>3</b> です。
<b>controlPlane</b>	コントロールプレーンを設定するマシンの設定。	<b>MachinePool</b> オブジェクトの配列。詳細は、以下の Machine-pool の表を参照してください。
<b>controlPlane.architecture</b>	プール内のマシンの命令セットアーキテクチャーを決定します。現時点で異種クラスターはサポートされていないため、すべてのプールが同じアーキテクチャーを指定する必要があります。有効な値は <b>amd64</b> (デフォルト) です。	文字列
<b>controlPlane.hyperthreading</b>	コントロールプレーンマシンで同時マルチスレッドまたは <b>hyperthreading</b> を有効/無効にするかどうか。デフォルトでは、同時スレッドはマシンのコアのパフォーマンスを上げるために有効にされます。  <div data-bbox="493 1088 600 1373" data-label="Image"> </div> <b>重要</b> 同時スレッドを無効にする場合は、容量計画においてマシンパフォーマンスの大幅な低下が考慮に入れていることを確認します。	<b>Enabled</b> または <b>Disabled</b>
<b>controlPlane.name</b>	<b>controlPlane</b> を使用する場合に必須です。マシンプールの名前。	<b>master</b>
<b>controlPlane.platform</b>	<b>controlPlane</b> を使用する場合に必須です。このパラメーターを使用して、コントロールプレーンマシンをホストするクラウドプロバイダーを指定します。このパラメーターの値は <b>compute.platform</b> パラメーターの値に一致する必要があります。	<b>aws</b> 、 <b>azure</b> 、 <b>gcp</b> 、 <b>openstack</b> 、 <b>ovirt</b> 、 <b>vsphere</b> 、または <b>{}</b>
<b>controlPlane.replicas</b>	プロビジョニングするコントロールプレーンマシンの数。	サポートされる値は <b>3</b> のみです (これはデフォルト値です)。

パラメーター	説明	値
<b>credentialsMode</b>	<p>Cloud Credential Operator (CCO) モード。モードを指定しないと、CCO は指定された認証情報の機能を動的に判別しようとします。この場合、複数のモードがサポートされるプラットフォームで Mint モードが優先されます。</p>  <p><b>注記</b></p> <p>すべてのクラウドプロバイダーですべての CCO モードがサポートされているわけではありません。CCO モードの詳細は、<b>Red Hat Operator</b>の<b>クラウド認証情報 Operator</b>を参照してください。</p>	<b>Mint、Passthrough、Manual</b> 、または空の文字列 ("")。
<b>fips</b>	<p>FIPS モードを有効または無効にします。デフォルトは <b>false</b> (無効) です。FIPS モードが有効にされている場合、OpenShift Container Platform が実行される Red Hat Enterprise Linux CoreOS (RHCOS) マシンがデフォルトの Kubernetes 暗号スイートをバイパスし、代わりに RHCOS で提供される暗号モジュールを使用します。</p>  <p><b>重要</b></p> <p>FIPS 検証済み/進行中のモジュール (Modules in Process) 暗号ライブラリーの使用は、<b>x86_64</b> アーキテクチャーの OpenShift Container Platform デプロイメントでのみサポートされています。</p>  <p><b>注記</b></p> <p>Azure File ストレージを使用している場合、FIPS モードを有効にすることはできません。</p>	<b>false</b> または <b>true</b>

パラメーター	説明	値
<b>imageContentSources</b>	release-image コンテンツのソースおよびリポジトリ。	オブジェクトの配列。この表の以下の行で説明されているように、 <b>source</b> およびオプションで <b>mirrors</b> が含まれます。
<b>imageContentSources.source</b>	<b>imageContentSources</b> を使用する場合に必須です。ユーザーが参照するリポジトリを指定します (例: イメージプル仕様)。	文字列
<b>imageContentSources.mirrors</b>	同じイメージが含まれる可能性のあるリポジトリを1つ以上指定します。	文字列の配列。
<b>publish</b>	Kubernetes API、OpenShift ルートなどのクラスターのユーザーに表示されるエンドポイントをパブリッシュまたは公開する方法。	<p><b>Internal</b> または <b>External</b>。デフォルト値は <b>External</b> です。</p> <p>このパラメーターを <b>Internal</b> に設定することは、クラウド以外のプラットフォームではサポートされません。</p> <div style="display: flex; align-items: flex-start;"> <div style="background-color: #ccc; width: 20px; height: 100px; margin-right: 10px;"></div> <div> <p><b>重要</b></p> <p>フィールドの値が <b>Internal</b> に設定されている場合、クラスターは機能しなくなります。詳細は、<a href="#">BZ#1953035</a> を参照してください。</p> </div> </div>
<b>sshKey</b>	<p>クラスターマシンへのアクセスを認証するための単一または複数の SSH キー。</p> <div style="display: flex; align-items: flex-start;"> <div style="background-color: #ccc; width: 20px; height: 100px; margin-right: 10px;"></div> <div> <p><b>注記</b></p> <p>インストールのデバッグまたは障害復旧を実行する必要がある実稼働用の OpenShift Container Platform クラスターでは、<b>ssh-agent</b> プロセスが使用する SSH キーを指定します。</p> </div> </div>	<p>1つ以上のキー。以下に例を示します。</p> <pre>sshKey:   &lt;key1&gt;   &lt;key2&gt;   &lt;key3&gt;</pre>

#### 1.2.9.2.4. 追加の Red Hat Virtualization (RHV) 設定パラメーター

追加の RHV 設定パラメーターは以下の表で説明されています。

表1.4 クラスターの追加 RHV パラメーター

パラメーター	説明	値
<code>platform.ovirt.ovirt_cluster_id</code>	必須。仮想マシンが作成されるクラスター。	文字列。例: <b>68833f9f-e89c-4891-b768-e2ba0815b76b</b>
<code>platform.ovirt.ovirt_storage_domain_id</code>	必須。仮想マシンディスクが作成されるストレージドメイン ID。	文字列。例: <b>ed7b0f4e-0e96-492a-8fff-279213ee1468</b>
<code>platform.ovirt.ovirt_network_name</code>	必須。仮想マシン NIC が作成されるネットワーク名。	文字列。例: <b>ocpcluster</b>
<code>platform.ovirt.vnicProfileID</code>	必須。仮想マシンネットワークインターフェイスの vNIC プロファイル ID。これは、クラスターネットワークに単一のプロファイルがある場合に示唆されます。	文字列。例: <b>3fa86930-0be5-4052-b667-b79f0a729692</b>
<code>platform.ovirt.api_vip</code>	必須。API 仮想 IP (VIP) に割り当てられるマシンネットワークの IP アドレス。このエンドポイントで OpenShift API にアクセスできます。	文字列。例: <b>10.46.8.230</b>
<code>platform.ovirt.ingress_vip</code>	必須。Ingress 仮想 IP (VIP) に割り当てられるマシンネットワークの IP アドレス。	文字列。例: <b>10.46.8.232</b>

#### 1.2.9.2.5. マシンプールの追加 RHV パラメーター

マシンプールの追加の RHV 設定パラメーターは以下の表で説明されています。

表1.5 マシンプールの追加 RHV パラメーター

パラメーター	説明	値
<code>&lt;machine-pool&gt;.platform.ovirt.cpu</code>	オプション。仮想マシンの CPU を定義します。	オブジェクト
<code>&lt;machine-pool&gt;.platform.ovirt.cpu.cores</code>	<code>&lt;machine-pool&gt;.platform.ovirt.cpu</code> を使用する場合に必須です。コア数。仮想 CPU (vCPU) の合計はコア * ソケットです。	整数
<code>&lt;machine-pool&gt;.platform.ovirt.cpu.sockets</code>	<code>&lt;machine-pool&gt;.platform.ovirt.cpu</code> を使用する場合に必須です。コアあたりのソケット数。仮想 CPU (vCPU) の合計はコア * ソケットです。	整数

パラメーター	説明	値
<code>&lt;machine-pool&gt;.platform.ovirt.memoryMB</code>	オプション。仮想マシンのメモリー (MiB 単位)。	整数
<code>&lt;machine-pool&gt;.platform.ovirt.instanceTypeID</code>	オプション。00000009-0009-0009-0009-0000000000f1 などのインスタンスタイプ UUID。これは <a href="https://&lt;engine-fqdn&gt;/ovirt-engine/api/instancetypes">https://&lt;engine-fqdn&gt;/ovirt-engine/api/instancetypes</a> エンドポイントから取得できます。	UUID の文字列
<code>&lt;machine-pool&gt;.platform.ovirt.osDisk</code>	オプション。仮想マシンの起動可能な初回の、および起動可能なディスクを定義します。	文字列
<code>&lt;machine-pool&gt;.platform.ovirt.osDisk.sizeGB</code>	<code>&lt;machine-pool&gt;.platform.ovirt.osDisk</code> を使用する場合に必須です。ディスクのサイズ (GiB 単位)。	数字
<code>&lt;machine-pool&gt;.platform.ovirt.vmType</code>	オプション。high-performance、server、または desktop などの仮想マシンワークロードタイプ。	文字列

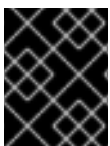


### 注記

`<machine-pool>` を `controlPlane` または `compute` に置き換えることができます。

## 1.2.10. クラスターのデプロイ

互換性のあるクラウドプラットフォームに OpenShift Container Platform をインストールできます。



### 重要

インストールプログラムの `create cluster` コマンドは、初期インストール時に 1 回だけ実行できます。

### 前提条件

- インストーラーを実行するマシンから `ovirt-imageio` ポートを Manager へのポートを開放する。デフォルトでは、ポートは `54322` です。
- OpenShift Container Platform インストールプログラム、およびクラスターのプルシークレットを取得します。

### 手順

1. インストールプログラムが含まれるディレクトリーに切り替え、クラスターのデプロイメントを初期化します。

```
$ ./openshift-install create cluster --dir <installation_directory> \ ❶
--log-level=info ❷
```

- ❶ <installation\_directory> については、カスタマイズした `./install-config.yaml` ファイルの場所を指定します。
- ❷ 異なるインストールの詳細情報を表示するには、`info` ではなく、`warn`、`debug`、または `error` を指定します。



### 注記

ホストに設定した AWS アカウントにクラスターをデプロイするための十分なパーミッションがない場合、インストールプログラムは停止し、不足しているパーミッションが表示されます。

クラスターのデプロイメントが完了すると、Web コンソールへのリンクや `kubeadmin` ユーザーの認証情報を含む、クラスターにアクセスするための指示がターミナルに表示されます。

### 出力例

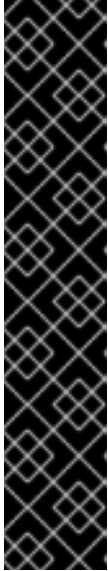
```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "4vYBz-Ee6gm-ymBZj-
Wt5AL"
INFO Time elapsed: 36m22s
```



### 注記

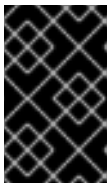
クラスターアクセスおよび認証情報の情報は、インストールが正常に実行される際に `<installation_directory>/openshift_install.log` に出力されます。





## 重要

- インストールプログラムが生成する Ignition 設定ファイルには、24 時間が経過すると期限切れになり、その後に更新される証明書が含まれます。証明書を更新する前にクラスターが停止し、24 時間経過した後にクラスターを再起動すると、クラスターは期限切れの証明書を自動的に復元します。例外として、kubelet 証明書を回復するために保留状態の **node-bootstrapper** 証明書署名要求 (CSR) を手動で承認する必要があります。詳細は、**コントロールプレーン証明書の期限切れの状態からのリカバリー** についてのドキュメントを参照してください。
- 24 時間証明書はクラスターのインストール後 16 時間から 22 時間にローテーションするため、Ignition 設定ファイルは、生成後 12 時間以内に使用することをお勧めします。12 時間以内に Ignition 設定ファイルを使用することにより、インストール中に証明書の更新が実行された場合のインストールの失敗を回避できます。



## 重要

インストールプログラム、またはインストールプログラムが作成するファイルを削除することはできません。これらはいずれもクラスターを削除するために必要になります。



## 重要

クラスターのインストールに必要な手順を完了している必要があります。残りの手順では、クラスターを検証し、インストールのトラブルシューティングを行う方法を説明します。

### 1.2.11. バイナリーのダウンロードによる OpenShift CLI のインストール

コマンドラインインターフェイスを使用して OpenShift Container Platform と対話するために CLI (**oc**) をインストールすることができます。**oc** は Linux、Windows、または macOS にインストールできます。



## 重要

以前のバージョンの **oc** をインストールしている場合、これを使用して OpenShift Container Platform 4.6 のすべてのコマンドを実行することはできません。新規バージョンの **oc** をダウンロードし、インストールします。

#### 1.2.11.1. Linux への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを Linux にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 Linux Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。

4. アーカイブを展開します。

```
$ tar xvzf <file>
```

5. **oc** バイナリーを、**PATH** にあるディレクトリーに配置します。  
**PATH** を確認するには、以下のコマンドを実行します。

```
$ echo $PATH
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
$ oc <command>
```

### 1.2.11.2. Windows への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを Windows にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 Windows Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。
4. ZIP プログラムでアーカイブを解凍します。
5. **oc** バイナリーを、**PATH** にあるディレクトリーに移動します。  
**PATH** を確認するには、コマンドプロンプトを開いて以下のコマンドを実行します。

```
C:\> path
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
C:\> oc <command>
```

### 1.2.11.3. macOS への OpenShift CLI のインストール

以下の手順を使用して、OpenShift CLI (**oc**) バイナリーを macOS にインストールできます。

#### 手順

1. Red Hat カスタマーポータルでの [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Version** ドロップダウンメニューで適切なバージョンを選択します。
3. **OpenShift v4.6 MacOSX Client** エントリーの横にある **Download Now** をクリックして、ファイルを保存します。
4. アーカイブを展開し、解凍します。

5. **oc** バイナリーをパスにあるディレクトリーに移動します。  
**PATH**を確認するには、ターミナルを開き、以下のコマンドを実行します。

```
$ echo $PATH
```

OpenShift CLI のインストール後に、**oc** コマンドを使用して利用できます。

```
$ oc <command>
```

### 1.2.12. CLI の使用によるクラスターへのログイン

クラスター **kubeconfig** ファイルをエクスポートし、デフォルトシステムユーザーとしてクラスターにログインできます。**kubeconfig** ファイルには、クライアントを正しいクラスターおよび API サーバーに接続するために CLI で使用されるクラスターについての情報が含まれます。このファイルはクラスターに固有のファイルであり、OpenShift Container Platform のインストール時に作成されます。

#### 前提条件

- OpenShift Container Platform クラスターをデプロイしていること。
- **oc** CLI をインストールしていること。

#### 手順

1. **kubeadmin** 認証情報をエクスポートします。

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

- 1 **<installation\_directory>** には、インストールファイルを保存したディレクトリーへのパスを指定します。

2. エクスポートされた設定を使用して、**oc** コマンドを正常に実行できることを確認します。

```
$ oc whoami
```

#### 出力例

```
system:admin
```

詳細は、[Getting started with the OpenShift CLI](#) を参照してください。

### 1.2.13. クラスターステータスの確認

インストール時またはインストール後に OpenShift Container Platform クラスターのステータスを確認することができます。

#### 手順

1. クラスター環境で、管理者の **kubeconfig** ファイルをエクスポートします。

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

- 1 **<installation\_directory>** には、インストールファイルを保存したディレクトリーへのパスを指定します。

**kubeconfig** ファイルには、クライアントを正しいクラスターおよび API サーバーに接続するために CLI で使用されるクラスターについての情報が含まれます。

2. デプロイメント後に作成されたコントロールプレーンおよびコンピューターマシンを表示します。

```
$ oc get nodes
```

3. クラスターのバージョンを表示します。

```
$ oc get clusterversion
```

4. Operator のステータスを表示します。

```
$ oc get clusteroperator
```

5. クラスター内のすべての実行中の Pod を表示します。

```
$ oc get pods -A
```

## トラブルシューティング

インストールが失敗すると、インストールプログラムがタイムアウトし、エラーメッセージが表示されます。詳細は、[インストールに関する問題のトラブルシューティング](#) を参照してください。

### 1.2.14. RHV での OpenShift Container Platform Web コンソールへのアクセス

OpenShift Container Platform クラスターの初期化後に、OpenShift Container Platform Web コンソールにログインできます。

#### 手順

1. オプション: Red Hat Virtualization (RHV) Administration Portal で、**Compute** → **Cluster** を開きます。
2. インストールプログラムが仮想マシンを作成することを確認します。
3. インストールプログラムが実行されているコマンドラインに戻ります。インストールプログラムが完了すると、OpenShift Container Platform Web コンソールにログインするためのユーザー名およびパスワードの一時パスワードが表示されます。
4. ブラウザーから OpenShift Container Platform の Web コンソールの URL を開きます。URL は以下の形式を使用します。

```
console-openshift-console.apps.<clustername>.<basedomain> 1
```

- 1 **<clustername>.<basedomain>** に、クラスター名およびベースドメインを指定します。

以下に例を示します。

■

```
console-openshift-console.apps.my-cluster.virtlab.example.com
```

### 1.2.15. OpenShift Container Platform の Telemetry アクセス

OpenShift Container Platform 4.6 では、クラスターのヘルスと更新の成功に関するメトリクスを提供するためにデフォルトで実行される Telemetry サービスには、インターネットアクセスが必要です。クラスターがインターネットに接続されている場合、Telemetry は自動的に実行され、クラスターは [OpenShift Cluster Manager](#) に登録されます。

[OpenShift Cluster Manager](#) インベントリが正常である (Telemetry によって自動的に維持、または OpenShift Cluster Manager を使用して手動で維持) ことを確認した後に、[subscription watch](#) を使用して、アカウントまたはマルチクラスターレベルで OpenShift Container Platform サブスクリプションを追跡します。

#### 関連情報

- Telemetry サービスの詳細は、[リモートヘルスマニターリング](#) について参照してください。

### 1.2.16. Red Hat Virtualization (RHV) へのインストールに関するよくある問題のトラブルシューティング

以下に、一般的な問題およびそれらについて考えられる原因および解決策を記載します。

#### 1.2.16.1. CPU 負荷が増大し、ノードが Not Ready 状態になる

- **現象:** CPU 負荷が大幅に増大し、ノードが **Not Ready** 状態に切り替わり始める。
- **原因:** 特にコントロールプレーンノード (別名マスターノード) の場合、ストレージドメインのレイテンシーが高すぎる可能性があります。
- **解決策:**  
Kubelet サービスを再起動して、ノードを再度 Ready 状態にします。

```
$ systemctl restart kubelet
```

OpenShift Container Platform メトリクスサービスを検査します。これは、etcd ディスクの同期期間などの有用なデータを収集し、これについて報告します。クラスターが機能している場合は、このデータを使用して、ストレージのレイテンシーまたはスループットが根本的な問題かどうかを判断します。その場合、レイテンシーが短く、スループットの高いストレージリソースの使用を検討してください。

未加工メトリクスを取得するには、kubeadmin または cluster-admin 権限を持つユーザーで以下のコマンドを実行します。

```
$ oc get --insecure-skip-tls-verify --server=https://localhost:<port> --raw=/metrics
```

詳細は、[Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x](#) を参照してください。

#### 1.2.16.2. OpenShift Container Platform クラスター API に接続できない

- **現象:** インストールプログラムは完了するが、OpenShift Container Platform クラスター API は利用できない。ブートストラップの仮想マシンは、ブートストラッププロセスの完了後も起動した状態になります。以下のコマンドを入力すると、応答がタイムアウトします。

```
$ oc login -u kubeadmin -p *** <apiurl>
```

- **原因:** ブートストラップ仮想マシンがインストールプログラムによって削除されず、クラスターの API IP アドレスをリリースしない。
- **解決策:** **wait-for** サブコマンドを使用して、ブートストラッププロセスの完了時に通知を受信する。

```
$ ./openshift-install wait-for bootstrap-complete
```

ブートストラッププロセスが完了したら、ブートストラップ仮想マシンを削除します。

```
$ ./openshift-install destroy bootstrap
```

### 1.2.17. インストール後のタスク

OpenShift Container Platform クラスターの初期化後に、以下のタスクを実行できます。

- オプション: デプロイメント後に、OpenShift Container Platform で Machine Config Operator (MCO) を使用して SSH キーを追加するか、または置き換えます。
- オプション: **kubeadmin** ユーザーを削除します。代わりに、認証プロバイダーを使用して cluster-admin 権限を持つユーザーを作成します。

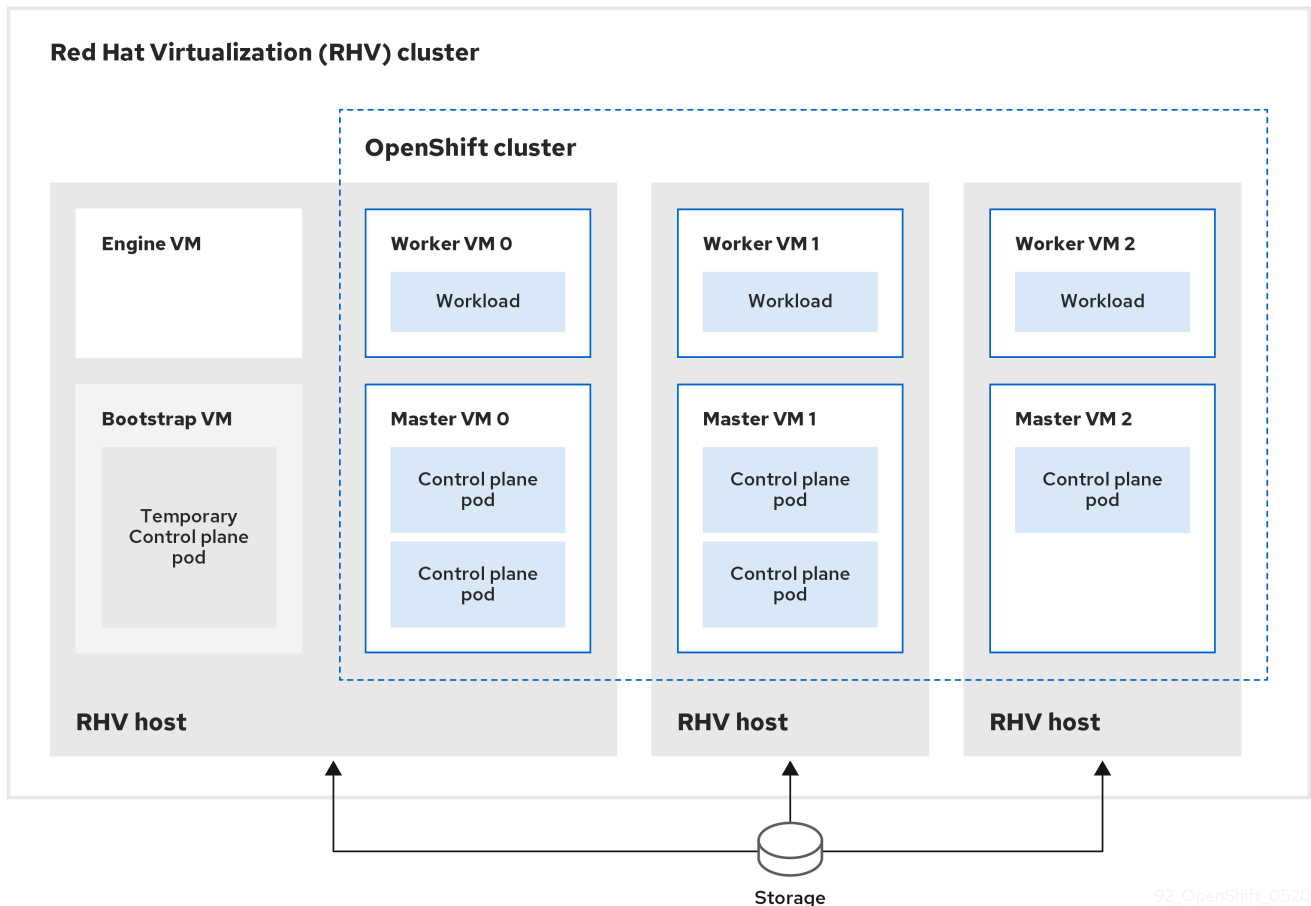
### 1.2.18. 次のステップ

- [クラスターをカスタマイズ](#) します。
- 必要な場合は、[リモートの健全性レポートをオプトアウト](#) することができます。

## 1.3. ユーザーによってプロビジョニングされるインフラストラクチャーを使用した RHV へのクラスターのインストール

OpenShift Container Platform バージョン 4.6 では、Red Hat Virtualization (RHV) および提供する他のインフラストラクチャーにカスタマイズされた OpenShift Container Platform クラスターをインストールできます。OpenShift Container Platform ドキュメントでは、**ユーザーによってプロビジョニングされるインフラストラクチャー** という用語を使用して、このインフラストラクチャータイプに言及しています。

以下の図は、RHV クラスターで実行される可能性のある OpenShift Container Platform クラスターの例を示しています。



92\_OpenShift\_0520

RHV ホストは、コントロールプレーンとコンピュート Pod の両方が含まれる仮想マシンを実行します。ホストのいずれかが Manage 仮想マシンと、一時的なコントロールプレーン Pod を含むブートストラップ仮想マシンも実行します。

### 1.3.1. 前提条件

OpenShift Container Platform クラスタを RHV 環境にインストールするには、以下の要件を満たしている必要があります。

- [Support Matrix for OpenShift Container Platform on Red Hat Virtualization \(RHV\)](#) に記載のあるサポートされるバージョンの組み合わせを使用できる。
- [OpenShift Container Platform のインストールおよび更新](#) プロセスについて理解している。

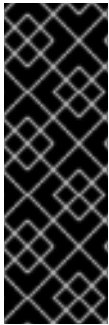
### 1.3.2. OpenShift Container Platform のインターネットアクセス

OpenShift Container Platform 4.6 では、クラスタをインストールするためにインターネットアクセスが必要になります。

インターネットへのアクセスは以下を実行するために必要です。

- [OpenShift Cluster Manager](#) にアクセスし、インストールプログラムをダウンロードし、サブスクリプション管理を実行します。クラスタにインターネットアクセスがあり、Telemetry を無効にしない場合、そのサービスは有効なサブスクリプションでクラスタを自動的に使用します。
- クラスタのインストールに必要なパッケージを取得するために [Quay.io](#) にアクセスします。

- クラスターの更新を実行するために必要なパッケージを取得します。



### 重要

クラスターでインターネットに直接アクセスできない場合、プロビジョニングする一部のタイプのインフラストラクチャーでネットワークが制限されたインストールを実行できます。このプロセスで、必要なコンテンツをダウンロードし、これを使用してミラーレジストリーにクラスターのインストールおよびインストールプログラムの生成に必要なパッケージを設定します。インストールタイプによっては、クラスターのインストール環境でインターネットアクセスが不要となる場合があります。クラスターを更新する前に、ミラーレジストリーのコンテンツを更新します。

### 1.3.3. RHV 環境の要件

OpenShift Container Platform クラスターをインストールし、実行するには、RHV 環境が以下の要件を満たしている必要があります。

これらの要件を満たさないと、インストールまたはプロセスが失敗する可能性があります。さらに、これらの要件を満たしていないと、OpenShift Container Platform クラスターはインストールしてから数日または数週間後に失敗する可能性があります。

CPU、メモリー、ストレージリソースについての以下の要件は、インストールプログラムが作成する仮想マシンのデフォルト数で乗算した **デフォルト** 値に基づいています。これらのリソースは、RHV 環境が OpenShift Container Platform 以外の操作に使用するものに **加え**、利用可能でなければなりません。

デフォルトでは、インストールプログラムは7つの仮想マシンをインストールプロセスで作成します。まず、ブートストラップ仮想マシンを作成し、OpenShift Container Platform クラスターの残りの部分を作成する間に一時サービスとコントロールプレーンを提供します。インストールプログラムがクラスターの作成を終了すると、ブートストラップマシンが削除され、そのリソースが解放されます。

RHV 環境の仮想マシン数を増やす場合は、リソースを適宜増やす必要があります。

#### 要件

- RHV 環境に **Up** 状態のデータセンターが1つあること。
- RHV データセンターに RHV クラスターが含まれていること。
- RHV クラスターに OpenShift Container Platform クラスター専用の以下のリソースがあること。
  - 最小 28 vCPU: インストール時に作成される 7 仮想マシンのそれぞれに 4 vCPU。
  - 以下を含む 112 GiB 以上の RAM。
    - 一時的なコントロールプレーンを提供するブートストラップマシン用に 16 GiB 以上。
    - コントロールプレーンを提供する 3 つのコントロールプレーンマシンのそれぞれに 16 GiB 以上。
    - アプリケーションワークロードを実行する 3 つのコンピュートマシンのそれぞれに 16 GiB 以上。
- RHV ストレージドメインは、[これらの etcd バックエンドのパフォーマンス要件](#) を満たす必要があります。



- 実稼働環境では、各仮想マシンに 120 GiB 以上が必要です。そのため、ストレージドメインはデフォルトの OpenShift Container Platform クラスターに 840 GiB 以上を提供する必要があります。リソースに制約のある環境または非実稼働環境では、各仮想マシンに 32 GiB 以上を指定する必要があるため、ストレージドメインにはデフォルトの OpenShift Container Platform クラスター用に 230 GiB 以上が必要になります。
- インストールおよび更新中に Red Hat Ecosystem Catalog からイメージをダウンロードするには、RHV クラスターがインターネット接続にアクセスできる必要があります。また、サブスクリプションおよびエンタイトルメントプロセスを単純化するために Telemetry サービスにもインターネット接続が必要です。
- RHV クラスターには、RHV Manager の REST API にアクセスできる仮想ネットワークが必要です。インストーラーが作成する仮想マシンが DHCP を使用して IP アドレスを取得するため、DHCP がこのネットワークで有効にされていることを確認します。
- ターゲット RHV クラスターに OpenShift Container Platform クラスターをインストールし、管理するための以下の最小限の権限を持つユーザーアカウントおよびグループ。
  - **DiskOperator**
  - **DiskCreator**
  - **UserTemplateBasedVm**
  - **TemplateOwner**
  - **TemplateCreator**
  - ターゲットクラスターの **ClusterAdmin**

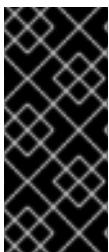


### 警告

最小権限の原則を適用します。インストールプロセスで RHV で **SuperUser** 権限を持つ管理者アカウントを使用することを避けます。インストールプログラムは、ユーザーが指定する認証情報を、危険にさらされる可能性のある一時的な **ovirt-config.yaml** ファイルに保存します。

### 1.3.4. RHV 環境の要件の確認

RHV 環境が OpenShift Container Platform クラスターをインストールし、実行するための要件を満たしていることを確認します。これらの要件を満たさないと、エラーが発生する可能性があります。



### 重要

これらの要件は、インストールプログラムがコントロールプレーンおよびコンピューターマシンの作成に使用するデフォルトのリソースに基づいています。これらのリソースには、vCPU、メモリー、およびストレージが含まれます。これらのリソースを変更するか、または OpenShift Container Platform マシンの数を増やす場合は、これらの要件を適宜調整します。

### 手順

1. RHV のバージョンを確認します。
  - a. RHV Administration Portal の右上にある ? ヘルプアイコンをクリックし、**About** を選択します。
  - b. 開かれるウィンドウで、**RHV ソフトウェアのバージョン** をメモします。
  - c. OpenShift Container Platform のバージョン 4.6 とメモした RHV のバージョンが、[Support Matrix for OpenShift Container Platform on RHV](#) でサポートされている組み合わせのいずれかであることを確認します。
2. データセンター、クラスター、およびストレージを検査します。
  - a. RHV 管理ポータルで、**Compute → Data Centers** をクリックします。
  - b. OpenShift Container Platform をインストールする予定のデータセンターにアクセスできることを確認します。
  - c. そのデータセンターの名前をクリックします。
  - d. データセンターの詳細の **Storage** タブで、OpenShift Container Platform をインストールする予定のストレージドメインが **Active** であることを確認します。
  - e. 後で使用できるように **ドメイン名** を記録します。
  - f. **空き領域** に 230 GiB 以上あることを確認します。
  - g. ストレージドメインが [これらの etcd バックエンドのパフォーマンス要件](#) を満たしていることを確認します。これは、[fio パフォーマンスベンチマークツール](#) を使用して測定できません。
  - h. データセンターの詳細で、**Clusters** タブをクリックします。
    - i. OpenShift Container Platform をインストールする予定の RHV クラスターを見つけます。後で使用できるようにクラスター名を記録します。
3. RHV ホストリソースを確認します。
  - a. RHV 管理ポータルで、**Compute > Clusters** をクリックします。
  - b. OpenShift Container Platform をインストールする予定のクラスターをクリックします。
  - c. クラスターの詳細で、**Hosts** タブをクリックします。
  - d. ホストを検査し、それらに OpenShift Container Platform クラスター **専用** として利用可能な **論理 CPU コア** の合計が 28 つ以上であることを確認します。
  - e. 後で使用できるように、利用可能な **論理 CPU コア** の数を記録します。
  - f. これらの CPU コアが分散され、インストール時に作成された 7 つの仮想マシンのそれぞれに 4 つのコアを持たせることができることを確認します。
  - g. ホストには、以下の OpenShift Container Platform マシンのそれぞれの要件を満たすように **新規仮想マシンをスケジュールするための最大空きメモリー** として 112 GiB があることを確認します。
    - ブートストラップマシンに 16 GiB が必要です。

- 3つのコントロールプレーンマシンのそれぞれに 16 GiB が必要です。
  - 3つのコンピュートマシンのそれぞれに 16 GiB が必要です。
- h. 後で使用できるように **新規仮想マシンをスケジュールするための最大空きメモリー** の量を記録します。
4. OpenShift Container Platform をインストールするための仮想ネットワークが RHV Manager の REST API にアクセスできることを確認します。このネットワーク上の仮想マシンから、RHV Manager の REST API に到達するために curl を使用します。

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

1 **<username>** については、RHV で OpenShift Container Platform クラスターを作成および管理する権限を持つ RHV アカウントのユーザー名を指定します。**<profile>** には、ログインプロファイルを指定します。ログインプロファイルは、RHV Administration Portal ログインページに移動し、**Profile** ドロップダウンリストで確認できます。**<password>** に、そのユーザー名のパスワードを指定します。

2 **<engine-fqdn>** に、RHV 環境の完全修飾ドメイン名を指定します。

以下に例を示します。

```
$ curl -k -u ocpadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

### 1.3.5. ユーザーによってプロビジョニングされるインフラストラクチャーのネットワーク要件

すべての Red Hat Enterprise Linux CoreOS (RHCOS) マシンでは、起動時に **initramfs** のネットワークがマシン設定サーバーから Ignition 設定をフェッチする必要があります。

初回の起動時に、Ignition 設定ファイルをダウンロードできるようネットワーク接続を確立するために、マシンには DHCP サーバーまたはその静的 IP アドレスが設定されている必要があります。

クラスターのマシンを長期間管理するために DHCP サーバーを使用することが推奨されています。DHCP サーバーが永続 IP アドレスおよびホスト名をクラスターマシンに提供するように設定されていることを確認します。

Kubernetes API サーバーはクラスターマシンのノード名を解決する必要があります。API サーバーおよびワーカーノードが異なるゾーンに置かれている場合、デフォルトの DNS 検索ゾーンを、API サーバーでノード名を解決できるように設定することができます。もう1つの実行可能な方法として、ノードオブジェクトとすべての DNS 要求の両方において、ホストを完全修飾ドメイン名で常に参照することができます。

マシン間のネットワーク接続を、クラスターのコンポーネントが通信できるように設定する必要があります。すべてのマシンではクラスターの他のすべてのマシンのホスト名を解決する必要があります。

#### ファイアウォール

クラスターが必要なサイトにアクセスできるようにファイアウォールを設定します。

以下も参照してください。

- [Red Hat Virtualization Manager ファイアウォールの要件](#)
- [ホストのファイアウォール要件](#)

## ロードバランサー

レイヤー 4 のロードバランサーを1つまたは2つ (推奨) 設定します。

- コントロールプレーンおよびブートストラップマシンのポート **6443** および **22623** に対して負荷分散を行います。ポート **6443** は Kubernetes API サーバーへのアクセスを提供し、内外で到達可能である必要があります。ポート **22623** はクラスター内のノードからアクセスできる必要があります。
- Ingress ルーターを実行するマシン (通常はデフォルト設定のコンピュートノード) 向けに、ポート **443** および **80** に対する負荷分散を行います。いずれのポートもクラスター内外でアクセスできる必要があります。

## DNS

インフラストラクチャーで提供される DNS を設定して、主要なコンポーネントとサービスの正しい解決を許可します。1つのロードバランサーのみを使用する場合、これらの DNS レコードは同じ IP アドレスを参照できます。

- `api.<cluster_name>.<base_domain>` (内部および外部解決) と、コントロールプレーンマシンのロードバランサーを参照する `api-int.<cluster_name>.<base_domain>` (内部解決) の DNS レコードを作成します。
- Ingress ルーターのロードバランサーを参照する `*.apps.<cluster_name>.<base_domain>` の DNS レコードを作成します。たとえば、コンピュートマシンのポート **443** および **80** などが含まれます。

表1.6 すべてのマシンに対応するすべてのマシン

プロトコル	ポート	説明
ICMP	該当なし	ネットワーク到達性のテスト
TCP	<b>1936</b>	メトリクス
	<b>9000-9999</b>	ホストレベルのサービス。ポート <b>9100-9101</b> のノードエクスポーター、ポート <b>9099</b> の Cluster Version Operator が含まれます。
	<b>10250-10259</b>	Kubernetes が予約するデフォルトポート
	<b>10256</b>	openshift-sdn
UDP	<b>4789</b>	VXLAN および Geneve
	<b>6081</b>	VXLAN および Geneve
	<b>9000-9999</b>	ポート <b>9100-9101</b> のノードエクスポーターを含む、ホストレベルのサービス。

プロトコル	ポート	説明
TCP/UDP	<b>30000-32767</b>	Kubernetes ノードポート

表1.7 コントロールプレーンへのすべてのマシン

プロトコル	ポート	説明
TCP	<b>6443</b>	Kubernetes API

表1.8 コントロールプレーンマシンへのコントロールプレーンマシン

プロトコル	ポート	説明
TCP	<b>2379-2380</b>	etcd サーバーおよびピアポート

### ネットワークポロジリー要件

クラスター用にプロビジョニングするインフラストラクチャーは、ネットワークポロジリーの以下の要件を満たす必要があります。



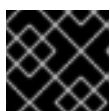
#### 重要

OpenShift Container Platform では、すべてのノードが、プラットフォームコンテナのイメージをプルし、Telemetry データを Red Hat に提供するためにインターネットへの直接のアクセスが必要です。

### ロードバランサー

OpenShift Container Platform をインストールする前に、以下の要件を満たす 2 つのロードバランサーをプロビジョニングする必要があります。

1. **API ロードバランサー:** プラットフォームと対話およびプラットフォームを設定するためのユーザー向けの共通のエンドポイントを提供します。以下の条件を設定します。
  - Layer 4 の負荷分散のみ。これは、Raw TCP、SSL パススルー、または SSL ブリッジモードと呼ばれます。SSL ブリッジモードを使用する場合は、API ルートの Server Name Indication (SNI) を有効にする必要があります。
  - ステートレス負荷分散アルゴリズム。オプションは、ロードバランサーの実装によって異なります。



#### 重要

API ロードバランサーのセッションの永続性は設定しないでください。

ロードバランサーのフロントとバックの両方で以下のポートを設定します。

表1.9 API ロードバランサー

ポート	バックエンドマシン (プールメンバー)	内部	外部	説明
6443	ブートストラップおよびコントロールプレーン。ブートストラップマシンがクラスターのコントロールプレーンを初期化した後に、ブートストラップマシンをロードバランサーから削除します。API サーバーのヘルスチェックプローブの <b>/readyz</b> エンドポイントを設定する必要があります。	X	X	Kubernetes API サーバー
22623	ブートストラップおよびコントロールプレーン。ブートストラップマシンがクラスターのコントロールプレーンを初期化した後に、ブートストラップマシンをロードバランサーから削除します。	X		マシン設定サーバー



### 注記

ロードバランサーは、API サーバーが **/readyz** エンドポイントをオフにしてからプールから API サーバーインスタンスを削除するまで最大 30 秒かかるように設定する必要があります。**/readyz** の後の時間枠内でエラーが返されたり、正常になったりする場合は、エンドポイントが削除または追加されているはずですが、5 秒または 10 秒ごとにプローブし、2 つの正常な要求が正常な状態になり、3 つの要求が正常な状態になりません。これらは十分にテストされた値です。

## 2. Application Ingress ロードバランサー: クラスター外から送られるアプリケーショントラフィックの Ingress ポイントを提供します。以下の条件を設定します。

- Layer 4 の負荷分散のみ。これは、Raw TCP、SSL パススルー、または SSL ブリッジモードと呼ばれます。SSL ブリッジモードを使用する場合は、Ingress ルートの Server Name Indication (SNI) を有効にする必要があります。
- 選択可能なオプションやプラットフォーム上でホストされるアプリケーションの種類に基づいて、接続ベースの永続化またはセッションベースの永続化が推奨されます。

ロードバランサーのフロントとバックの両方で以下のポートを設定します。

表1.10 アプリケーション Ingress ロードバランサー

ポート	バックエンドマシン (プールメンバー)	内部	外部	説明
443	デフォルトで Ingress ルーター Pod、コンピュート、またはワーカーを実行するマシン。	X	X	HTTPS トラフィック

ポート	バックエンドマシン (プールメン バー)	内部	外部	説明
80	デフォルトで Ingress ルーター Pod、 コンピュート、またはワーカーを実 行するマシン。	X	X	HTTP トラ フィック

## ヒント

クライアントの実際の IP アドレスがロードバランサーによって確認できる場合、ソースの IP ベースのセッション永続化を有効にすると、エンドツーエンドの TLS 暗号化を使用するアプリケーションのパフォーマンスを強化できます。



### 注記

Ingress ルーターの作業用の設定が OpenShift Container Platform クラスタに必要です。コントロールプレーンの初期化後に Ingress ルーターを設定する必要があります。

## NTP 設定

OpenShift Container Platform クラスタは、デフォルトでパブリック Network Time Protocol (NTP) サーバーを使用するように設定されます。ローカルのエンタープライズ NTP サーバーを使用する必要があるか、またはクラスタが切断されたネットワークにデプロイされている場合は、特定のタイムサーバーを使用するようにクラスタを設定できます。詳細は、[chrony タイムサービスの設定](#)のドキュメントを参照してください。

DHCP サーバーが NTP サーバー情報を提供する場合、Red Hat Enterprise Linux CoreOS (RHCOS) マシンの chrony タイムサービスは情報を読み取り、NTP サーバーとクロックを同期できます。

### 1.3.6. インストールマシンの設定

バイナリー **openshift-install** インストールプログラムおよび Ansible スクリプトを実行するには、Manager 上の RHV 環境および REST API にネットワークでアクセスできるように、RHV Manager または Red Hat Enterprise Linux (RHEL) を設定します。

#### 手順

1. Python3 および Ansible を更新またはインストールします。以下に例を示します。

```
# dnf update python3 ansible
```

2. [python3-ovirt-engine-sdk4](#) パッケージをインストールして、Python Software Development Kit を取得します。
3. **ovirt.image-template** Ansible ロールをインストールします。RHV Manager およびその他の Red Hat Enterprise Linux (RHEL) マシンでは、このロールは **ovirt-ansible-image-template** パッケージとして提供されます。たとえば、以下を入力します。

```
# dnf install ovirt-ansible-image-template
```

4. **ovirt.vm-infra** Ansible ロールをインストールします。RHV Manager およびその他の RHEL マシンでは、このロールは **ovirt-ansible-vm-infra** パッケージとして提供されます。

```
# dnf install ovirt-ansible-vm-infra
```

- 環境変数を作成し、その環境変数に絶対パスまたは相対パスを割り当てます。たとえば、以下を入力します。

```
$ export ASSETS_DIR=./wrk
```



### 注記

インストールプログラムはこの変数を使用して、重要なインストール関連のファイルを保存するディレクトリを作成します。その後、インストールプロセスはこの変数を再利用して、これらのアセットファイルを見つけます。このアセットディレクトリを削除しないでください。これは、クラスタのアンインストールに必要なになります。

## 1.3.7. RHV 用の CA 証明書の設定

Red Hat Virtualization (RHV) Manager から CA 証明書をダウンロードし、インストールマシンにこれを設定します。

RHV Manager からの Web サイトまたは **curl** コマンドを使用して、証明書をダウンロードできます。

その後、インストールプログラムに証明書を提供します。

### 手順

- 以下の 2 つの方法のいずれかを使用して CA 証明書をダウンロードします。
  - Manager の Web ページ (<https://<engine-fqdn>/ovirt-engine/>) に移動します。次に、**Downloads** で **CA Certificate** のリンクをクリックします。
  - 以下のコマンドを実行します。

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem ①
```

- ① **<engine-fqdn>** には、RHV Manager の完全修飾ドメイン名 (例: **rhv-env.virtlab.example.com**) を指定します。

- ルートレスユーザーに Manager へのアクセスを付与するように CA ファイルを設定します。CA ファイルのパーミッションを 8 進数の **0644** に設定します (シンボリック値: **-rw-r--r--**):

```
$ sudo chmod 0644 /tmp/ca.pem
```

- Linux の場合は、サーバー証明書のディレクトリに CA 証明書をコピーします。-p を使用してパーミッションを保存します。

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

- オペレーティングシステム用の証明書マネージャーに証明書を追加します。
  - MacOS の場合は、証明書ファイルをダブルクリックして、**Keychain Access** ユーティリティを使用してファイルを **System** キーチェーンに追加します。



- Linux の場合は、CA 信頼を更新します。

```
$ sudo update-ca-trust
```



### 注記

独自の認証局を使用する場合は、システムがこれを信頼することを確認します。

### 関連情報

- 詳細は、RHV ドキュメントの [Authentication and Security](#) を参照してください。

### 1.3.8. SSH プライベートキーの生成およびエージェントへの追加

クラスターでインストールのデバッグまたは障害復旧を実行する必要がある場合、**ssh-agent** とインストールプログラムの両方に SSH キーを指定する必要があります。このキーを使用してパブリッククラスターのブートストラップマシンにアクセスし、インストールの問題をトラブルシューティングできます。



### 注記

実稼働環境では、障害復旧およびデバッグが必要です。

このキーを使用して、ユーザー **core** としてマスターノードに対して SSH を実行できます。クラスターをデプロイする際に、キーは **core** ユーザーの `~/.ssh/authorized_keys` 一覧に追加されます。



### 注記

[AWS キーペア](#) などのプラットフォームに固有の方法で設定したキーではなく、ローカルキーを使用する必要があります。

### 手順

1. パスワードなしの認証に設定されている SSH キーがコンピューター上にない場合は、これを作成します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ ssh-keygen -t ed25519 -N "" \
  -f <path>/<file_name> ①
```

- ① `~/.ssh/id_rsa` などの、新規 SSH キーのパスおよびファイル名を指定します。既存のキーペアがある場合は、公開鍵が `~/.ssh` ディレクトリーにあることを確認します。

このコマンドを実行すると、指定した場所にパスワードを必要としない SSH キーが生成されます。



### 注記

FIPS で検証済み/進行中のモジュール (Modules in Process) 暗号ライブラリーを使用する OpenShift Container Platform クラスターを **x86\_64** アーキテクチャーにインストールする予定の場合は、**ed25519** アルゴリズムを使用するキーは作成しないでください。代わりに、**rsa** アルゴリズムまたは **ecdsa** アルゴリズムを使用するキーを作成します。

2. **ssh-agent** プロセスをバックグラウンドタスクとして開始します。

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

### 出力例

```
Agent pid 31874
```



### 注記

クラスターが FIPS モードにある場合は、FIPS 準拠のアルゴリズムのみを使用して SSH キーを生成します。鍵は RSA または ECDSA のいずれかである必要があります。

3. SSH プライベートキーを **ssh-agent** に追加します。

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

### 出力例

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

- 1** `~/.ssh/id_rsa` などの、SSH プライベートキーのパスおよびファイル名を指定します。

## 次のステップ

- OpenShift Container Platform をインストールする際に、SSH パブリックキーをインストールプログラムに指定します。

## 1.3.9. インストールプログラムの取得

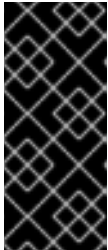
OpenShift Container Platform をインストールする前に、インストールファイルをローカルコンピューターにダウンロードします。

### 前提条件

- 500 MB のローカルディスク領域がある Linux または macOS を実行するコンピューターが必要です。

### 手順

1. OpenShift Cluster Manager サイトの [インフラストラクチャプロバイダー](#) ページにアクセスします。Red Hat アカウントがある場合は、認証情報を使ってログインします。アカウントがない場合はこれを作成します。
2. インフラストラクチャプロバイダーを選択します。
3. 選択するインストールタイプのページに移動し、オペレーティングシステムのインストールプログラムをダウンロードし、ファイルをインストール設定ファイルを保存するディレクトリーに配置します。



### 重要

インストールプログラムは、クラスターのインストールに使用するコンピューターにいくつかのファイルを作成します。クラスターのインストール完了後は、インストールプログラムおよびインストールプログラムが作成するファイルを保持する必要があります。ファイルはいずれもクラスターを削除するために必要になります。



### 重要

インストールプログラムで作成されたファイルを削除しても、クラスターがインストール時に失敗した場合でもクラスターは削除されません。クラスターを削除するには、特定のクラウドプロバイダー用の OpenShift Container Platform のアンインストール手順を実行します。

4. インストールプログラムを展開します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ tar xvf openshift-install-linux.tar.gz
```

5. [Red Hat OpenShift Cluster Manager](#) から [インストールプルシークレット](#) をダウンロードします。このプルシークレットを使用し、OpenShift Container Platform コンポーネントのコンテナイメージを提供する Quay.io など、組み込まれた各種の認証局によって提供されるサービスで認証できます。

## 1.3.10. Ansible Playbook のダウンロード

RHV に OpenShift Container Platform バージョン 4.6 をインストールするために Ansible Playbook をダウンロードします。

### 手順

- インストールマシンで、以下のコマンドを実行します。

```
$ mkdir playbooks
```

```
$ cd playbooks
```

```
$ curl -s -L -X GET https://api.github.com/repos/openshift/installer/contents/upi/ovirt?
ref=release-4.6 |
grep 'download_url.*\.yml' |
awk '{ print $2 }' | sed -r 's/(\"|,)//g' |
xargs -n 1 curl -O
```

## 次のステップ

- これらの Ansible Playbook をダウンロードしたら、インストールプログラムを実行してインストール設定ファイルを作成する前に、アセットディレクトリーの環境変数を作成し、**inventory.yml** ファイルをカスタマイズする必要もあります。

### 1.3.11. inventory.yml ファイル

**inventory.yml** ファイルを使用して、インストールする OpenShift Container Platform クラスターの各種の要素を定義し、作成します。これには、Red Hat Enterprise Linux CoreOS(RHCOS) イメージ、仮想マシンテンプレート、ブートストラップマシン、コントロールプレーンノード、ワーカーノードなどの要素が含まれます。また、**inventory.yml** を使用してクラスターを破棄します。

以下の **inventory.yml** の例は、パラメーターとそれらのデフォルト値を示しています。これらのデフォルト値の量と数は、RHV 環境で実稼働用の OpenShift Container Platform クラスターを実行するための要件を満たしています。

#### inventory.yml ファイルの例

```
---
all:
  vars:

    ovirt_cluster: "Default"
    ocp:
      assets_dir: "{{ lookup('env', 'ASSETS_DIR') }}"
      ovirt_config_path: "{{ lookup('env', 'HOME') }}/.ovirt/ovirt-config.yaml"

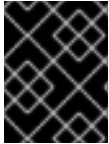
    # ---
    # {op-system} section
    # ---
    rhcos:
      image_url: "https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.6/latest/rhcos-
openstack.x86_64.qcow2.gz"
      local_cmp_image_path: "/tmp/rhcos.qcow2.gz"
      local_image_path: "/tmp/rhcos.qcow2"

    # ---
    # Profiles section
    # ---
    control_plane:
      cluster: "{{ ovirt_cluster }}"
      memory: 16GiB
      sockets: 4
      cores: 1
      template: rhcos_tpl
      operating_system: "rhcos_x64"
      type: high_performance
      graphical_console:
        headless_mode: false
      protocol:
        - spice
        - vnc
      disks:
        - size: 120GiB
          name: os
```

```
interface: virtio_scsi
storage_domain: depot_nvme
nics:
- name: nic1
  network: lab
  profile: lab

compute:
cluster: "{{ ovirt_cluster }}"
memory: 16GiB
sockets: 4
cores: 1
template: worker_rhcos_tpl
operating_system: "rhcos_x64"
type: high_performance
graphical_console:
  headless_mode: false
protocol:
- spice
- vnc
disks:
- size: 120GiB
  name: os
  interface: virtio_scsi
  storage_domain: depot_nvme
nics:
- name: nic1
  network: lab
  profile: lab

# ---
# Virtual machines section
# ---
vms:
- name: "{{ metadata.infraID }}-bootstrap"
  ocp_type: bootstrap
  profile: "{{ control_plane }}"
  type: server
- name: "{{ metadata.infraID }}-master0"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-master1"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-master2"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-worker0"
  ocp_type: worker
  profile: "{{ compute }}"
- name: "{{ metadata.infraID }}-worker1"
  ocp_type: worker
  profile: "{{ compute }}"
- name: "{{ metadata.infraID }}-worker2"
  ocp_type: worker
  profile: "{{ compute }}"
```



## 重要

Enter から始まる説明のあるパラメーターの値を入力します。それ以外の場合は、デフォルト値を使用するか、またはこえを新しい値に置き換えることができます。

### General セクション

- **ovirt\_cluster**: OpenShift Container Platform クラスターをインストールする既存の RHV クラスターの名前を入力します。
- **ocp.assets\_dir**: **openshift-install** インストールプログラムが生成するファイルを保存するために作成するディレクトリーのパス。
- **ocp.ovirt\_config\_path**: インストールプログラムが生成する **ovirt-config.yaml** ファイルのパス (**./wrk/install-config.yaml** など)。このファイルには、Manager の REST API との対話に必要な認証情報が含まれます。

### Red Hat Enterprise Linux CoreOS (RHCOS) セクション

- **image\_url**: ダウンロード用に指定した RHCOS イメージの URL を入力します。
- **local\_cmp\_image\_path**: 圧縮された RHCOS イメージのローカルダウンロードディレクトリーのパス。
- **local\_image\_path**: 展開した RHCOS イメージのローカルディレクトリーのパス。

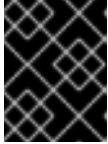
### Profiles セクション

このセクションは、2つのプロファイルで設定されます。

- **control\_plane**: ブートストラップおよびコントロールプレーンノードのプロファイル。
- **compute**: コンピュートプレーン内のワーカーノードのプロファイル。

これらのプロファイルには以下のパラメーターが含まれます。パラメーターのデフォルト値は、実稼働クラスターを実行するために必要な最小要件を満たします。これらの値は、ワークロードの要件に応じて増減したり、カスタマイズしたりできます。

- **cluster**: 値は、General セクションの **ovirt\_cluster** からクラスター名を取得します。
- **memory**: 仮想マシンに必要なメモリーの量 (GB)。
- **sockets**: 仮想マシンのソケット数。
- **cores**: 仮想マシンのコア数。
- **template**: 仮想マシンテンプレートの名前。複数のクラスターをインストールする計画があり、これらのクラスターが異なる仕様が含まれるテンプレートを使用する場合には、テンプレート名の先頭にクラスターの ID を付けます。
- **operating\_system**: 仮想マシンのゲストオペレーティングシステムのタイプ。oVirt/RHV バージョン 4.4 では、**Ignition script** の値を仮想マシンに渡すことができるようにするために、この値を **rhcos\_x64** にする必要があります。
- **type**: 仮想マシンのタイプとして **server** を入力します。



## 重要

**type** パラメーターの値を **high\_performance** から **server** に変更する必要があります。

- **disks**: ディスクの仕様。 **control\_plane** と **compute** ノードには、異なるストレージドメインを設定できます。
- **size**: ディスクの最小サイズ。
- **name**: RHV のターゲットクラスターに接続されたディスクの名前を入力します。
- **interface**: 指定したディスクのインターフェイスタイプを入力します。
- **storage\_domain**: 指定したディスクのストレージドメインを入力します。
- **nics**: 仮想マシンが使用する **name** および **network** を入力します。仮想ネットワークインターフェイスプロファイルを指定することもできます。デフォルトでは、NIC は oVirt/RHV MAC プールから MAC アドレスを取得します。

## 仮想マシンセクション

この最後のセクション **vms** は、クラスターで作成およびデプロイする予定の仮想マシンを定義します。デフォルトで、実稼働環境用の最小数のコントロールプレーンおよびワーカーノードが提供されます。

**vms** には 3 つの必須要素が含まれます。

- **name**: 仮想マシンの名前。この場合、 **metadata.infraID** は、仮想マシン名の先頭に **metadata.yml** ファイルのインフラストラクチャー ID を付けます。
- **ocp\_type**: OCP クラスター内の仮想マシンのロール。使用できる値は **bootstrap**、**master**、**worker** です。
- **profile**: それぞれの仮想マシンが仕様を継承するプロファイルの名前。この例で使用可能な値は **control\_plane** または **compute** です。  
仮想マシンがプロファイルから継承する値を上書きできます。これを実行するには、 **inventory.yml** の仮想マシンに **profile** 属性の名前を追加し、これに上書きする値を割り当てます。この例を確認するには、直前の **inventory.yml** の例の **name: "{{ metadata.infraID }}-bootstrap"** 仮想マシンを検査します。これには値が **server** の **type** 属性があり、この仮想マシンがそれ以外の場合に **control\_plane** プロファイルから継承する **type** 属性の値を上書きします。

## メタデータ変数

仮想マシンの場合、 **metadata.infraID** は、仮想マシンの名前の先頭に、Ignition ファイルのビルド時に作成する **metadata.json** ファイルのインフラストラクチャー ID を付けます。

Playbook は以下のコードを使用して、 **ocp.assets\_dir** にある特定のファイルから **infraID** を読み取ります。

```
---
- name: include metadata.json vars
  include_vars:
    file: "{{ ocp.assets_dir }}/metadata.json"
```

```
name: metadata
```

```
...
```

### 1.3.12. RHCOS イメージ設定の指定

**inventory.yml** ファイルの Red Hat Enterprise Linux CoreOS (RHCOS) イメージ設定を更新します。後にこのファイルを Playbook のいずれかとして実行すると、圧縮された Red Hat Enterprise Linux CoreOS (RHCOS) イメージが **image\_url** URL から **local\_cmp\_image\_path** ディレクトリーにダウンロードされます。次に Playbook はイメージを **local\_image\_path** ディレクトリーに展開し、これを使用して oVirt/RHV テンプレートを作成します。

#### 手順

1. インストールする OpenShift Container Platform バージョンの RHCOS イメージダウンロードページを見つけます (例: [/pub/openshift-v4/dependencies/rhcos/latest/latest のインデックス](#))。
2. そのダウンロードページから、**https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.6/latest/rhcos-opensstack.x86\_64.qcow2.gz** などの OpenStack **qcow2** イメージの URL をコピーします。
3. 先のステップでダウンロードした **inventory.yml** Playbook を編集します。この中で、URL を **image\_url** の値として貼り付けます。以下に例を示します。

```
rhcos:
  "https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.6/latest/rhcos-
  opensstack.x86_64.qcow2.gz"
```

### 1.3.13. インストール設定ファイルの作成

インストールプログラム **openshift-install** を実行し、先に指定または収集した情報でプロンプトに回答し、インストール設定ファイルを作成します。

プロンプトに回答すると、インストールプログラムは、以前に指定したアセットディレクトリーの **install-config.yaml** ファイルの初期バージョンを作成します (例: **./wrk/install-config.yaml**)。

インストールプログラムは、Manager に到達して REST API を使用するために必要なすべての接続パラメーターが含まれる **\$HOME/.ovirt/ovirt-config.yaml** ファイルも作成します。

注: インストールプロセスでは、**Internal API virtual IP** および **Ingress virtual IP** などの一部のパラメーターに指定する値を使用しません。それらの値はインフラストラクチャー DNS にすでに設定されているためです。

また、**oVirt cluster**、**oVirt storage**、および **oVirt network** などの値のような **inventory.yml** のパラメーターに指定する値を使用します。また、スクリプトを使用して **install-config.yaml** の同じ値を削除するか、またはこれを前述の **virtual IPs** に置き換えます。

#### 手順

1. インストールプログラムを実行します。

```
$ openshift-install create install-config --dir $ASSETS_DIR
```



2. インストールプログラムのプロンプトに回答し、システムに関する情報を提供します。

## 出力例

```
? SSH Public Key /home/user/.ssh/id_dsa.pub
? Platform <ovirt>
? Engine FQDN[:PORT] [? for help] <engine.fqdn>
? Enter ovirt-engine username <ocpadmin@internal>
? Enter password <*****>
? oVirt cluster <cluster>
? oVirt storage <storage>
? oVirt network <net>
? Internal API virtual IP <172.16.0.252>
? Ingress virtual IP <172.16.0.251>
? Base Domain <example.org>
? Cluster Name <ocp4>
? Pull Secret [? for help] <*****>
```

```
? SSH Public Key /home/user/.ssh/id_dsa.pub
? Platform <ovirt>
? Engine FQDN[:PORT] [? for help] <engine.fqdn>
? Enter ovirt-engine username <ocpadmin@internal>
? Enter password <*****>
? oVirt cluster <cluster>
? oVirt storage <storage>
? oVirt network <net>
? Internal API virtual IP <172.16.0.252>
? Ingress virtual IP <172.16.0.251>
? Base Domain <example.org>
? Cluster Name <ocp4>
? Pull Secret [? for help] <*****>
```

**Internal API virtual IP** および **Ingress virtual IP** について、DNS サービスの設定時に指定した IP アドレスを指定します。

さらに、**oVirt cluster** および **Base Domain** プロンプトに対して入力する値は REST API および作成するアプリケーションの URL の一部を設定します (例: <https://api.ocp4.example.org:6443/> and <https://console-openshift-console.apps.ocp4.example.org/>)。

[Red Hat OpenShift Cluster Manager からプルシークレット](#) を取得できます。

### 1.3.14. install-config.yaml のカスタマイズ

ここでは、3つの python スクリプトを使用して、インストールプログラムのデフォルト動作の一部を上書きします。

- デフォルトでは、インストールプログラムはマシン API を使用してノードを作成します。このデフォルトの動作を上書きするには、コンピュートノードの数をゼロ (0) レプリカに設定します。後に Ansible Playbook を使用してコンピュートノードを作成します。
- デフォルトでは、インストールプログラムはノードのマシンネットワークの IP 範囲を設定します。このデフォルトの動作を上書きするには、インフラストラクチャーに一致するように IP 範囲を設定します。
- デフォルトでは、インストールプログラムはプラットフォームを **ovirt** に設定します。ただ

し、ユーザーによってプロビジョニングされるインフラストラクチャーにクラスターをインストールすることは、ベアメタルにクラスターをインストールすることに似ています。したがって、ovirt プラットフォームセクションを **install-config.yaml** から削除し、プラットフォームを **none** に変更します。代わりに、**inventory.yml** を使用して、必要な設定をすべて指定します。



## 注記

これらのスニペットは Python 3 および Python 2 で動作します。

## 手順

1. コンピュートノードの数をゼロ (0) レプリカに設定します。

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
conf["compute"][0]["replicas"] = 0
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

2. マシンネットワークの IP 範囲を設定します。たとえば、範囲を **172.16.0.0/16** に設定するには、以下を実行します。

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
conf["networking"]["machineNetwork"][0]["cidr"] = "172.16.0.0/16"
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

3. **ovirt** セクションを削除し、プラットフォームを **none** に変更します。

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
platform = conf["platform"]
del platform["ovirt"]
platform["none"] = {}
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

### 1.3.15. マニフェストファイルの生成

インストールプログラムを使用して、アセットディレクトリーにマニフェストファイルのセットを生成します。

マニフェストファイルを生成するコマンドにより、**install-config.yaml** ファイルを使用する前に警告メッセージが表示されます。

**install-config.yaml** ファイルを再利用する予定の場合には、マニフェストファイルを生成する前にバックアップしてからバックアップコピーを作成してください。

## 手順

1. オプション: **install-config.yaml** ファイルのバックアップコピーを作成します。

```
$ cp install-config.yaml install-config.yaml.backup
```

- 2. アセットディレクトリーにマニフェストのセットを生成します。

```
$ openshift-install create manifests --dir $ASSETS_DIR
```

このコマンドにより、以下の情報が表示されます。

### 出力例

```
INFO Consuming Install Config from target directory
WARNING Making control-plane schedulable by setting MastersSchedulable to true for
Scheduler cluster settings
```

このコマンドにより、以下のマニフェストファイルが生成されます。

### 出力例

```
$ tree
├── wrk
│   ├── manifests
│   │   ├── 04-openshift-machine-config-operator.yaml
│   │   ├── cluster-config.yaml
│   │   ├── cluster-dns-02-config.yml
│   │   ├── cluster-infrastructure-02-config.yml
│   │   ├── cluster-ingress-02-config.yml
│   │   ├── cluster-network-01-crd.yml
│   │   ├── cluster-network-02-config.yml
│   │   ├── cluster-proxy-01-config.yaml
│   │   ├── cluster-scheduler-02-config.yml
│   │   ├── cvo-overrides.yaml
│   │   ├── etcd-ca-bundle-configmap.yaml
│   │   ├── etcd-client-secret.yaml
│   │   ├── etcd-host-service-endpoints.yaml
│   │   ├── etcd-host-service.yaml
│   │   ├── etcd-metric-client-secret.yaml
│   │   ├── etcd-metric-serving-ca-configmap.yaml
│   │   ├── etcd-metric-signer-secret.yaml
│   │   ├── etcd-namespace.yaml
│   │   ├── etcd-service.yaml
│   │   ├── etcd-serving-ca-configmap.yaml
│   │   ├── etcd-signer-secret.yaml
│   │   ├── kube-cloud-config.yaml
│   │   ├── kube-system-configmap-root-ca.yaml
│   │   ├── machine-config-server-tls-secret.yaml
│   │   └── openshift-config-secret-pull-secret.yaml
│   └── openshift
│       ├── 99_kubeadmin-password-secret.yaml
│       ├── 99_openshift-cluster-api_master-user-data-secret.yaml
│       ├── 99_openshift-cluster-api_worker-user-data-secret.yaml
│       ├── 99_openshift-machineconfig_99-master-ssh.yaml
│       ├── 99_openshift-machineconfig_99-worker-ssh.yaml
│       └── openshift-install-manifests.yaml
```

## 次のステップ

- コントロールプレーンノードをスケジュール対象外にします。

### 1.3.16. コントロールプレーンノードのスケジュール対象外の設定

コントロールプレーンマシンを手動で作成し、デプロイしているため、コントロールプレーンノードをスケジュール対象外にするようにマニフェストファイルを設定する必要があります。

#### 手順

1. コントロールプレーンノードをスケジュール対象外にするには、以下を入力します。

```
$ python3 -c 'import os, yaml
path = "%s/manifests/cluster-scheduler-02-config.yml" % os.environ["ASSETS_DIR"]
data = yaml.safe_load(open(path))
data["spec"]["mastersSchedulable"] = False
open(path, "w").write(yaml.dump(data, default_flow_style=False))'
```

### 1.3.17. Ignition ファイルのビルド

生成および変更したマニフェストファイルから Ignition ファイルを作成するには、インストールプログラムを実行します。このアクションにより、Ignition ファイルをフェッチし、ノードを作成するために必要な設定を実行する Red Hat Enterprise Linux CoreOS (RHCOS) マシン **initramfs** が作成されます。

Ignition ファイルのほかに、インストールプログラムは以下を生成します。

- **oc** および **kubectl** ユーティリティを使用してクラスターに接続するための管理者認証情報が含まれる **auth** ディレクトリー。
- OpenShift Container Platform クラスター名、クラスター ID、および現行インストールのインフラストラクチャー ID などの情報を含む **metadata.json** ファイル。

このインストールプロセスの Ansible Playbook は、**infraID** の値を、作成する仮想マシンの接頭辞として使用します。これにより、同じ oVirt/RHV クラスターに複数のインストールがある場合の命名の競合が回避されます。



#### 注記

Ignition 設定ファイルの証明書は 24 時間後に有効期限が切れます。最初の証明書のローテーションが終了するように、クラスターのインストールを完了し、クラスターを動作が低下していない状態で 24 時間実行し続ける必要があります。

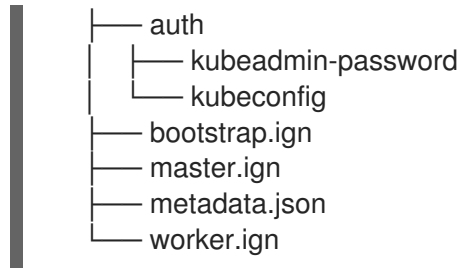
#### 手順

1. Ignition ファイルをビルドするには、以下を入力します。

```
$ openshift-install create ignition-configs --dir $ASSETS_DIR
```

#### 出力例

```
$ tree
.
└── wrk
```



### 1.3.18. テンプレートおよび仮想マシンの作成

**inventory.yml** の変数を確認した後に、最初の Ansible プロビジョニング Playbook **create-templates-and-vms.yml** を実行します。

この Playbook は、**\$HOME/.ovirt/ovirt-config.yaml** から RHV Manager の接続パラメーターを使用し、アセットディレクトリーで **metadata.json** を読み取ります。

ローカルの Red Hat Enterprise Linux CoreOS (RHCOS) イメージが存在しない場合、Playbook は **inventory.yml** の **image\_url** に指定した URL からダウンロードします。これはイメージを展開し、これを RHV にアップロードしてテンプレートを作成します。

Playbook は、**inventory.yml** ファイルの **control\_plane** と **compute** プロファイルに基づいてテンプレートを作成します。これらのプロファイルの名前が異なる場合、2つのテンプレートが作成されます。

Playbook が完了すると、作成される仮想マシンは停止します。他のインフラストラクチャー要素の設定に役立つ情報を取得できます。たとえば、仮想マシンの MAC アドレスを取得して、仮想マシンに永続的な IP アドレスを割り当てるように DHCP を設定できます。

#### 手順

1. **inventory.yml** の **control\_plane** および **compute** 変数で、**type: high\_performance** の両方のインスタンスを **type: server** に変更します。
2. オプション: 同じクラスターに複数のインストールを実行する予定の場合には、OCP インストールごとに異なるテンプレートを作成します。**inventory.yml** ファイルで、**template** の値の先頭に **infraID** を付けます。以下に例を示します。

```

control_plane:
  cluster: "{{ ovirt_cluster }}"
  memory: 16GiB
  sockets: 4
  cores: 1
  template: "{{ metadata.infraID }}-rhcos_tpl"
  operating_system: "rhcos_x64"
  ...
  
```

3. テンプレートおよび仮想マシンを作成します。

```
$ ansible-playbook -i inventory.yml create-templates-and-vms.yml
```

### 1.3.19. ブートストラップマシンの作成

**bootstrap.yml** Playbook を実行してブートストラップマシンを作成します。この Playbook はブートストラップ仮想マシンを起動し、これをアセットディレクトリーから **bootstrap.ign** Ignition ファイルに

渡します。ブートストラップノードは、Ignition ファイルをコントロールプレーンノードに送信できるように設定します。

ブートストラッププロセスをモニターするには、RHV 管理ポータルでコンソールを使用するか、SSH を使用して仮想マシンに接続します。

### 手順

1. ブートストラップマシンを作成します。

```
$ ansible-playbook -i inventory.yml bootstrap.yml
```

2. 管理ポータルまたは SSH のコンソールを使用してブートストラップマシンに接続します。<code>bootstrap\_ip</code> をブートストラップノードの IP アドレスに置き換えます。SSH を使用するには、以下を入力します。

```
$ ssh core@<bootstrap.ip>
```

3. ブートストラップノードからリリースイメージサービスについての **bootkube.service** journald ユニットログを収集します。

```
[core@ocp4-1k6b4-bootstrap ~]$ journalctl -b -f -u release-image.service -u bootkube.service
```



### 注記

ブートストラップノードの **bootkube.service** ログは、etcd の **connection refused** エラーを出力し、ブートストラップサーバーがコントロールプレーンノード (別名マスターノード) の etcd に接続できないことを示します。etcd が各コントロールプレーンノードで起動し、ノードがクラスターに参加した後は、エラーは発生しなくなるはずで

## 1.3.20. コントロールプレーンノードの作成

**masters.yml** Playbook を実行してコントロールプレーンノードを作成します。この Playbook は **master.ign** Ignition ファイルをそれぞれの仮想マシンに渡します。Ignition ファイルには、<https://api-int.ocp4.example.org:22623/config/master> などの URL から Ignition を取得するためのコントロールプレーンノードのディレクティブが含まれます。この URL のポート番号はロードバランサーによって管理され、クラスター内でのみアクセスできます。

### 手順

1. コントロールプレーンノードを作成します。

```
$ ansible-playbook -i inventory.yml masters.yml
```

2. Playbook がコントロールプレーンを作成する間に、ブートストラッププロセスをモニターします。

```
$ openshift-install wait-for bootstrap-complete --dir $ASSETS_DIR
```

### 出力例

```
INFO API v1.18.3+b74c5ed up
INFO Waiting up to 40m0s for bootstrapping to complete...
```

3. コントロールプレーンノードおよび etcd のすべての Pod が実行されている場合、インストールプログラムは以下の出力を表示します。

### 出力例

```
INFO It is now safe to remove the bootstrap resources
```

## 1.3.21. クラスタステータスの確認

インストール時またはインストール後に OpenShift Container Platform クラスタのステータスを確認することができます。

### 手順

1. クラスタ環境で、管理者の kubeconfig ファイルをエクスポートします。

```
$ export KUBECONFIG=$ASSETS_DIR/auth/kubeconfig
```

**kubeconfig** ファイルには、クライアントを正しいクラスタおよび API サーバーに接続するために CLI で使用されるクラスタについての情報が含まれます。

2. デプロイメント後に作成されたコントロールプレーンおよびコンピュータマシンを表示します。

```
$ oc get nodes
```

3. クラスタのバージョンを表示します。

```
$ oc get clusterversion
```

4. Operator のステータスを表示します。

```
$ oc get clusteroperator
```

5. クラスタ内のすべての実行中の Pod を表示します。

```
$ oc get pods -A
```

## 1.3.22. ブートストラップマシンの削除

**wait-for** コマンドがブートストラッププロセスが完了したことを示していることを確認したら、ブートストラップ仮想マシンを削除してコンピュータ、メモリ、およびストレージリソースを解放する必要があります。また、ロードバランサーディレクティブからブートストラップマシンの設定を削除します。

### 手順

1. クラスタからブートストラップマシンを削除するには、以下を実行します。

```
$ ansible-playbook -i inventory.yml retire-bootstrap.yml
```

- ロードバランサーディレクティブからブートストラップマシンの設定を削除します。

### 1.3.23. ワーカーノードの作成およびインストールの完了

ワーカーノードの作成は、コントロールプレーンノードの作成と同様です。ただし、ワーカーノードはクラスターに自動的に参加しません。これらをクラスターに追加するには、ワーカーの保留状態の CSR(証明書署名要求)を確認し、承認します。

最初の要求の承認後に、ワーカーノードがすべて承認されるまで CSR の承認を続けます。このプロセスが完了すると、ワーカーノードは **Ready** になり、Pod がそれらで実行されるようにスケジュールできます。

最後に、コマンドラインを監視し、インストールプロセスが完了するタイミングを確認します。

#### 手順

- ワーカーノードを作成します。

```
$ ansible-playbook -i inventory.yml workers.yml
```

- すべての CSR を一覧表示するには、以下を入力します。

```
$ oc get csr -A
```

最終的に、このコマンドはノードごとに1つの CSR を表示します。以下に例を示します。

#### 出力例

```
NAME          AGE  SIGNERNAME                                REQUESTOR
CONDITION
csr-2lnxd     63m  kubernetes.io/kubelet-serving            system:node:ocp4-1k6b4-
master0.ocp4.example.org                Approved,Issued
csr-hff4q     64m  kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-hsn96     60m  kubernetes.io/kubelet-serving            system:node:ocp4-1k6b4-
master2.ocp4.example.org                Approved,Issued
csr-m724n     6m2s kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
csr-p4dz2     60m  kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-t9vfj     60m  kubernetes.io/kubelet-serving            system:node:ocp4-1k6b4-
master1.ocp4.example.org                Approved,Issued
csr-tggtr     61m  kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-wcbrf     7m6s kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
```

- 一覧をフィルターし、保留中の CSR のみを表示するには、以下を実行します。



```
$ watch "oc get csr -A | grep pending -i"
```

このコマンドは 2 秒ごとに出力を更新し、保留中の CSR のみを表示します。以下に例を示します。

### 出力例

```
Every 2.0s: oc get csr -A | grep pending -i

csr-m724n 10m kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
csr-wcbrf 11m kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
```

- 保留中のそれぞれの要求を検査します。以下に例を示します。

### 出力例

```
$ oc describe csr csr-m724n
```

### 出力例

```
Name:          csr-m724n
Labels:        <none>
Annotations:   <none>
CreationTimestamp: Sun, 19 Jul 2020 15:59:37 +0200
Requesting User: system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper
Signer:        kubernetes.io/kube-apiserver-client-kubelet
Status:        Pending
Subject:
  Common Name:  system:node:ocp4-lk6b4-worker1.ocp4.example.org
  Serial Number:
  Organization: system:nodes
Events: <none>
```

- CSR 情報が正しい場合は、要求を承認します。

```
$ oc adm certificate approve csr-m724n
```

- インストールプロセスが完了するまで待機します。

```
$ openshift-install wait-for install-complete --dir $ASSETS_DIR --log-level debug
```

インストールが完了すると、コマンドラインには OpenShift Container Platform Web コンソールの URL と、管理者のユーザー名およびパスワードが表示されます。

### 1.3.24. OpenShift Container Platform の Telemetry アクセス

OpenShift Container Platform 4.6 では、クラスターのヘルスと更新の成功に関するメトリクスを提供するためにデフォルトで実行される Telemetry サービスには、インターネットアクセスが必要です。クラスターがインターネットに接続されている場合、Telemetry は自動的に実行され、クラスターは [OpenShift Cluster Manager](#) に登録されます。

OpenShift Cluster Manager インベントリが正常である (Telemetry によって自動的に維持、または OpenShift Cluster Manager を使用して手動で維持) ことを確認した後に、[subscription watch](#) を使用して、アカウントまたはマルチクラスターレベルで OpenShift Container Platform サブスクリプションを追跡します。

## 関連情報

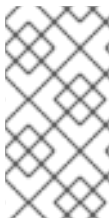
- Telemetry サービスの詳細は、[リモートヘルスマニターリング](#) について参照してください。

## 1.4. RHV でのクラスターのアンインストール

OpenShift Container Platform クラスターを Red Hat Virtualization (RHV) から削除することができます。

### 1.4.1. インストーラーでプロビジョニングされるインフラストラクチャーを使用するクラスターの削除

インストーラーでプロビジョニングされるインフラストラクチャーを使用するクラスターは、クラウドから削除できます。



#### 注記

アンインストール後に、とくにユーザーによってプロビジョニングされるインフラストラクチャー (UPI) クラスターで適切に削除されていないリソースがあるかどうかについて、クラウドプロバイダーを確認します。インストーラーが作成されなかったり、インストーラーがアクセスできない場合には、リソースがある可能性があります。

## 前提条件

- クラスターをデプロイするために使用したインストールプログラムのコピーがあります。
- クラスター作成時にインストールプログラムが生成したファイルがあります。

## 手順

1. クラスターをインストールするために使用したコンピューターのインストールプログラムが含まれるディレクトリーから、以下のコマンドを実行します。

```
$ ./openshift-install destroy cluster \
--dir <installation_directory> --log-level info 1 2
```

- 1** **<installation\_directory>** には、インストールファイルを保存したディレクトリーへのパスを指定します。
- 2** 異なる詳細情報を表示するには、**info** ではなく、**warn**、**debug**、または **error** を指定します。



#### 注記

クラスターのクラスター定義ファイルが含まれるディレクトリーを指定する必要があります。クラスターを削除するには、インストールプログラムでこのディレクトリーにある **metadata.json** ファイルが必要になります。

2. オプション: `<installation_directory>` ディレクトリーおよび OpenShift Container Platform インストールプログラムを削除します。

### 1.4.2. ユーザーによってプロビジョニングされるインフラストラクチャーを使用するクラスターの削除

クラスターの使用が完了したら、ユーザーによってプロビジョニングされるインフラストラクチャーを使用するクラスターをクラウドから削除できます。

#### 前提条件

- クラスターのインストールに使用した元の Playbook ファイル、アセットディレクトリーおよびファイル、および `$ASSETS_DIR` 環境変数が含まれます。通常、クラスターのインストール時に使用したのと同じコンピューターを使用してこれを実行できます。

#### 手順

1. クラスターを削除するには、以下を入力します。

```
$ ansible-playbook -i inventory.yml \  
  retire-bootstrap.yml \  
  retire-masters.yml \  
  retire-workers.yml
```

2. DNS、ロードバランサー、およびこのクラスターの他のインフラストラクチャーに追加した設定を削除します。