



# Red Hat

## OpenShift Container Platform 4.14

### Nutanix へのインストール

Nutanix への OpenShift Container Platform のインストール



# OpenShift Container Platform 4.14 Nutanix へのインストール

---

Nutanix への OpenShift Container Platform のインストール

## Legal Notice

Copyright © 2025 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

Node.js® 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® 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.

## Abstract

このドキュメントでは、Nutanix に OpenShift Container Platform をインストールする方法を説明します。

## Table of Contents

<b>第1章 NUTANIX へのインストールの準備</b>	3
1.1. NUTANIX バージョン要件	3
1.2. 環境要件	3
1.3. CLOUD CREDENTIAL OPERATOR ユーティリティーの設定	7
<b>第2章 クラスターの NUTANIX へのインストール</b>	9
2.1. 前提条件	9
2.2. OPENSHIFT CONTAINER PLATFORM のインターネットアクセス	9
2.3. PRISM CENTRAL のインターネットアクセス	10
2.4. クラスターノードの SSH アクセス用のキーペアの生成	10
2.5. インストールプログラムの取得	12
2.6. NUTANIX ROOT CA 証明書をシステム信頼に追加する	13
2.7. インストール設定ファイルの作成	13
2.8. バイナリーのダウンロードによる OPENSHIFT CLI のインストール	20
2.9. NUTANIX の IAM の設定	22
2.10. NUTANIX CCM に必要な CONFIG MAP とシークレットリソースの追加	24
2.11. クラスターのデプロイ	25
2.12. デフォルトのストレージコンテナーの設定	27
2.13. OPENSHIFT CONTAINER PLATFORM の TELEMETRY アクセス	27
2.14. 関連情報	27
2.15. 次のステップ	27
<b>第3章 ネットワークが制限された環境での NUTANIX へのクラスターのインストール</b>	28
3.1. 前提条件	28
3.2. ネットワークが制限された環境でのインストールについて	28
3.3. クラスターノードの SSH アクセス用のキーペアの生成	29
3.4. NUTANIX ROOT CA 証明書をシステム信頼に追加する	31
3.5. RHCOS クラスターイメージのダウンロード	31
3.6. インストール設定ファイルの作成	32
3.7. バイナリーのダウンロードによる OPENSHIFT CLI のインストール	40
3.8. NUTANIX の IAM の設定	42
3.9. クラスターのデプロイ	45
3.10. インストール後の設定	46
3.11. OPENSHIFT CONTAINER PLATFORM の TELEMETRY アクセス	48
3.12. 関連情報	48
3.13. 次のステップ	48
<b>第4章 NUTANIX への 3 ノードクラスターのインストール</b>	49
4.1. 3 ノードクラスターの設定	49
4.2. 次のステップ	49
<b>第5章 NUTANIX でのクラスターのアンインストール</b>	50
5.1. INSTALLER-PROVISIONED INFRASTRUCTURE を使用するクラスターの削除	50
<b>第6章 NUTANIX のインストール設定パラメーター</b>	51
6.1. NUTANIX で使用可能なインストール設定パラメーター	51



# 第1章 NUTANIX へのインストールの準備

OpenShift Container Platform クラスターをインストールする前に、Nutanix 環境が以下の要件を満たしていることを確認してください。

## 1.1. NUTANIX バージョン要件

以下の要件を満たす Nutanix 環境に OpenShift Container Platform クラスターをインストールする必要があります。

表1.1 Nutanix 仮想環境のバージョン要件

コンポーネント	必須バージョン
Nutanix AOS	6.5.2.7 以降
Prism Central	pc.2022.6 以降

## 1.2. 環境要件

OpenShift Container Platform クラスターをインストールする前に、以下の Nutanix AOS 環境要件を確認してください。

### 1.2.1. 必要なアカウント特権

インストールプログラムには、クラスターをデプロイし、その日次の操作を維持するために必要な権限のある Nutanix アカウントへのアクセスが必要です。以下のオプションを使用できます。

- 管理者権限を持つローカル Prism Central ユーザーアカウントを使用できます。ローカルアカウントを使用することは、必要な権限を持つアカウントへのアクセスを許可する最も簡単な方法です。
- 組織のセキュリティーポリシーにより、より制限の厳しい権限セットを使用する必要がある場合は、次の表にリストされている権限を使用して、Prism Central でカスタムの Cloud Native ロールを作成します。その後、Prism Central 認証ディレクターのメンバーであるユーザーアカウントにロールを割り当てることができます。

このユーザーアカウントを管理するときは、次の点を考慮してください。

- エンティティをロールに割り当てるときは、ユーザーが仮想マシンのデプロイに必要な Prism Element とサブネットのみにアクセスできるようにしてください。
- ユーザーが、仮想マシンを割り当てる必要があるプロジェクトのメンバーであることを確認してください。

詳細は、[Custom Cloud Native ロール](#) の作成、[ロールの割り当て](#)、[プロジェクトへのユーザーの追加](#)に関する Nutanix ドキュメントを参照してください。

### 例1.1 Custom Cloud Native ロールの作成に必要な権限

Nutanix オブジェクト	必要になる場合	Nutanix API で必要な権限	説明
Categories	常時	<b>Create_Category_Mapping</b> <b>Create_Or_Update_Name_Category</b> <b>Create_Or_Update_Value_Category</b> <b>Delete_Category_Mapping</b> <b>Delete_Name_Category</b> <b>Delete_Value_Category</b> <b>View_Category_Mapping</b> <b>View_Name_Category</b> <b>View_Value_Category</b>	OpenShift Container Platform マシンに割り当てられたカテゴリを作成、読み取り、削除します。
Images	常時	<b>Create_Image</b> <b>Delete_Image</b> <b>View_Image</b>	OpenShift Container Platform マシンに使用されるオペレーティングシステムイメージを作成、読み取り、削除します。
仮想マシン	常時	<b>Create_Virtual_Machine</b> <b>Delete_Virtual_Machine</b> <b>View_Virtual_Machine</b>	OpenShift Container Platform マシンを作成、読み取り、削除します。
クラスター	常時	<b>View_Cluster</b>	OpenShift Container Platform マシンをホストする Prism Element クラスターを表示します。
サブネット	常時	<b>View_Subnet</b>	OpenShift Container Platform マシンをホストするサブネットを表示します。

Nutanix オブジェクト	必要になる場合	Nutanix API で必要な権限	説明
プロジェクト	プロジェクトをコンピューティングマシン、コントロールプレーンマシン、またはすべてのマシンに関連付ける場合。	<a href="#">View_Project</a>	Prism Central で定義されたプロジェクトを表示し、プロジェクトを OpenShift Container Platform マシンに割り当てることができるようになります。

### 1.2.2. クラスターの制限

利用可能なリソースはクラスターによって異なります。Nutanix 環境内で可能なクラスターの数は、主に、使用可能なストレージスペースと、クラスターが作成するリソースに関連する制限、および IP アドレスやネットワークなど、クラスターをデプロイするために必要なリソースによって制限されます。

### 1.2.3. クラスターリソース

標準クラスターを使用するには、最低 800 GB のストレージが必要です。

installer-provisioned infrastructure を使用する OpenShift Container Platform クラスターをデプロイする場合、インストールプログラムは Nutanix インスタンスに複数のリソースを作成できる必要があります。これらのリソースは 856 GB のストレージを使用しますが、ブートストラップノードはインストールプロセスの一部として破棄されます。

標準的な OpenShift Container Platform インストールでは、以下のリソースを作成します。

- 1 ラベル
- 仮想マシン:
  - 1 ディスクイメージ
  - 1 一時的ブートストラップノード
  - 3 コントロールプレーンノード
  - 3 コンピュートマシン

### 1.2.4. ネットワーク要件

ネットワークには AHV IP アドレス管理 (IPAM) または動的ホスト設定プロトコル (DHCP) のいずれかを使用し、クラスターマシンに永続的な IP アドレスを提供するように設定されていることを確認する必要があります。さらに、OpenShift Container Platform クラスターをインストールする前に、以下のネットワークリソースを作成します。

- IP アドレス
- DNS レコード



## 注記

クラスターの各 OpenShift Container Platform ノードは、DHCP を使用して検出可能な Network Time Protocol (NTP) サーバーにアクセスできることが推奨されます。NTP サーバーなしでインストールが可能です。ただし、NTP サーバーは、非同期サーバークロックに通常関連するエラーを防ぎます。

### 1.2.4.1. 必要な IP アドレス

installer-provisioned installation には、2 つの静的仮想 IP (VIP) アドレスが必要です。

- API の VIP アドレスが必要です。このアドレスは、クラスター API にアクセスするために使用されます。
- Ingress 用の VIP アドレスが必要です。このアドレスは、クラスターの Ingress トライフィックに使用されます。

これらの IP アドレスは、OpenShift Container Platform クラスターをインストールするときに指定します。

### 1.2.4.2. DNS レコード

OpenShift Container Platform クラスターをホストする Nutanix インスタンスについて 2 つの静的 IP アドレスの DNS レコードを適切な DNS サーバーに作成する必要があります。各レコードで、`<cluster_name>` はクラスター名で、`<base_domain>` は、クラスターのインストール時に指定するクラスターのベースドメインです。

独自の DNS または DHCP サーバーを使用する場合は、ブートストラップ、コントロールプレーン、コンピュートノードなどの各ノードのレコードも作成する必要があります。

完全な DNS レコードは `<component>.<cluster_name>.<base_domain>` の形式を取ります。

表1.2 必要な DNS レコード

コンポーネント	レコード	説明
API VIP	<code>api.&lt;cluster_name&gt;.&lt;base_domain&gt;</code>	この DNS A/AAAA または CNAME レコードは、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必要があります。このレコードは、クラスター外のクライアントおよびクラスター内のすべてのノードで解決できる必要があります。

コン ポーネ ント	レコード	説明
Ingress VIP	*.apps.<cluster_name>.<base_domain>.	Ingress ルーター Pod を実行するマシンをターゲットにするロードバランサーを参照するワイルドカード DNS A/AAAA または CNAME レコードです。このレコードは、クラスター外のクライアントおよびクラスター内のすべてのノードで解決できる必要があります。

### 1.3. CLOUD CREDENTIAL OPERATOR ユーティリティーの設定

Cloud Credential Operator (CCO) は、クラウドプロバイダーの認証情報を Kubernetes カスタムリソース定義 (CRD) として管理します。Nutanix にクラスターをインストールするには、インストールプロセスの一部として CCO を **manual** モードに設定する必要があります。

Cloud Credential Operator (CCO) が手動モードで動作しているときにクラスターの外部からクラウドクレデンシャルを作成および管理するには、CCO ユーティリティー (**ccctl**) バイナリーを抽出して準備します。



#### 注記

**ccctl** ユーティリティーは、Linux 環境で実行する必要がある Linux バイナリーです。

#### 前提条件

- クラスター管理者のアクセスを持つ OpenShift Container Platform アカウントを使用できる。
- OpenShift CLI (**oc**) がインストールされている。

#### 手順

- 次のコマンドを実行して、OpenShift Container Platform リリースイメージの変数を設定します。

```
$ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
```

- 以下のコマンドを実行して、OpenShift Container Platform リリースイメージから CCO コンテナイメージを取得します。

```
$ CCO_IMAGE=$(oc adm release info --image-for='cloud-credential-operator'  
$RELEASE_IMAGE -a ~/.pull-secret)
```



#### 注記

**\$RELEASE\_IMAGE** のアーキテクチャーが、**ccctl** ツールを使用する環境のアーキテクチャーと一致していることを確認してください。

3. 以下のコマンドを実行して、OpenShift Container Platform リリースイメージ内の CCO コンテナイメージから **ccocctl** バイナリーを抽出します。

```
$ oc image extract $CCO_IMAGE --file="/usr/bin/ccocctl" -a ~/.pull-secret
```

4. 次のコマンドを実行して、権限を変更して **ccocctl** を実行可能にします。

```
$ chmod 775 ccocctl
```

## 検証

- **ccocctl** が使用できることを確認するには、**help** ファイルを表示します。コマンドを実行するときは、相対ファイル名を使用します。以下に例を示します。

```
$ ./ccocctl.rhel9
```

## 出力例

```
OpenShift credentials provisioning tool

Usage:
  ccocctl [command]

Available Commands:
  alibabacloud  Manage credentials objects for alibaba cloud
  aws          Manage credentials objects for AWS cloud
  azure        Manage credentials objects for Azure
  gcp          Manage credentials objects for Google cloud
  help         Help about any command
  ibmcloud     Manage credentials objects for IBM Cloud
  nutanix      Manage credentials objects for Nutanix

Flags:
  -h, --help  help for ccocctl

Use "ccocctl [command] --help" for more information about a command.
```

## 関連情報

- [手動で維持された認証情報でクラスターを更新する準備](#)

## 第2章 クラスターの NUTANIX へのインストール

OpenShift Container Platform バージョン 4.14 では、以下のオプションのいずれかを選択して、Nutanix インスタンスにクラスターをインストールできます。

**インストーラーによってプロビジョニングされたインフラストラクチャーの使用:** インストーラーによってプロビジョニングされたインフラストラクチャーを使用するには、次のセクションの手順を使用します。installer-provisioned infrastructure は、接続されたネットワーク環境または切断されたネットワーク環境でのインストールに最適です。installer-provisioned infrastructure には、クラスターの基礎となるインフラストラクチャーをプロビジョニングするインストールプログラムが含まれています。

**Assisted Installer の使用:** Assisted Installer は [console.redhat.com](https://console.redhat.com) でホストされます。Assisted Installer は、切断された環境では使用できません。Assisted Installer はクラスターの基礎となるインフラストラクチャーをプロビジョニングしないため、Assisted Installer を実行する前にインフラストラクチャーをプロビジョニングする必要があります。Assisted Installer を使用してインストールすると、Nutanix との統合も提供され、自動スケーリングが可能になります。詳細は、Assisted Installer を使用したオンプレミスクラスターのインストールを参照してください。

**user-provisioned infrastructure の使用:** [任意のプラットフォームへのクラスターのインストール](#) ドキュメントに概説されている関連手順を完了します。

### 2.1. 前提条件

- [OpenShift Container Platform のインストールおよび更新](#) プロセスの詳細を確認した。
  - インストールプログラムで、Prism Central および Prism Element のポート 9440 にアクセスできる。ポート 9440 にアクセスできることを確認している。
  - ファイアウォールを使用している場合は、次の前提条件を満たしています。
    - ポート 9440 にアクセスできることを確認している。コントロールプレーンノードがポート 9440 で Prism Central および Prism Element にアクセスできる（インストールが成功するためには必要）。
    - OpenShift Container Platform が必要とするサイトへの [アクセスを許可する](#) ようにファイアウォールを設定している。これには、Telemetry の使用が含まれます。
  - Nutanix 環境でデフォルトの自己署名 SSL 証明書を使用している場合は、CA によって署名された証明書に置き換える。インストールプログラムには、Prism Central API にアクセスするための有効な CA 署名付き証明書が必要です。自己署名証明書の置き換えに関する詳細は、[Nutanix AOS Security Guide](#) を参照してください。
- Nutanix 環境で内部 CA を使用して証明書を発行する場合は、インストールプロセスの一部としてクラスター全体のプロキシーを設定する必要があります。詳細は、[カスタム PKI の設定](#) を参照してください。



#### 重要

2048 ビット証明書を使用します。Prism Central 2022.x で 4096 ビット証明書を使用すると、インストールに失敗します。

### 2.2. OPENSHIFT CONTAINER PLATFORM のインターネットアクセス

OpenShift Container Platform 4.14 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。クラスターがインターネットにアクセスでき、Telemetry を無効にしていない場合、そのサービスによってクラスターのサブスクリプションが自動的に有効化されます。
- 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インストールパッケージを設定します。インストールタイプに応じて、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。

## 2.3. PRISM CENTRAL のインターネットアクセス

Prism Central では、クラスターのインストールに必要な Red Hat Enterprise Linux CoreOS (RHCOS) イメージを取得するためにインターネットアクセスが必要です。Nutanix の RHCOS イメージは、[rhcoss.mirror.openshift.com](http://rhcoss.mirror.openshift.com) で入手できます。

## 2.4. クラスターノードの SSH アクセス用のキーペアの生成

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できます。キーは、Ignition 設定ファイルを介して Red Hat Enterprise Linux CoreOS (RHCOS) ノードに渡され、ノードへの SSH アクセスを認証するために使用されます。このキーは各ノードの `core` ユーザーの `~/.ssh/authorized_keys` リストに追加され、パスワードなしの認証が可能になります。

鍵がノードに渡されたら、鍵ペアを使用して、`core` ユーザーとして RHCOS ノードに SSH 接続できます。SSH 経由でノードにアクセスするには、秘密鍵のアイデンティティーをローカルユーザーの SSH で管理する必要があります。

インストールのデバッグまたは障害復旧を実行するためにクラスターノードに対して SSH を実行する場合は、インストールプロセスの間に SSH 公開鍵を指定する必要があります。`./openshift-install gather` コマンドでは、SSH 公開鍵がクラスターノードに配置されている必要があります。



### 重要

障害復旧およびデバッグが必要な実稼働環境では、この手順を省略しないでください。



### 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. クラスターノードへの認証に使用するローカルマシンに既存の SSH キーペアがない場合は、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 新しい SSH キーのパスとファイル名 (~/.ssh/id\_ed25519 など) 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。



#### 注記

**x86\_64**、**ppc64le**、および **s390x** アーキテクチャーのみで FIPS 140-2/140-3 検証のために NIST に提出された RHEL 暗号化ライブラリーを使用する OpenShift Container Platform クラスターをインストールする予定がある場合は、**ed25519** アルゴリズムを使用するキーを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 公開鍵を表示します。

```
$ cat <path>/<file_name>.pub
```

たとえば、次のコマンドを実行して ~/.ssh/id\_ed25519.pub 公開鍵を表示します。

```
$ cat ~/.ssh/id_ed25519.pub
```

3. ローカルユーザーの SSH エージェントに SSH 秘密鍵 ID が追加されていない場合は、それを追加します。キーの SSH エージェント管理は、クラスターノードへのパスワードなしの SSH 認証、または **./openshift-install gather** コマンドを使用する場合は必要になります。



#### 注記

一部のディストリビューションでは、~/.ssh/id\_rsa および ~/.ssh/id\_dsa などのデフォルトの SSH 秘密鍵のアイデンティティーは自動的に管理されます。

- a. **ssh-agent** プロセスがローカルユーザーに対して実行されていない場合は、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。

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

\$ ssh-add <path>/<file\_name> ①

- ① `~/.ssh/id_ed25519` 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。

## 出力例

Identity added: /home/<you>/<path>/<file\_name> (<computer\_name>)

## 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。

## 2.5. 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、インストールに使用しているホストにインストールファイルをダウンロードします。

### 前提条件

- Linux または macOS を実行し、少なくとも 1.2 GB のローカルディスク容量を備えたコンピューターがある。

### 手順

- Red Hat Hybrid Cloud Console の [Cluster Type](#) ページに移動します。Red Hat アカウントがある場合は、認証情報を使用してログインします。アカウントがない場合はこれを作成します。
- ページの **Run it yourself** セクションからインフラストラクチャープロバイダーを選択します。
- OpenShift Installer** のドロップダウンメニューからホストオペレーティングシステムとアーキテクチャーを選択し、**Download Installer** をクリックします。
- ダウンロードしたファイルを、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。クラスターを削除するには、両方のファイルが必要です。
- 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。

- 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
```

- 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ー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。

## ヒント

Red Hat カスタマー ポータル からインストール プログラムを取得することもできます。このページでは、ダウンロードするインストール プログラムのバージョンを指定できます。ただし、このページにアクセスするには、有効なサブスクリプションが必要です。

## 2.6. NUTANIX ROOT CA 証明書をシステム信頼に追加する

インストール プログラムは Prism Central API へのアクセスを必要とするため、OpenShift Container Platform クラスターをインストールする前に、Nutanix の信頼された root CA 証明書をシステム信頼に追加する必要があります。

### 手順

- Prism Central Web コンソールから、Nutanix root CA 証明書をダウンロードします。
- Nutanix root CA 証明書を含む圧縮ファイルを抽出します。
- オペレーティングシステム用のファイルをシステム信頼に追加します。たとえば、Fedora オペレーティングシステムで以下のコマンドを実行します。

```
# cp certs/lin/* /etc/pki/ca-trust/source/anchors
```

- システム信頼を更新します。たとえば、Fedora オペレーティングシステムで以下のコマンドを実行します。

```
# update-ca-trust extract
```

## 2.7. インストール設定ファイルの作成

Nutanix 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 がある。
- Nutanix のネットワーク要件を満たしていることが確認されました。詳細は、「Nutanix へのインストールの準備」を参照してください。

### 手順

- 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> ①
```

- 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ディレクトリ名を指定します。

ディレクトリーを指定する場合:

- ディレクトリーに **execute** 権限があることを確認します。この権限は、インストールディレクトリーで Terraform バイナリーを実行するために必要です。
- 空のディレクトリーを使用します。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は有効期限が短いため、インストールディレクトリーを再利用しないでください。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ください。

#### 注記

古い設定の再利用を回避するために、`~/.powervs` ディレクトリーは必ず削除してください。以下のコマンドを実行します。

```
$ rm -rf ~/.powervs
```

- b. プロンプト時に、クラウドの設定の詳細情報を指定します。

- i. オプション: クラスターマシンにアクセスするために使用する SSH キーを選択します。



#### 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のある実稼働用の OpenShift Container Platform クラスターでは、**ssh-agent** プロセスが使用する SSH キーを指定します。

- ii. ターゲットとするプラットフォームとして `nutanix` を選択します。
- iii. Prism Central のドメイン名または IP アドレスを入力します。
- iv. Prism Central へのログインに使用するポートを入力します。
- v. Prism Central へのログインに使用する認証情報を入力します。  
インストールプログラムが Prism Central に接続します。
- vi. OpenShift Container Platform クラスターを管理する Prism Element を選択します。
- vii. 使用するネットワークサブネットを選択します。
- viii. コントロールプレーン API のアクセス用に設定した仮想 IP アドレスを入力します。
- ix. クラスター Ingress に設定した仮想 IP アドレスを入力します。

- x. ベースドメインを入力します。このベースドメインは、DNS レコードで設定したものと同じである必要があります。
  - xi. クラスターの記述名を入力します。  
入力するクラスター名は、DNS レコードの設定時に指定したクラスター名と一致する必要があります。
2. オプション: **install.config.yaml** ファイル内の 1つ以上のデフォルト設定パラメーターを更新して、インストールをカスタマイズします。  
パラメーターの詳細は、「インストール設定パラメーター」を参照してください。



### 注記

3 ノードクラスターをインストールする場合は、必ず **compute.replicas** パラメーターを **0** に設定してください。これにより、クラスターのコントロールプレーンがスケジュール可能になります。詳細は、「Nutanix への 3 ノードクラスターのインストール」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

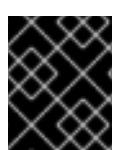
**install-config.yaml** ファイルはインストールプロセス時に使用されます。このファイルを再利用する必要がある場合は、この段階でこれをバックアップしてください。

## 関連情報

- Nutanix のインストール設定パラメーター

### 2.7.1. Nutanix 用にカスタマイズされた **install-config.yaml** ファイルのサンプル

**install-config.yaml** ファイルをカスタマイズして、OpenShift Container Platform クラスターのプラットフォームに関する詳細を指定するか、必要なパラメーターの値を変更することができます。



### 重要

このサンプルの YAML ファイルは参照用にのみ提供されます。インストールプログラムを使用して **install-config.yaml** ファイルを取得し、これを変更する必要があります。

```
apiVersion: v1
baseDomain: example.com ①
compute: ②
- hyperthreading: Enabled ③
  name: worker
  replicas: 3
  platform:
    nutanix: ④
    cpus: 2
    coresPerSocket: 2
    memoryMiB: 8196
    osDisk:
```

```
diskSizeGiB: 120
categories: 5
- key: <category_key_name>
  value: <category_value>
controlPlane: 6
  hyperthreading: Enabled 7
  name: master
  replicas: 3
  platform:
    nutanix: 8
    cpus: 4
    coresPerSocket: 2
    memoryMiB: 16384
    osDisk:
      diskSizeGiB: 120
      categories: 9
      - key: <category_key_name>
        value: <category_value>
metadata:
  creationTimestamp: null
  name: test-cluster 10
networking:
  clusterNetwork:
    - cidr: 10.128.0.0/14
      hostPrefix: 23
  machineNetwork:
    - cidr: 10.0.0.0/16
  networkType: OVNKubernetes 11
  serviceNetwork:
    - 172.30.0.0/16
platform:
  nutanix:
    apiVIPs:
      - 10.40.142.7 12
  defaultMachinePlatform:
    bootType: Legacy
    categories: 13
    - key: <category_key_name>
      value: <category_value>
    project: 14
      type: name
      name: <project_name>
  ingressVIPs:
    - 10.40.142.8 15
prismCentral:
  endpoint:
    address: your.prismcentral.domainname 16
    port: 9440 17
    password: <password> 18
    username: <username> 19
prismElements:
  - endpoint:
      address: your.prismelement.domainname
      port: 9440
```

```

uuid: 0005b0f1-8f43-a0f2-02b7-3cecef193712
subnetUUIDs:
- c7938dc6-7659-453e-a688-e26020c68e43
clusterOSImage: http://example.com/images/rhcos-47.83.202103221318-0-nutanix.x86_64.qcow2
20
credentialsMode: Manual
publish: External
pullSecret: '{"auths": ...}' 21
fips: false 22
sshKey: ssh-ed25519 AAAA... 23

```

**1** **10** **12** **15** **16** **17** **18** **19** **21** 必須。インストールプログラムはこの値の入力を求めるプロンプトを出します。

**2** **6** **controlPlane** セクションは單一マッピングですが、コンピュートセクションはマッピングのシーケンスになります。複数の異なるデータ構造の要件を満たすには、**compute** セクションの最初の行はハイフン - で始め、**controlPlane** セクションの最初の行はハイフンで始めることができます。どちらのセクションも、現時点では單一のマシンプールを定義しますが、OpenShift Container Platform の今後のバージョンでは、インストール時の複数のコンピュートプールの定義をサポートする可能性があります。1つのコントロールプレーンプールのみが使用されます。

**3** **7** 同時マルチスレッドまたは **hyperthreading** を有効/無効にするかどうか。デフォルトでは、同時マルチ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化されます。パラメーター値を **Disabled** に設定するとこれを無効にすることができます。一部のクラスターマシンで同時マルチスレッドを無効にする場合は、これをすべてのクラスターマシンで無効にする必要があります。



### 重要

同時マルチ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られていることを確認します。

**4** **8** オプション: コンピュートおよびコントロールプレーンマシンのマシンプールパラメーターの追加設定を指定します。

**5** **9** **13** オプション: プリズムカテゴリーとプリズムカテゴリー値のペアを1つ以上指定します。これらのカテゴリーのキーと値のペアは、Prism Central に存在する必要があります。コンピューティングマシン、コントロールプレーンマシン、またはすべてのマシンに個別のカテゴリーを指定できます。

**11** インストールするクラスターネットワークプラグイン。サポートされている値は **OVNKubernetes** と **OpenShiftSDN** です。デフォルトの値は **OVNkubernetes** です。

**14** オプション: VM が関連付けられているプロジェクトを指定します。プロジェクトタイプの **name** または **uuid** を指定してから、対応する UUID またはプロジェクト名を指定します。プロジェクトは、コンピューティングマシン、コントロールプレーンマシン、またはすべてのマシンに関連付けすることができます。

**20** オプション: デフォルトでは、インストールプログラムは Red Hat Enterprise Linux CoreOS (RHCOS) イメージをダウンロードしてインストールします。Prism Central がインターネットにアクセスできない場合は、任意の HTTP サーバーで RHCOS イメージをホストし、インストールプログラムがそのイメージを指すようにすることで、デフォルトの動作をオーバーライドできます。

**22** FIPS モードを有効または無効にするかどうか。デフォルトでは、FIPS モードは有効にされません。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。



## 重要

FIPS モードでブートされた Red Hat Enterprise Linux (RHEL) または Red Hat Enterprise Linux CoreOS (RHCOS) を実行する場合、OpenShift Container Platform コアコンポーネントは、x86\_64、ppc64le、および s390x アーキテクチャーのみで、FIPS 140-2/140-3 検証のために NIST に提出された RHEL 暗号化ライブラリーを使用します。

- 23 オプション: クラスター内のマシンにアクセスするのに使用する **sshKey** 値をオプションで指定できます。



## 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のある実稼働用の OpenShift Container Platform クラスターでは、**ssh-agent** プロセスが使用する SSH キーを指定します。

### 2.7.2. インストール時のクラスター全体のプロキシーの設定

実稼働環境では、インターネットへの直接アクセスを拒否し、代わりに HTTP または HTTPS プロキシーを使用することができます。プロキシー設定を **install-config.yaml** ファイルで行うことにより、新規の OpenShift Container Platform クラスターをプロキシーを使用するように設定できます。

#### 前提条件

- 既存の **install-config.yaml** ファイルがある。
- クラスターがアクセスする必要のあるサイトを確認済みで、それらのいずれかがプロキシーをバイパスする必要があるかどうかを判別している。デフォルトで、すべてのクラスター Egress トラフィック (クラスターをホストするクラウドに関するクラウドプロバイダー API に対する呼び出しを含む) はプロキシーされます。プロキシーを必要に応じてバイパスするために、サイトを **Proxy** オブジェクトの **spec.noProxy** フィールドに追加している。



## 注記

**Proxy** オブジェクトの **status.noProxy** フィールドには、インストール設定の **networking.machineNetwork[].cidr**、**networking.clusterNetwork[].cidr**、および **networking.serviceNetwork[]** フィールドの値が設定されます。

Amazon Web Services (AWS)、Google Cloud、Microsoft Azure、および Red Hat OpenStack Platform (RHOSP)へのインストールの場合、**Proxy** オブジェクトの **status.noProxy** フィールドには、インスタンスマタデータのエンドポイント (**169.254.169.254**) も設定されます。

#### 手順

- install-config.yaml** ファイルを編集し、プロキシー設定を追加します。以下に例を示します。

```
apiVersion: v1
```

```

baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> ①
  httpsProxy: https://<username>:<pswd>@<ip>:<port> ②
  noProxy: example.com ③
  additionalTrustBundle: | ④
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> ⑤

```

- ① クラスター外の HTTP 接続を作成するために使用するプロキシー URL。URL スキームは **http** である必要があります。
- ② クラスター外で HTTPS 接続を作成するために使用するプロキシー URL。
- ③ プロキシーから除外するための宛先ドメイン名、IP アドレス、または他のネットワーク CIDR のコンマ区切りのリスト。サブドメインのみと一致するように、ドメインの前に . を付けます。たとえば、**.y.com** は **x.y.com** に一致しますが、**y.com** には一致しません。\* を使用し、すべての宛先のプロキシーをバイパスします。
- ④ 指定されている場合、インストールプログラムは HTTPS 接続のプロキシーに必要な 1 つ以上の追加の CA 証明書が含まれる **user-ca-bundle** という名前の設定マップを **openshift-config** namespace に生成します。次に Cluster Network Operator は、これらのコンテンツを Red Hat Enterprise Linux CoreOS (RHCOS) 信頼バンドルにマージする **trusted-ca-bundle** 設定マップを作成し、この設定マップは **Proxy** オブジェクトの **trustedCA** フィールドで参照されます。**additionalTrustBundle** フィールドは、プロキシーのアイデンティティー証明書が RHCOS 信頼バンドルからの認証局によって署名されない限り必要になります。
- ⑤ オプション: **trustedCA** フィールドの **user-ca-bundle** 設定マップを参照する **Proxy** オブジェクトの設定を決定するポリシー。許可される値は **Proxyonly** および **Always** です。**Proxyonly** を使用して、**http/https** プロキシーが設定されている場合にのみ **user-ca-bundle** 設定マップを参照します。**Always** を使用して、常に **user-ca-bundle** 設定マップを参照します。デフォルト値は **Proxyonly** です。



### 注記

インストールプログラムは、プロキシーの **readinessEndpoints** フィールドをサポートしません。



### 注記

インストーラーがタイムアウトした場合は、インストーラーの **wait-for** コマンドを使用してデプロイメントを再起動してからデプロイメントを完了します。以下に例を示します。

```
$ ./openshift-install wait-for install-complete --log-level debug
```

2. ファイルを保存し、OpenShift Container Platform のインストール時にこれを参照します。

インストールプログラムは、指定の **install-config.yaml** ファイルのプロキシー設定を使用する **cluster** という名前のクラスター全体のプロキシーを作成します。プロキシー設定が指定されていない場合、**cluster Proxy** オブジェクトが依然として作成されますが、これには **spec** がありません。



### 注記

**cluster** という名前の **Proxy** オブジェクトのみがサポートされ、追加のプロキシーを作成することはできません。

## 2.8. 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 4.14 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。

### 2.8.1. 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の [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Product Variant** ドロップダウンリストからアーキテクチャーを選択します。
3. **バージョン** ドロップダウンリストから適切なバージョンを選択します。
4. **OpenShift v4.14 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。
5. アーカイブを展開します。

```
$ tar xvf <file>
```

6. **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
```

#### 検証

- 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>
```

## 2.8.2. 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 ポータルの [OpenShift Container Platform ダウンロード ページ](#) に移動します。
2. バージョン ドロップダウンリストから適切なバージョンを選択します。
3. **OpenShift v4.14 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展開します。
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
```

### 検証

- 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>
```

## 2.8.3. 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 ポータルの [OpenShift Container Platform ダウンロード ページ](#) に移動します。
2. バージョン ドロップダウンリストから適切なバージョンを選択します。
3. **OpenShift v4.14 macOS 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。



### 注記

macOS arm64 の場合は、**OpenShift v4.14 macOS arm64 Client** エントリーを選択します。

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
```

## 検証

- **oc** コマンドを使用してインストールを確認します。

```
$ oc <command>
```

## 2.9. NUTANIX の IAM の設定

クラスターをインストールするには、Cloud Credential Operator (CCO) が手動モードで動作する必要があります。インストールプログラムが手動モード用に CCO を設定する間、ID およびアクセス管理シークレットを指定する必要があります。

### 前提条件

- **ccctl** バイナリーを設定している。
- **install-config.yaml** ファイルがある。

### 手順

- 1 認証情報データを含む YAML ファイルを次の形式で作成します。

#### 認証情報データの形式

```
credentials:
- type: basic_auth ①
  data:
    prismCentral: ②
      username: <username_for_prism_central>
      password: <password_for_prism_central>
    prismElements: ③
      - name: <name_of_prism_element>
        username: <username_for_prism_element>
        password: <password_for_prism_element>
```

- 1 認証タイプを指定します。Basic 認証のみがサポートされています。
- 2 Prism Central の認証情報を指定します。
- 3 オプション: Prism Element 認証情報を指定します。

- 2 次のコマンドを実行して、インストールファイルのリリースイメージを **\$RELEASE\_IMAGE** 変数に設定します。

```
$ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
```

- 3 以下のコマンドを実行して、OpenShift Container Platform リリースイメージから **CredentialsRequest** カスタムリソース (CR) のリストを抽出します。

```
$ oc adm release extract \
--from=$RELEASE_IMAGE \
--credentials-requests \
```

```
--included \①
--install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \②
--to=<path_to_directory_for_credentials_requests> ③
```

- ① **--included** パラメーターには、特定のクラスター設定に必要なマニフェストのみが含まれます。
- ② **install-config.yaml** ファイルの場所を指定します。
- ③ **CredentialsRequest** オブジェクトを保存するディレクトリーへのパスを指定します。指定したディレクトリーが存在しない場合は、このコマンドによって作成されます。

### サンプル CredentialsRequest オブジェクト

```
apiVersion: cloudcredential.openshift.io/v1
kind: CredentialsRequest
metadata:
  annotations:
    include.release.openshift.io/self-managed-high-availability: "true"
  labels:
    controller-tools.k8s.io: "1.0"
  name: openshift-machine-api-nutanix
  namespace: openshift-cloud-credential-operator
spec:
  providerSpec:
    apiVersion: cloudcredential.openshift.io/v1
    kind: NutanixProviderSpec
  secretRef:
    name: nutanix-credentials
    namespace: openshift-machine-api
```

4. 次のコマンドを実行し、**ccctl** ツールを使用して **CredentialsRequest** オブジェクトをすべて処理します。

```
$ ccctl nutanix create-shared-secrets \
--credentials-requests-dir=<path_to_credentials_requests_directory> \①
--output-dir=<ccctl_output_dir> \②
--credentials-sourcefilepath=<path_to_credentials_file> ③
```

- ① コンポーネント **CredentialsRequests** オブジェクトのファイルを含むディレクトリーへのパスを指定します。
- ② オプション: **ccctl** ユーティリティーがオブジェクトを作成するディレクトリーを指定します。デフォルトでは、ユーティリティーは、コマンドが実行されるディレクトリーにオブジェクトを作成します。
- ③ オプション: 認証情報データ YAML ファイルを含むディレクトリーを指定します。デフォルトでは、**ccctl** はこのファイルが `<home_directory>/.nutanix/credentials` にあると想定します。

5. **credentialsMode** パラメーターが **Manual** に設定されるように、**install-config.yaml** 設定ファイルを編集します。

## サンプル `install-config.yaml` 設定ファイル

```
apiVersion: v1
baseDomain: cluster1.example.com
credentialsMode: Manual ①
...
...
```

- ① この行を追加して、`credentialsMode` パラメーターを **Manual** に設定します。

6. 次のコマンドを実行して、インストールマニフェストを作成します。

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

- ① クラスターの `install-config.yaml` ファイルを含むディレクトリーへのパスを指定します。

7. 次のコマンドを実行して、生成された認証情報ファイルをターゲットマニフェストディレクトリーにコピーします。

```
$ cp <ccctl_output_dir>/manifests/*credentials.yaml ./<installation_directory>/manifests
```

## 検証

- `manifests` ディレクトリーに適切なシークレットが存在することを確認します。

```
$ ls ./<installation_directory>/manifests
```

## 出力例

```
cluster-config.yaml
cluster-dns-02-config.yaml
cluster-infrastructure-02-config.yaml
cluster-ingress-02-config.yaml
cluster-network-01-crd.yaml
cluster-network-02-config.yaml
cluster-proxy-01-config.yaml
cluster-scheduler-02-config.yaml
cvo-overrides.yaml
kube-cloud-config.yaml
kube-system-configmap-root-ca.yaml
machine-config-server-tls-secret.yaml
openshift-config-secret-pull-secret.yaml
openshift-cloud-controller-manager-nutanix-credentials-credentials.yaml
openshift-machine-api-nutanix-credentials-credentials.yaml
```

## 2.10. NUTANIX CCM に必要な CONFIG MAP とシークレットリソースの追加

Nutanix にインストールするには、Nutanix Cloud Controller Manager (CCM) と統合するために追加の **ConfigMap** および **Secret** リソースが必要です。

## 前提条件

- インストールディレクトリー内に **manifests** ディレクトリーが作成されました。

## 手順

- manifests** ディレクトリーに移動します。

```
$ cd <path_to_installation_directory>/manifests
```

- openshift-cloud-controller-manager-cloud-config.yaml** という名前で **cloud-conf ConfigMap** ファイルを作成し、以下の情報を追加します。

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: cloud-conf
  namespace: openshift-cloud-controller-manager
data:
  cloud.conf: |
    {"prismCentral": {
      "address": "<prism_central_FQDN/IP>", ①
      "port": 9440,
      "credentialRef": {
        "kind": "Secret",
        "name": "nutanix-credentials",
        "namespace": "openshift-cloud-controller-manager"
      }
    },
    "topologyDiscovery": {
      "type": "Prism",
      "topologyCategories": null
    },
    "enableCustomLabeling": true
  }
```

- ① Prism Central FQDN/IP を指定します。

- ファイル **cluster-infrastructure-02-config.yml** が存在し、次の情報が含まれていることを確認します。

```
spec:
  cloudConfig:
    key: config
    name: cloud-provider-config
```

## 2.11. 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

- 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がある。
- ホスト上のクラウドプロバイダーアカウントに、クラスターをデプロイするための正しい権限があることを確認した。アカウントの権限が正しくないと、インストールプロセスが失敗し、不足している権限を示すエラーメッセージが表示されます。

### 手順

- 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>** に、カスタマイズした **./install-config.yaml** ファイルの場所を指定します。
- 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。

### 検証

クラスターのデプロイが正常に完了すると、次のようにになります。

- ターミナルには、Web コンソールへのリンクや **kubeadmin** ユーザーの認証情報など、クラスターにアクセスするための指示が表示されます。
- 認証情報は **<installation\_directory>/openshift\_install.log** にも出力されます。



### 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。

### 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: "password"
INFO Time elapsed: 36m22s
```



## 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](#)に関する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推奨します。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。

## 2.12. デフォルトのストレージコンテナーの設定

クラスターをインストールしたら、Nutanix CSI Operator をインストールし、クラスターのデフォルトのストレージコンテナーを設定する必要があります。

詳細は、[CSI Operator のインストール と レジストリーストレージの設定](#)に関する Nutanix のドキュメントを参照してください。

## 2.13. OPENSHIFT CONTAINER PLATFORM の TELEMETRY アクセス

OpenShift Container Platform 4.14 では、クラスターの健全性および正常に実行された更新についての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 サービスにも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](#) インベントリーが正常である (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。

## 2.14. 関連情報

- [リモートヘルスモニタリングについて](#)

## 2.15. 次のステップ

- [リモートヘルスレポート](#)
- [クラスターのカスタマイズ](#)

## 第3章 ネットワークが制限された環境での NUTANIX へのクラスターのインストール

OpenShift Container Platform 4.14 では、インストールリリースコンテンツの内部ミラーを作成すると、制限されたネットワーク内の Nutanix インフラストラクチャーにクラスターをインストールできます。

### 3.1. 前提条件

- [OpenShift Container Platform のインストールおよび更新](#) プロセスの詳細を確認した。
- インストールプログラムで、Prism Central および Prism Element のポート 9440 にアクセスできる。ポート 9440 にアクセスできることを確認している。
- ファイアウォールを使用している場合は、次の前提条件を満たしています。
  - ポート 9440 にアクセスできることを確認している。コントロールプレーンノードがポート 9440 で Prism Central および Prism Element にアクセスできる（インストールが成功するためには必要）。
  - OpenShift Container Platform が必要とするサイトへの [アクセスを許可する](#) ようにファイアウォールを設定している。これには、Telemetry の使用が含まれます。
- Nutanix 環境でデフォルトの自己署名 SSL/TLS 証明書を使用している場合は、CA によって署名された証明書に置き換える。インストールプログラムには、Prism Central API にアクセスするための有効な CA 署名付き証明書が必要です。自己署名証明書の置き換えに関する詳細は、[Nutanix AOS Security Guide](#) を参照してください。  
Nutanix 環境で内部 CA を使用して証明書を発行する場合は、インストールプロセスの一部としてクラスター全体のプロキシーを設定する必要があります。詳細は、[カスタム PKI の設定](#) を参照してください。



#### 重要

2048 ビット証明書を使用します。Prism Central 2022.x で 4096 ビット証明書を使用すると、インストールに失敗します。

- Red Hat Quay などのコンテナイメージレジストリーがある。レジストリーがまだない場合は、[Red Hat OpenShift のミラーレジストリー](#) を使用してミラーレジストリーを作成できます。
- [oc-mirror OpenShift CLI \(oc\) プラグイン](#) を使用して、必要なすべての OpenShift Container Platform コンテンツと、Nutanix CSI Operator を含むその他のイメージをミラーレジストリーにミラーリングした。



#### 重要

インストールメディアはミラーホストにあるため、そのコンピューターを使用してすべてのインストール手順を完了することができます。

### 3.2. ネットワークが制限された環境でのインストールについて

OpenShift Container Platform 4.14 では、ソフトウェアコンポーネントを取得するためにインターネットへのアクティブな接続を必要としないインストールを実行できます。ネットワークが制限された環境

のインストールは、クラスターのインストール先となるクラウドプラットフォームに応じて、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また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を使用して実行できます。

クラウドプラットフォーム上でネットワークが制限されたインストールの実行を選択した場合でも、そのクラウド API へのアクセスが必要になります。Amazon Web Service の Route 53 DNS や IAM サービスなどの一部のクラウド機能には、インターネットアクセスが必要です。ネットワークによっては、ベアメタルハードウェア、Nutanix、または VMware vSphere へのインストールに必要なインターネットアクセスが少なくて済む場合があります。

ネットワークが制限されたインストールを完了するには、OpenShift イメージレジストリーのコンテンツをミラーリングし、インストールメディアを含むレジストリーを作成する必要があります。このミラーは、インターネットと制限されたネットワークの両方にアクセスできるミラーホストで、または制限に対応する他の方法を使用して作成できます。

### 3.2.1. その他の制限

ネットワークが制限された環境のクラスターには、以下の追加の制限および制約があります。

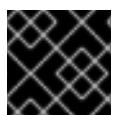
- **ClusterVersion** ステータスには **Unable to retrieve available updates** エラーが含まれます。
- デフォルトでは、必要なイメージストリームタグにアクセスできないため、開発者カタログのコンテンツは使用できません。

### 3.3. クラスターノードの SSH アクセス用のキーペアの生成

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できます。キーは、Ignition 設定ファイルを介して Red Hat Enterprise Linux CoreOS (RHCOS) ノードに渡され、ノードへの SSH アクセスを認証するために使用されます。このキーは各ノードの **core** ユーザーの `~/.ssh/authorized_keys` リストに追加され、パスワードなしの認証が可能になります。

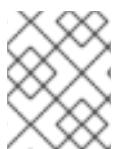
鍵がノードに渡されたら、鍵ペアを使用して、**core** ユーザーとして RHCOS ノードに SSH 接続できます。SSH 経由でノードにアクセスするには、秘密鍵のアイデンティティーをローカルユーザーの SSH で管理する必要があります。

インストールのデバッグまたは障害復旧を実行するためにクラスターノードに対して SSH を実行する場合は、インストールプロセスの間に SSH 公開鍵を指定する必要があります。`./openshift-install gather` コマンドでは、SSH 公開鍵がクラスターノードに配置されている必要があります。



#### 重要

障害復旧およびデバッグが必要な実稼働環境では、この手順を省略しないでください。



#### 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. クラスターノードへの認証に使用するローカルマシンに既存の SSH キーペアがない場合は、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
```

- 新しい SSH キーのパスとファイル名 (~/.ssh/id\_ed25519 など) 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。



#### 注記

**x86\_64**、**ppc64le**、および **s390x** アーキテクチャーのみで FIPS 140-2/140-3 検証のために NIST に提出された RHEL 暗号化ライブラリーを使用する OpenShift Container Platform クラスターをインストールする予定がある場合は、**ed25519** アルゴリズムを使用するキーを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。

- SSH 公開鍵を表示します。

```
$ cat <path>/<file_name>.pub
```

たとえば、次のコマンドを実行して **~/.ssh/id\_ed25519.pub** 公開鍵を表示します。

```
$ cat ~/.ssh/id_ed25519.pub
```

- ローカルユーザーの SSH エージェントに SSH 秘密鍵 ID が追加されていない場合は、それを追加します。キーの SSH エージェント管理は、クラスターノードへのパスワードなしの SSH 認証、または **./openshift-install gather** コマンドを使用する場合は必要になります。



#### 注記

一部のディストリビューションでは、**~/.ssh/id\_rsa** および **~/.ssh/id\_dsa** などのデフォルトの SSH 秘密鍵のアイデンティティーは自動的に管理されます。

- ssh-agent** プロセスがローカルユーザーに対して実行されていない場合は、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。

- SSH プライベートキーを **ssh-agent** に追加します。

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

- 1 ~/.ssh/id\_ed25519 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。

## 出力例

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

## 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。

## 3.4. NUTANIX ROOT CA 証明書をシステム信頼に追加する

インストールプログラムは Prism Central API へのアクセスを必要とするため、OpenShift Container Platform クラスターをインストールする前に、Nutanix の信頼された root CA 証明書をシステム信頼に追加する必要があります。

### 手順

- Prism Central Web コンソールから、Nutanix root CA 証明書をダウンロードします。
- Nutanix root CA 証明書を含む圧縮ファイルを抽出します。
- オペレーティングシステム用のファイルをシステム信頼に追加します。たとえば、Fedora オペレーティングシステムで以下のコマンドを実行します。

```
# cp certs/lin/* /etc/pki/ca-trust/source/anchors
```

- システム信頼を更新します。たとえば、Fedora オペレーティングシステムで以下のコマンドを実行します。

```
# update-ca-trust extract
```

## 3.5. RHCOS クラスターイメージのダウンロード

Prism Central は、クラスターをインストールするために Red Hat Enterprise Linux CoreOS (RHCOS) イメージにアクセスする必要があります。インストールプログラムを使用して、RHCOS イメージを見つけてダウンロードし、内部 HTTP サーバーまたは Nutanix オブジェクトを介して利用できるようにすることができます。

### 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する。ネットワークが制限されたインストールでは、これらのファイルがミラーホスト上に置かれます。

### 手順

- 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 coreos print-stream-json
```

2. コマンドの出力を使用して Nutanix イメージの場所を見つけ、リンクをクリックしてダウンロードします。

## 出力例

```
"nutanix": {
  "release": "411.86.202210041459-0",
  "formats": {
    "qcow2": {
      "disk": {
        "location": "https://rhcos.mirror.openshift.com/art/storage/releases/rhcos-4.11/411.86.202210041459-0/x86_64/rhcos-411.86.202210041459-0-nutanix.x86_64.qcow2",
        "sha256": "42e227cac6f11ac37ee8a2f9528bb3665146566890577fd55f9b950949e5a54b"
      }
    }
  }
}
```

3. 内部 HTTP サーバーまたは Nutanix オブジェクトを介してイメージを利用できるようにします。
4. ダウンロードしたイメージの場所に注意してください。クラスターをデプロイする前に、インストール設定ファイル (`install-config.yaml`) の **platform** セクションをイメージの場所で更新します。

## RHCOS イメージを指定する `install-config.yaml` ファイルのスニペット

```
■ platform:
  nutanix:
    clusterOSImage: http://example.com/images/rhcos-411.86.202210041459-0-nutanix.x86_64.qcow2
```

## 3.6. インストール設定ファイルの作成

Nutanix 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がある。ネットワークが制限されたインストールでは、これらのファイルがミラーホスト上に置かれます。
- レジストリーをミラーリングしたときに作成された `imageContentSourcePolicy.yaml` ファイルを用意する。
- ダウンロードした Red Hat Enterprise Linux CoreOS (RHCOS) イメージの場所を用意する。
- ミラーレジストリーの証明書の内容を取得している。
- Red Hat Enterprise Linux CoreOS (RHCOS) イメージを取得し、アクセス可能な場所にアップロードしました。
- Nutanix のネットワーク要件を満たしていることが確認されました。詳細は、「Nutanixへのインストールの準備」を参照してください。

## 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> ①
```

- 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名を指定します。

ディレクトリーを指定する場合:

- ディレクトリーに **execute** 権限があることを確認します。この権限は、インストールディレクトリーで Terraform バイナリーを実行するために必要です。
- 空のディレクトリーを使用します。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は有効期限が短いため、インストールディレクトリーを再利用しないでください。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ください。

#### 注記

古い設定の再利用を回避するために、`~/.powervs` ディレクトリーは必ず削除してください。以下のコマンドを実行します。

```
$ rm -rf ~/.powervs
```

- プロンプト時に、クラウドの設定の詳細情報を指定します。

- オプション: クラスターマシンにアクセスするために使用する SSH キーを選択します。

#### 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のある実稼働用の OpenShift Container Platform クラスターでは、**ssh-agent** プロセスが使用する SSH キーを指定します。

- ターゲットとするプラットフォームとして `nutanix` を選択します。
- Prism Central のドメイン名または IP アドレスを入力します。
- Prism Central へのログインに使用するポートを入力します。
- Prism Central へのログインに使用する認証情報を入力します。  
インストールプログラムが Prism Central に接続します。
- OpenShift Container Platform クラスターを管理する Prism Element を選択します。

- vii. 使用するネットワークサブネットを選択します。
- viii. コントロールプレーン API のアクセス用に設定した仮想 IP アドレスを入力します。
- ix. クラスター Ingress に設定した仮想 IP アドレスを入力します。
- x. ベースドメインを入力します。このベースドメインは、DNS レコードで設定したものと同じである必要があります。
- xi. クラスターの記述名を入力します。  
入力するクラスター名は、DNS レコードの設定時に指定したクラスター名と一致する必要があります。

2. **install-config.yaml** ファイルで、**platform.nutanix.clusterOSImage** の値をイメージの場所または名前に設定します。以下に例を示します。

```
platform:
  nutanix:
    clusterOSImage: http://mirror.example.com/images/rhcos-47.83.202103221318-0-nutanix.x86_64.qcow2
```

3. **install-config.yaml** ファイルを編集し、ネットワークが制限された環境でのインストールに必要な追加の情報を提供します。

- a. **pullSecret** の値を更新して、レジストリーの認証情報を追加します。

```
pullSecret: '{"auths":{"<mirror_host_name>:5000": {"auth": "<credentials>","email": "you@example.com"}}}'
```

**<mirror\_host\_name>** の場合、ミラーレジストリーの証明書で指定したレジストリードメイン名を指定し、**<credentials>** の場合は、ミラーレジストリーの base64 でエンコードされたユーザー名およびパスワードを指定します。

- b. **additionalTrustBundle** パラメーターおよび値を追加します。

```
additionalTrustBundle: |
  -----BEGIN CERTIFICATE-----
  ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
  -----END CERTIFICATE-----
```

この値は、ミラーレジストリーに使用した証明書ファイルの内容である必要があります。証明書ファイルは、既存の信頼できる認証局、またはミラーレジストリー用に生成した自己署名証明書のいずれかです。

- c. 次の YAML の抜粋のようなイメージコンテンツリソースを追加します。

```
imageContentSources:
- mirrors:
  - <mirror_host_name>:5000/<repo_name>/release
    source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - <mirror_host_name>:5000/<repo_name>/release
    source: registry.redhat.io/ocp/release
```

これらの値には、レジストリーをミラーリングしたときに作成された **imageContentSourcePolicy.yaml** ファイルを使用します。

- オプション: **install.config.yaml** ファイル内の1つ以上のデフォルト設定パラメーターを更新して、インストールをカスタマイズします。  
パラメーターの詳細は、「インストール設定パラメーター」を参照してください。



### 注記

3ノードクラスターをインストールする場合は、必ず **compute.replicas** パラメーターを **0** に設定してください。これにより、クラスターのコントロールプレーンがスケジュール可能になります。詳細は、「**{platform}** への3ノードクラスターのインストール」を参照してください。

- 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

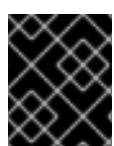
**install-config.yaml** ファイルはインストールプロセス時に使用されます。このファイルを再利用する必要がある場合は、この段階でこれをバックアップしてください。

## 関連情報

- [Nutanix のインストール設定パラメーター](#)

### 3.6.1. Nutanix 用にカスタマイズされた **install-config.yaml** ファイルのサンプル

**install-config.yaml** ファイルをカスタマイズして、OpenShift Container Platform クラスターのプラットフォームに関する詳細を指定するか、必要なパラメーターの値を変更することができます。



### 重要

このサンプルの YAML ファイルは参照用にのみ提供されます。インストールプログラムを使用して **install-config.yaml** ファイルを取得し、これを変更する必要があります。

```
apiVersion: v1
baseDomain: example.com 1
compute: 2
- hyperthreading: Enabled 3
  name: worker
  replicas: 3
  platform:
    nutanix: 4
    cpus: 2
    coresPerSocket: 2
    memoryMiB: 8196
    osDisk:
      diskSizeGiB: 120
    categories: 5
    - key: <category_key_name>
```

```
  value: <category_value>
controlPlane: 6
  hyperthreading: Enabled 7
  name: master
  replicas: 3
  platform:
    nutanix: 8
    cpus: 4
    coresPerSocket: 2
    memoryMiB: 16384
    osDisk:
      diskSizeGiB: 120
    categories: 9
      - key: <category_key_name>
        value: <category_value>
metadata:
  creationTimestamp: null
  name: test-cluster 10
networking:
  clusterNetwork:
    - cidr: 10.128.0.0/14
      hostPrefix: 23
  machineNetwork:
    - cidr: 10.0.0.0/16
  networkType: OVNKubernetes 11
  serviceNetwork:
    - 172.30.0.0/16
platform:
  nutanix:
    apiVIP: 10.40.142.7 12
    ingressVIP: 10.40.142.8 13
    defaultMachinePlatform:
      bootType: Legacy
      categories: 14
        - key: <category_key_name>
          value: <category_value>
      project: 15
        type: name
        name: <project_name>
    prismCentral:
      endpoint:
        address: your.prismcentral.domainname 16
        port: 9440 17
        password: <password> 18
        username: <username> 19
    prismElements:
      - endpoint:
          address: your.prismelement.domainname
          port: 9440
          uuid: 0005b0f1-8f43-a0f2-02b7-3cecef193712
    subnetUUIDs:
      - c7938dc6-7659-453e-a688-e26020c68e43
    clusterOSImage: http://example.com/images/rhcos-47.83.202103221318-0-nutanix.x86_64.qcow2
```

20

1 10 12 13 16 17 18 19 必須。インストールプログラムはこの値の入力を求めるプロンプトを出します。

2 6 **controlPlane** セクションは単一マッピングですが、コンピュートセクションはマッピングのシーケンスになります。複数の異なるデータ構造の要件を満たすには、**compute** セクションの最初の行はハイフン - で始め、**controlPlane** セクションの最初の行はハイフンで始めることができません。どちらのセクションも、現時点では単一のマシンプールを定義しますが、OpenShift Container Platform の今後のバージョンでは、インストール時の複数のコンピュートプールの定義をサポートする可能性があります。1つのコントロールプレーンプールのみが使用されます。

37 同時マルチスレッドまたは **hyperthreading** を有効/無効にするかどうか。デフォルトでは、同時マルチ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化されます。パラメーター値を **Disabled** に設定するとこれを無効にすることができます。一部のクラスターマシンで同時マルチスレッドを無効にする場合は、これをすべてのクラスターマシンで無効にする必要があります。



重要

同時マルチ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られていることを確認します。

48 オプション: コンピュートおよびコントロールプレーンマシンのマシンプールパラメーターの追加設定を指定します。

**5 9 14** オプション: プリズムカテゴリーとプリズムカテゴリー値のペアを1つ以上指定します。これらのカテゴリーのキーと値のペアは、Prism Centralに存在する必要があります。コンピューティングマシン、コントロールプレーンマシン、またはすべてのマシンに個別のカテゴリーを指定できます。

11 インストールするクラスターネットワークプラグイン。サポートされている値は **OVNKubernetes** と **OpenShiftSDN** です。デフォルトの値は **OVNkubernetes** です。

15 オプション: VM が関連付けられているプロジェクトを指定します。プロジェクトタイプの **name** または **uuid** を指定してから、対応する UUID またはプロジェクト名を指定します。プロジェクトは、コンピューティングマシン、コントロールプレーンマシン、またはすべてのマシンに関連付けることができます。

- 20 オプション: デフォルトでは、インストールプログラムは Red Hat Enterprise Linux CoreOS (RHCOS) イメージをダウンロードしてインストールします。Prism Central がインターネットにアクセスするようにするには、オプションで `imageContentSourcePolicy.yaml` ファイルを指定します。
- 21 <local\_registry> には、レジストリードメイン名と、ミラーレジストリーがコンテンツを提供するために使用するポートをオプションで指定します。例: `registry.example.com` または `registry.example.com:5000<credentials>` に、ミラーレジストリーの base64 でエンコードされたユーザー名およびパスワードを指定します。
- 22 FIPS モードを有効または無効にするかどうか。デフォルトでは、FIPS モードは有効にされません。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。



### 重要

FIPS モードでブートされた Red Hat Enterprise Linux (RHEL) または Red Hat Enterprise Linux CoreOS (RHCOS) を実行する場合、OpenShift Container Platform コアコンポーネントは、x86\_64、ppc64le、および s390x アーキテクチャーのみで、FIPS 140-2/140-3 検証のために NIST に提出された RHEL 暗号化ライブラリーを使用します。

- 23 オプション: クラスター内のマシンにアクセスするのに使用する `sshKey` 値をオプションで指定できます。



### 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のある実稼働用の OpenShift Container Platform クラスターでは、`ssh-agent` プロセスが使用する SSH キーを指定します。

- 24 ミラーレジストリーに使用した証明書ファイルの内容を指定します。
- 25 レジストリーをミラーリングしたときに作成された `imageContentSourcePolicy.yaml` ファイルの `metadata.name: release-0` セクションからこれらの値を指定します。

## 3.6.2. インストール時のクラスター全体のプロキシーの設定

実稼働環境では、インターネットへの直接アクセスを拒否し、代わりに HTTP または HTTPS プロキシーを使用することができます。プロキシー設定を `install-config.yaml` ファイルで行うことにより、新規の OpenShift Container Platform クラスターをプロキシーを使用するように設定できます。

### 前提条件

- 既存の `install-config.yaml` ファイルがある。
- クラスターがアクセスする必要のあるサイトを確認済みで、それらのいずれかがプロキシーをバイパスする必要があるかどうかを判別している。デフォルトで、すべてのクラスター Egress トラフィック (クラスターをホストするクラウドに関するクラウドプロバイダー API に対する呼び出しを含む) はプロキシーされます。プロキシーを必要に応じてバイパスするために、サイトを `Proxy` オブジェクトの `spec.noProxy` フィールドに追加している。



## 注記

Proxy オブジェクトの **status.noProxy** フィールドには、インストール設定の **networking.machineNetwork[].cidr**、**networking.clusterNetwork[].cidr**、および **networking.serviceNetwork[]** フィールドの値が設定されます。

Amazon Web Services (AWS)、Google Cloud、Microsoft Azure、および Red Hat OpenStack Platform (RHOSP)へのインストールの場合、Proxy オブジェクトの **status.noProxy** フィールドには、インスタンスマタデータのエンドポイント(169.254.169.254)も設定されます。

## 手順

1. **install-config.yaml** ファイルを編集し、プロキシー設定を追加します。以下に例を示します。

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> ①
  httpsProxy: https://<username>:<pswd>@<ip>:<port> ②
  noProxy: example.com ③
  additionalTrustBundle: | ④
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> ⑤
```

- 1 クラスター外の HTTP 接続を作成するために使用するプロキシー URL。URL スキームは **http** である必要があります。
- 2 クラスター外で HTTPS 接続を作成するために使用するプロキシー URL。
- 3 プロキシーから除外するための宛先ドメイン名、IP アドレス、または他のネットワーク CIDR のコンマ区切りのリスト。サブドメインのみと一致するように、ドメインの前に . を付けます。たとえば、**.y.com** は **x.y.com** に一致しますが、**y.com** には一致しません。\* を使用し、すべての宛先のプロキシーをバイパスします。
- 4 指定されている場合、インストールプログラムは HTTPS 接続のプロキシーに必要な1つ以上の追加の CA 証明書が含まれる **user-ca-bundle** という名前の設定マップを **openshift-config** namespace に生成します。次に Cluster Network Operator は、これらのコンテンツを Red Hat Enterprise Linux CoreOS (RHCOS) 信頼バンドルにマージする **trusted-ca-bundle** 設定マップを作成し、この設定マップは Proxy オブジェクトの **trustedCA** フィールドで参照されます。**additionalTrustBundle** フィールドは、プロキシーのアイデンティティー証明書が RHCOS 信頼バンドルからの認証局によって署名されない限り必要になります。
- 5 オプション: **trustedCA** フィールドの **user-ca-bundle** 設定マップを参照する Proxy オブジェクトの設定を決定するポリシー。許可される値は **Proxyonly** および **Always** です。**Proxyonly** を使用して、**http/https** プロキシーが設定されている場合にのみ **user-ca-bundle** 設定マップを参照します。**Always** を使用して、常に **user-ca-bundle** 設定マップを参照します。デフォルト値は **Proxyonly** です。



### 注記

インストールプログラムは、プロキシーの **readinessEndpoints** フィールドをサポートしません。



### 注記

インストーラーがタイムアウトした場合は、インストーラーの **wait-for** コマンドを使用してデプロイメントを再起動してからデプロイメントを完了します。以下に例を示します。

```
$ ./openshift-install wait-for install-complete --log-level debug
```

2. ファイルを保存し、OpenShift Container Platform のインストール時にこれを参照します。

インストールプログラムは、指定の **install-config.yaml** ファイルのプロキシー設定を使用する **cluster** という名前のクラスター全体のプロキシーを作成します。プロキシー設定が指定されていない場合、**cluster Proxy** オブジェクトが依然として作成されますが、これには **spec** がありません。



### 注記

**cluster** という名前の **Proxy** オブジェクトのみがサポートされ、追加のプロキシーを作成することはできません。

## 3.7. 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 4.14 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。

### 3.7.1. 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の [OpenShift Container Platform ダウンロードページ](#) に移動します。
2. **Product Variant** ドロップダウンリストからアーキテクチャーを選択します。
3. **バージョン** ドロップダウンリストから適切なバージョンを選択します。
4. **OpenShift v4.14 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。
5. アーカイブを展開します。

```
$ tar xvf <file>
```

6. **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
```

## 検証

- 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>
```

### 3.7.2. 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 ポータルの [OpenShift Container Platform ダウンロード ページ](#) に移動します。
2. バージョン ドロップダウンリストから適切なバージョンを選択します。
3. **OpenShift v4.14 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展開します。
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
```

## 検証

- 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>
```

### 3.7.3. 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 ポータルの [OpenShift Container Platform ダウンロード ページ](#) に移動します。
2. バージョン ドロップダウンリストから適切なバージョンを選択します。

3. OpenShift v4.14 macOS 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。



### 注記

macOS arm64 の場合は、OpenShift v4.14 macOS arm64 Client エントリーを選択します。

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
```

### 検証

- **oc** コマンドを使用してインストールを確認します。

```
$ oc <command>
```

## 3.8. NUTANIX の IAM の設定

クラスターをインストールするには、Cloud Credential Operator (CCO) が手動モードで動作する必要があります。インストールプログラムが手動モード用に CCO を設定する間、ID およびアクセス管理シークレットを指定する必要があります。

### 前提条件

- **ccocctl** バイナリーを設定している。
- **install-config.yaml** ファイルがある。

### 手順

1. 認証情報データを含む YAML ファイルを次の形式で作成します。

#### 認証情報データの形式

```
credentials:
- type: basic_auth ①
  data:
    prismCentral: ②
      username: <username_for_prism_central>
      password: <password_for_prism_central>
    prismElements: ③
      - name: <name_of_prism_element>
        username: <username_for_prism_element>
        password: <password_for_prism_element>
```

- ① 認証タイプを指定します。Basic 認証のみがサポートされています。

- 2 Prism Central の認証情報を指定します。
- 3 オプション: Prism Element 認証情報を指定します。

2. 次のコマンドを実行して、インストールファイルのリリースイメージを **\$RELEASE\_IMAGE** 変数に設定します。

```
$ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
```

3. 以下のコマンドを実行して、OpenShift Container Platform リリースイメージから **CredentialsRequest** カスタムリソース (CR) のリストを抽出します。

```
$ oc adm release extract \
--from=$RELEASE_IMAGE \
--credentials-requests \
--included \① \
--install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \② \
--to=<path_to_directory_for_credentials_requests> \③
```

- 1 **--included** パラメーターには、特定のクラスター設定に必要なマニフェストのみが含まれます。
- 2 **install-config.yaml** ファイルの場所を指定します。
- 3 **CredentialsRequest** オブジェクトを保存するディレクトリーへのパスを指定します。指定したディレクトリーが存在しない場合は、このコマンドによって作成されます。

### サンプル **CredentialsRequest** オブジェクト

```
apiVersion: cloudcredential.openshift.io/v1
kind: CredentialsRequest
metadata:
  annotations:
    include.release.openshift.io/self-managed-high-availability: "true"
  labels:
    controller-tools.k8s.io: "1.0"
  name: openshift-machine-api-nutanix
  namespace: openshift-cloud-credential-operator
spec:
  providerSpec:
    apiVersion: cloudcredential.openshift.io/v1
    kind: NutanixProviderSpec
  secretRef:
    name: nutanix-credentials
    namespace: openshift-machine-api
```

4. 次のコマンドを実行し、**ccctl** ツールを使用して **CredentialsRequest** オブジェクトをすべて処理します。

```
$ ccctl nutanix create-shared-secrets \
--credentials-requests-dir=<path_to_credentials_requests_directory> \① \
--output-dir=<ccctl_output_dir> \②
```

```
--credentials-sourcefilepath=<path_to_credentials_file> ③
```

- ① コンポーネント **CredentialsRequests** オブジェクトのファイルを含むディレクトリーへのパスを指定します。
- ② オプション: **ccctl** ユーティリティーがオブジェクトを作成するディレクトリーを指定します。デフォルトでは、ユーティリティーは、コマンドが実行されるディレクトリーにオブジェクトを作成します。
- ③ オプション: 認証情報データ YAML ファイルを含むディレクトリーを指定します。デフォルトでは、**ccctl** はこのファイルが **<home\_directory>/nutanix/credentials** にあると想定します。

5. **credentialsMode** パラメーターが **Manual** に設定されるように、**install-config.yaml** 設定ファイルを編集します。

#### サンプル **install-config.yaml** 設定ファイル

```
apiVersion: v1
baseDomain: cluster1.example.com
credentialsMode: Manual ①
...
...
```

- ① この行を追加して、**credentialsMode** パラメーターを **Manual** に設定します。
6. 次のコマンドを実行して、インストールマニフェストを作成します。

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

- ① クラスターの **install-config.yaml** ファイルを含むディレクトリーへのパスを指定します。
7. 次のコマンドを実行して、生成された認証情報ファイルをターゲットマニフェストディレクトリーにコピーします。

```
$ cp <ccctl_output_dir>/manifests/*credentials.yaml ./<installation_directory>/manifests
```

#### 検証

- **manifests** ディレクトリーに適切なシークレットが存在することを確認します。

```
$ ls ./<installation_directory>/manifests
```

#### 出力例

```
cluster-config.yaml
cluster-dns-02-config.yaml
cluster-infrastructure-02-config.yaml
cluster-ingress-02-config.yaml
cluster-network-01-crd.yaml
cluster-network-02-config.yaml
```

```
cluster-proxy-01-config.yaml
cluster-scheduler-02-config.yaml
cvo-overrides.yaml
kube-cloud-config.yaml
kube-system-configmap-root-ca.yaml
machine-config-server-tls-secret.yaml
openshift-config-secret-pull-secret.yaml
openshift-cloud-controller-manager-nutanix-credentials-credentials.yaml
openshift-machine-api-nutanix-credentials-credentials.yaml
```

### 3.9. 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

- 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がある。
- ホスト上のクラウドプロバイダーアカウントに、クラスターをデプロイするための正しい権限があることを確認した。アカウントの権限が正しくないと、インストールプロセスが失敗し、不足している権限を示すエラーメッセージが表示されます。

#### 手順

- 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>** に、カスタマイズした **./install-config.yaml** ファイルの場所を指定します。
- 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。

#### 検証

クラスターのデプロイが正常に完了すると、次のようにになります。

- ターミナルには、Web コンソールへのリンクや **kubeadm** ユーザーの認証情報など、クラスターにアクセスするための指示が表示されます。
- 認証情報は **<installation\_directory>/openshift\_install.log** にも出力されます。



## 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。

## 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: "password"
INFO Time elapsed: 36m22s
```



## 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](#)に関する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推奨します。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。

## 3.10. インストール後の設定

以下の手順を実行して、クラスターの設定を完了します。

### 3.10.1. デフォルトの OperatorHub カタログソースの無効化

Red Hat によって提供されるコンテンツを調達する Operator カタログおよびコミュニティープロジェクトは、OpenShift Container Platform のインストール時にデフォルトで OperatorHub に設定されます。ネットワークが制限された環境では、クラスター管理者としてデフォルトのカタログを無効にする必要があります。

#### 手順

- disableAllDefaultSources: true** を **OperatorHub** オブジェクトに追加して、デフォルトカタログのソースを無効にします。

```
$ oc patch OperatorHub cluster --type json \
-p '[{"op": "add", "path": "/spec/disableAllDefaultSources", "value": true}]'
```

## ヒント

または、Web コンソールを使用してカタログソースを管理できます。Administration → Cluster Settings → Configuration → OperatorHub ページから、Sources タブをクリックして、個別のソースを作成、更新、削除、無効化、有効化できます。

### 3.10.2. クラスターへのポリシーリソースのインストール

oc-mirror OpenShift CLI (oc) プラグインを使用して OpenShift Container Platform コンテンツをミラーリングすると、**catalogSource-certified-operator-index.yaml** および **imageContentSourcePolicy.yaml** を含むリソースが作成されます。

- **ImageContentSourcePolicy** リソースは、ミラーレジストリーをソースレジストリーに関連付け、イメージフル要求をオンラインレジストリーからミラーレジストリーにリダイレクトします。
- **CatalogSource** リソースは、Operator Lifecycle Manager (OLM) によって使用され、ミラーレジストリーで使用可能な Operator に関する情報を取得します。これにより、ユーザーは Operator を検出してインストールできます。

クラスターをインストールしたら、これらのリソースをクラスターにインストールする必要があります。

#### 前提条件

- 非接続環境で、イメージセットをレジストリーミラーにミラーリングしました。
- **cluster-admin** ロールを持つユーザーとしてクラスターにアクセスできる。

#### 手順

1. **cluster-admin** ロールを持つユーザーとして OpenShift CLI にログインします。
2. results ディレクトリーからクラスターに YAML ファイルを適用します。

```
$ oc apply -f ./oc-mirror-workspace/results-<id>/
```

#### 検証

1. **ImageContentSourcePolicy** リソースが正常にインストールされたことを確認します。
 

```
$ oc get imagecontentsourcepolicy
```
2. **CatalogSource** リソースが正常にインストールされたことを確認します。
 

```
$ oc get catalogsource --all-namespaces
```

### 3.10.3. デフォルトのストレージコンテナーの設定

クラスターをインストールしたら、Nutanix CSI Operator をインストールし、クラスターのデフォルトのストレージコンテナーを設定する必要があります。

詳細は、[CSI Operator のインストール](#) と [レジストリーストレージの設定](#) に関する Nutanix のドキュメントを参照してください。

### 3.11. OPENSHIFT CONTAINER PLATFORM の TELEMETRY アクセス

OpenShift Container Platform 4.14 では、クラスターの健全性および正常に実行された更新についての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 サービスにも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](#) インベントリーが正常である (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。

### 3.12. 関連情報

- [リモートヘルスモニタリングについて](#)

### 3.13. 次のステップ

- 必要に応じて、[リモートヘルスレポート](#) を参照してください。
- 必要に応じて、[非接続クラスターの登録](#) を参照してください。
- [クラスターのカスタマイズ](#)

## 第4章 NUTANIXへの3ノードクラスターのインストール

OpenShift Container Platform バージョン 4.14 では、Nutanix に3ノードクラスターをインストールできます。3ノードクラスターは、コンピューティングマシンとしても機能する3つのコントロールプレーンマシンで設定されます。このタイプのクラスターは、クラスター管理者および開発者がテスト、開発、および実稼働に使用するためのより小さくリソース効率の高いクラスターを提供します。

### 4.1.3 ノードクラスターの設定

クラスターをデプロイする前に、**install-config.yaml** ファイルでワーカーノードの数を **0** に設定して、3ノードクラスターを設定します。ワーカーノードの数を **0** に設定すると、コントロールプレーンマシンがスケジュール可能になります。これにより、アプリケーションワークロードをコントロールプレーンノードから実行するようにスケジュールできます。



#### 注記

アプリケーションワークロードはコントロールプレーンノードから実行され、コントロールプレーンノードはコンピュートノードと見なされるため、追加のサブスクリプションが必要です。

#### 前提条件

- 既存の **install-config.yaml** ファイルがある。

#### 手順

- 次の **compute** スタンザに示すように、**install-config.yaml** ファイルでコンピューティングレプリカの数を **0** に設定します。

#### 3ノードクラスターの **install-config.yaml** ファイルの例

```
apiVersion: v1
baseDomain: example.com
compute:
- name: worker
  platform: {}
  replicas: 0
# ...
```

### 4.2. 次のステップ

- クラスターの Nutanix へのインストール

## 第5章 NUTANIX でのクラスターのアンインストール

Nutanix にデプロイしたクラスターを削除できます。

### 5.1. INSTALLER-PROVISIONED INFRASTRUCTURE を使用するクラスターの削除

installer-provisioned infrastructure 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 user-provisioned infrastructure (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しなかったリソースや、インストーラーがアクセスできないリソースが存在す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る。
- 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

- 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。

- 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。

# 第6章 NUTANIX のインストール設定パラメーター

OpenShift Container Platform クラスターを Nutanix にデプロイする前に、クラスターとそれをホストするプラットフォームをカスタマイズするためのパラメーターを指定します。 **install-config.yaml** ファイルを作成するときは、コマンドラインを使用して必要なパラメーターの値を指定します。その後、**install-config.yaml** ファイルを変更して、クラスターをさらにカスタマイズできます。

## 6.1. NUTANIX で使用可能なインストール設定パラメーター

次の表では、インストールプロセスの一部として設定できる、必須、オプション、および Nutanix 固有のインストール設定パラメーターを指定します。



### 重要

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。

### 6.1.1. 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6.1 必須パラメーター

パラメーター	説明	値
<code>apiVersion:</code>	<code>install-config.yaml</code> 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。インストールプログラムは、古い API バージョンもサポートしている場合があります。	String
<code>baseDomain:</code>	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、 <code>&lt;metadata.name&gt;.&lt;baseDomain&gt;</code> 形式を使用する <code>baseDomain</code> と <code>metadata.name</code> パラメーターの値を組み合わせたものです。	<code>example.com</code> などの完全修飾ドメインまたはサブドメイン名。
<code>metadata:</code>	Kubernetes リソース <code>ObjectMeta</code> 。ここからは <code>name</code> パラメーターのみが消費されます。	オブジェクト

パラメーター	説明	値
metadata: name:	クラスターの名前。クラスターの DNS レコードはすべて <code>{{.metadata.name}}.{{.baseDomain}}</code> のサブドメインです。	小文字いちばんハイフン (-) の文字列 (dev など)。
platform:	インストールを実行する特定のプラットフォームの設定: <b>alibabacloud</b> 、 <b>aws</b> 、 <b>baremetal</b> 、 <b>azure</b> 、 <b>gcp</b> 、 <b>ibmcloud</b> 、 <b>nutanix</b> 、 <b>openstack</b> 、 <b>powervs</b> 、 <b>vsphere</b> 、または <code>{}</code> 。 <b>platform</b> <platform> 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を参照してください。	オブジェクト
pullSecret:	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証します。	<pre>{   "auths": {     "cloud.openshift.com": {       "auth": "b3Blb=",       "email": "you@example.com"     },     "quay.io": {       "auth": "b3Blb=",       "email": "you@example.com"     }   } }</pre>

### 6.1.2. 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したり、デフォルトとは異なる IP アドレスブロックを設定したりすることができます。

IPv4 アドレスのみがサポートされます。



#### 注記

Globalnet は、Red Hat OpenShift Data Foundation ディザスター・リカバリー・ソリューションではサポートされていません。局地的なディザスター・リカバリーのシナリオでは、各クラスター内のクラスターとサービスネットワークに重複しない範囲のプライベート IP アドレスを使用するようにしてください。

表6.2 ネットワークパラメーター

パラメーター	説明	値
networking:	クラスターのネットワークの設定。	<p>Object</p>  <p><b>注記</b></p> <p>インストール後に <b>networking</b> オブジェクトによって指定されたパラメーターを変更することはできません。</p>
networking: networkType:	インストールする Red Hat OpenShift Networking ネットワークプラグイン。	<b>OpenShiftSDN</b> または <b>OVNKubernetes</b> のいずれか。 <b>OpenShiftSDN</b> は、全 Linux ネットワーク用の CNI プラグインです。 <b>OVNKubernetes</b> は、Linux ネットワークと、Linux サーバーと Windows サーバーの両方を含む Linux ネットワークおよびハイブリッドネットワーク用の CNI プラグインです。デフォルトの値は <b>OVNKubernetes</b> です。
networking: clusterNetwork:	<p>Pod の IP アドレスブロック。</p> <p>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。</p> <p>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。</p>	<p>オブジェクトの配列。以下に例を示します。</p> <pre>networking:   clusterNetwork:     - cidr: 10.128.0.0/14       hostPrefix: 23</pre>
networking: clusterNetwork: cidr:	<p><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。</p> <p>IPv4 ネットワーク</p>	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> の間にあります。
networking: clusterNetwork: hostPrefix:	<p>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、<math>510 (2^{32-23}-2)</math> Pod IP アドレスを提供します。</p>	<p>サブネット接頭辞。</p> <p>デフォルト値は <b>23</b> です。</p>

パラメーター	説明	値
networking: serviceNetwork:	<p>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。</p> <p>OpenShift SDN および OVN-Kubernetes ネットワークプラグインは、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。</p>	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。
networking: machineNetwork:	<p>マシンの IP アドレスブロック。</p> <p>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。</p>	<p>オブジェクトの配列。以下に例を示します。</p> <p>networking: machineNetwork: - cidr: <b>10.0.0.0/16</b></p>
networking: machineNetwork: cidr:	<p><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 と IBM Power® Virtual Server を除くすべてのプラットフォームのデフォルト値は <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。IBM Power® Virtual Server の場合、デフォルト値は <b>192.168.0.0/24</b> です。</p>	<p>CIDR 表記の IP ネットワークブロック。</p> <p>例: <b>10.0.0.0/16</b></p> <p></p> <p><b>注記</b></p> <p>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。</p>

### 6.1.3. 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6.3 オプションのパラメーター

パラメーター	説明	値
additionalTrustBundle:	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、プロキシーが設定されている場合にも使用することができます。	String

パラメーター	説明	値
capabilities:	オプションのコアクラスターコンポーネントのインストールを制御します。オプションのコンポーネントを無効にすることで、OpenShift Container Platform クラスターのフットプリントを削減できます。詳細は、インストールの「クラスター機能ページ」を参照してください。	文字列配列
capabilities: baselineCapabilitySet:	有効にするオプション機能の初期セットを選択します。有効な値は <b>None</b> 、 <b>v4.11</b> 、 <b>v4.12</b> 、 <b>vCurrent</b> です。デフォルト値は <b>vCurrent</b> です。	文字列
capabilities: additionalEnabledCapabilities:	オプションの機能のセットを、 <b>baselineCapabilitySet</b> で指定したものを超えて拡張します。このパラメーターでは複数の機能を指定できます。	String array
cpuPartitioningMode:	ワークロードパーティション設定を使用して、OpenShift Container Platform サービス、クラスター管理ワークロード、およびインフラストラクチャー Pod を分離し、予約された CPU セットで実行できます。ワークロードパーティショニングを有効にできるのはインストール時のみです。インストール後は無効にできません。このフィールドはワークロードのパーティショニングを有効にしますが、特定の CPU を使用するようにワークロードを設定するわけではありません。詳細は、スケーラビリティとパフォーマンス セクションのワークロードパーティショニング ページを参照してください。	<b>None</b> または <b>AllNodes</b> 。デフォルト値は <b>None</b> です。
compute:	コンピュートノードを構成するマシンの設定。	<b>MachinePool</b> オブジェクトの配列。
compute: architecture:	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。現在、さまざまなアーキテクチャーのクラスターはサポートされていません。すべてのプールは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

パラメーター	説明	値
compute: hyperthreading:	コンピュート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、同時マルチ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化されます。	<b>Enabled</b> または <b>Disabled</b>
	<p><b>重要</b></p>  <p>同時マルチ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られていることを確認します。</p>	
compute: name:	<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>
compute: platform:	<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>alibabacloud</b> 、 <b>aws</b> 、 <b>azure</b> 、 <b>gcp</b> 、 <b>ibmcloud</b> 、 <b>nutanix</b> 、 <b>openstack</b> 、 <b>powervs</b> 、 <b>vsphere</b> 、または <code>{}</code>
compute: replicas:	プロビジョニングするコンピュートマシン(ワーカーマシンとしても知られる)の数。	2以上の正の整数。デフォルト値は3です。
featureSet:	機能セットのクラスターを有効にします。機能セットは、デフォルトで有効にされない OpenShift Container Platform 機能のコレクションです。インストール中に機能セットを有効にする方法の詳細は、「機能ゲートの使用による各種機能の有効化」を参照してください。	文字列。 <b>TechPreviewNoUpgrade</b> など、有効にする機能セットの名前。
controlPlane:	コントロールプレーンを構成するマシンの設定。	<b>MachinePool</b> オブジェクトの配列。

パラメーター	説明	値
controlPlane: architecture:	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。現在、さまざまなアーキテクチャーのクラスターはサポートされていません。すべてのプールは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
controlPlane: hyperthreading:	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、同時マルチ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化されます。	<b>Enabled</b> または <b>Disabled</b>
	<p><b>重要</b></p>  <p>同時マルチ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られていることを確認します。</p>	
controlPlane: name:	<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>
controlPlane: platform:	<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>alibabacloud</b> 、 <b>aws</b> 、 <b>azure</b> 、 <b>gcp</b> 、 <b>ibmcloud</b> 、 <b>nutanix</b> 、 <b>openstack</b> 、 <b>powervs</b> 、 <b>vsphere</b> 、または <b>{}</b>
controlPlane: replicas:	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> 、シングルノード OpenShift をデプロイする場合は <b>1</b> です。
credentialsMode:	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。	<b>Mint</b> 、 <b>Passthrough</b> 、 <b>Manual</b> 、または空の文字列 ("")。 <sup>[1]</sup>
	FIPS モードを有効または無効にしま	<b>false</b> または <b>true</b>

パラメーター	説明	値
	<p><b>fips:</b> す。デフォルトは <b>false</b> (無効) です。  <b>FIPS</b> モードを有効にすると、</p> <p>OpenShift Container Platform が稼働している Red Hat Enterprise Linux CoreOS (RHCOS) マシンで、デフォルトの Kubernetes 暗号化スイートが無視され、代わりに RHCOS が提供する暗号化モジュールが使用されます。</p> <p><b>重要</b></p> <p>クラスターで FIPS モードを有効にするには、FIPS モードで動作するように設定された Red Hat Enterprise Linux (RHEL) コンピューターからインストールプログラムを実行する必要があります。RHEL で FIPS モードを設定する方法の詳細は、RHEL から FIPS モードへの切り替え を参照してください。</p> <p>FIPS モードでブートされた Red Hat Enterprise Linux (RHEL) または Red Hat Enterprise Linux CoreOS (RHCOS) を実行する場合、OpenShift Container Platform コアコンポーネントは、x86_64、ppc64le、および s390x アーキテクチャーのみで、FIPS 140-2/140-3 検証のために NIST に提出された RHEL 暗号化ライブラリーを使用します。</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>	

パラメーター	説明	値
imageContentSources:	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。
imageContentSources: source:	<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: イメージフル仕様)。	String
imageContentSources: mirrors:	同じイメージが含まれている可能性があるリポジトリを1つ以上指定します。	文字列の配列
publish:	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>
sshKey:	クラスターマシンへのアクセスを認証するための SSH キー。	<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、クラスターが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>

### 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のある実稼働用の OpenShift Container Platform クラスターでは、**ssh-agent** プロセスが使用する SSH キーを指定します。

- 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、[認証と認可](#)コンテンツの「クラウドプロバイダーの認証情報の管理」を参照してください。

#### 6.1.4. 追加の Nutanix 設定パラメーター

追加の Nutanix 設定パラメーターは、次の表で説明します。

表6.4 追加の Nutanix クラスターパラメーター

パラメーター	説明	値
compute: platform: nutanix: categories: key:	コンピューティング VM に適用するプリズムカテゴリーのキーの名前。このパラメーターには、 <b>value</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。カテゴリーの詳細は、 <a href="#">カテゴリー管理</a> を参照してください。	文字列
compute: platform: nutanix: categories: value:	コンピューティング VM に適用するプリズムカテゴリーのキーと値のペアの値。このパラメーターには、 <b>key</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。	String
compute: platform: nutanix: project: type:	コンピューティング VM のプロジェクトを選択するために使用する識別子のタイプ。プロジェクトは、権限、ネットワーク、およびその他のパラメーターを管理するためのユーザーロールの論理グループを定義します。プロジェクトの詳細は、 <a href="#">プロジェクトの概要</a> を参照してください。	<b>name</b> または <b>uuid</b>
compute: platform: nutanix: project: name: or uuid:	コンピューティング VM が関連付けられているプロジェクトの名前または UUID。このパラメーターには、 <b>type</b> パラメーターを指定する必要があります。	文字列

パラメーター	説明	値
compute: platform: nutanix: bootType:	コンピューティングマシンが使用するブートタイプ。OpenShift Container Platform 4.14 では、 <b>Legacy</b> ブートタイプを使用する必要があります。ブートタイプの詳細については、 <a href="#">仮想化環境内のUEFI、セキュアブート、およびTPMについて</a> を参照してください。	<b>Legacy</b> 、 <b>SecureBoot</b> 、または <b>UEFI</b> 。デフォルトは、 <b>Legacy</b> です。
controlPlane: platform: nutanix: categories: key:	コントロールプレーン VM に適用するプリズムカテゴリーのキーの名前。このパラメーターには、 <b>value</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。カテゴリーの詳細は、 <a href="#">カテゴリー管理</a> を参照してください。	文字列
controlPlane: platform: nutanix: categories: value:	コントロールプレーン VM に適用するプリズムカテゴリーのキーと値のペアの値。このパラメーターには、 <b>key</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。	String
controlPlane: platform: nutanix: project: type:	コントロールプレーン VM のプロジェクトを選択するために使用する識別子のタイプ。プロジェクトは、権限、ネットワーク、およびその他のパラメーターを管理するためのユーザー ロールの論理グループを定義します。プロジェクトの詳細は、 <a href="#">プロジェクトの概要</a> を参照してください。	<b>name</b> または <b>uuid</b>
controlPlane: platform: nutanix: project: name: or uuid:	コントロールプレーン VM が関連付けられているプロジェクトの名前または UUID。このパラメーターには、 <b>type</b> パラメーターを指定する必要があります。	文字列

パラメーター	説明	値
platform: nutanix:  defaultMachinePlatform: categories: key:	すべての VM に適用するプリズムカテゴリーのキーの名前。このパラメーターには、 <b>value</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。カテゴリーの詳細は、 <a href="#">カテゴリー管理</a> を参照してください。	文字列
platform: nutanix:  defaultMachinePlatform: categories: value:	すべての VM に適用するプリズムカテゴリーのキーと値のペアの値。このパラメーターには、 <b>key</b> パラメーターを指定する必要があります、 <b>key</b> と <b>value</b> の両方のパラメーターが Prism Central に存在する必要があります。	String
platform: nutanix:  defaultMachinePlatform: project: type:	すべての VM のプロジェクトを選択するためには、使用する識別子のタイプ。プロジェクトは、権限、ネットワーク、およびその他のパラメーターを管理するためのユーザーロールの論理グループを定義します。プロジェクトの詳細は、 <a href="#">プロジェクトの概要</a> を参照してください。	<b>name</b> または <b>uuid</b> 。
platform: nutanix:  defaultMachinePlatform: project: name: or uuid:	すべての VM が関連付けられているプロジェクトの名前または UUID。このパラメーターには、 <b>type</b> パラメーターを指定する必要があります。	文字列
platform: nutanix:  defaultMachinePlatform: bootType:	すべてのマシンのブートタイプ。OpenShift Container Platform 4.14 では、 <b>Legacy</b> ブートタイプを使用する必要があります。ブートタイプの詳細については、 <a href="#">仮想化環境内の UEFI、セキュアブート、および TPM</a> についてを参照してください。	<b>Legacy</b> 、 <b>SecureBoot</b> 、または <b>UEFI</b> 。デフォルトは、 <b>Legacy</b> です。

パラメーター	説明	値
platform: nutanix: apiVIP:	コントロールプレーン API のアクセス用に設定した仮想 IP (VIP) アドレス。	IP アドレス
platform: nutanix: ingressVIP:	クラスター Ingress に設定した仮想 IP (VIP) アドレス。	IP アドレス
platform: nutanix: prismCentral: endpoint: address:	Prism Central ドメイン名または IP アドレス。	文字列
platform: nutanix: prismCentral: endpoint: port:	Prism Central へのログインに使用されるポート。	文字列
platform: nutanix: prismCentral: password:	Prism Central ユーザー名のパスワード。	文字列
platform: nutanix: prismCentral: username:	Prism Central へのログインに使用されるユーザー名。	文字列
platform: nutanix: prismElements: endpoint: address:	Prism Element ドメイン名または IP アドレス。[ <sup>1</sup> ]	文字列

パラメーター	説明	値
platform: nutanix: prismElements: endpoint: port:	Prism Elementへのログインに使用されるポート。	文字列
platform: nutanix: prismElements: uuid:	Prism Element の Universally Unique Identifier (UUID)。	文字列
platform: nutanix: subnetUUIDs:	設定した仮想 IP アドレスと DNS レコードを含む Prism Element ネットワークの UUID。[ <sup>2</sup> ]	文字列
platform: nutanix: clusterOSImage:	オプション: デフォルトでは、インストールプログラムは Red Hat Enterprise Linux CoreOS (RHCOS) イメージをダウンロードしてインストールします。Prism Central がインターネットにアクセスできない場合は、任意の HTTP サーバーで RHCOS イメージをホストし、インストールプログラムがそのイメージを指すようにすることで、デフォルトの動作をオーバーライドできます。	HTTP または HTTPS の URL (オプションで SHA-256 形式のチェックサムを使用)。例: <a href="http://example.com/images/rhcos-47.83.202103221318-0-nutanix.x86_64.qcow2">http://example.com/images/rhcos-47.83.202103221318-0-nutanix.x86_64.qcow2</a>

1. **prismElements** セクションには、Prism Elements (クラスター) のリストが含まれています。Prism Element は、OpenShift Container Platform クラスターをホストするために使用されるすべての Nutanix リソース (仮想マシンやサブネットなど) を包含します。サポートされている Prism Element は1つだけです。
2. OpenShift Container Platform クラスターごとに1つのサブネットのみがサポートされます。