



Red Hat Advanced Cluster Management for Kubernetes 2.0

クラスターの管理

クラスターの管理

Red Hat Advanced Cluster Management for Kubernetes 2.0 クラスターの管理

クラスターの管理

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/Manage_cluster.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[®] 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.

概要

Red Hat Advanced Cluster Management for Kubernetes でのクラスターの管理

目次

第1章 RED HAT ADVANCED CLUSTER MANAGEMENT FOR KUBERNETES でのクラスターの管理	4
第2章 サポート対象のクラウド	5
2.1. サポート対象のハブクラスタープロバイダー	5
2.2. サポート対象のマネージドクラスタープロバイダー	5
2.3. KUBECTL の設定	5
第3章 クラスターのサイズ調整	6
3.1. AMAZON WEB SERVICES	6
3.2. GOOGLE CLOUD PLATFORM	6
3.3. MICROSOFT AZURE	6
3.4. ベアメタルクラスター	7
3.5. IBM KUBERNETES SERVICE	7
第4章 リリースイメージ	8
4.1. 利用可能なリリースイメージの同期	8
4.1.1. 接続時におけるリリースイメージのカスタム一覧の管理	10
4.1.2. 非接続時におけるリリースイメージのカスタム一覧の管理	11
第5章 ベアメタルアセットの作成および変更	12
5.1. 前提条件	12
5.2. コンソールを使用したベアメタルアセットの作成	13
5.3. ベアメタルアセットの変更	14
5.4. ベアメタルアセットの削除	14
第6章 プロバイダー接続の作成	15
6.1. AMAZON WEB SERVICES のプロバイダー接続の作成	15
6.1.1. 前提条件	15
6.1.2. コンソールを使用したプロバイダー接続の作成	15
6.1.3. プロバイダー接続の削除	16
6.2. MICROSOFT AZURE のプロバイダー接続の作成	16
6.2.1. 前提条件	17
6.2.2. コンソールを使用したプロバイダー接続の作成	17
6.2.3. プロバイダー接続の削除	18
6.3. GOOGLE CLOUD PLATFORM のプロバイダー接続の作成	18
6.3.1. 前提条件	19
6.3.2. コンソールを使用したプロバイダー接続の作成	19
6.3.3. プロバイダー接続の削除	20
6.4. ベアメタルのプロバイダー接続の作成	20
6.4.1. 前提条件	21
6.4.2. コンソールを使用したプロバイダー接続の作成	22
6.4.3. プロバイダー接続の削除	23
第7章 RED HAT ADVANCED CLUSTER MANAGEMENT FOR KUBERNETES でのクラスターの作成	24
7.1. AMAZON WEB SERVICES でのクラスターの作成	24
7.1.1. 前提条件	24
7.1.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成	24
7.1.3. クラスターへのアクセス	26
7.1.4. マネージメントからのクラスターの削除	26
7.2. MICROSOFT AZURE でのクラスターの作成	27
7.2.1. 前提条件	27
7.2.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成	27
7.2.3. クラスターへのアクセス	29

7.2.4. マネージメントからのクラスターの削除	29
7.3. GOOGLE CLOUD PLATFORM でのクラスターの作成	30
7.3.1. 前提条件	30
7.3.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成	30
7.3.3. クラスターへのアクセス	32
7.3.4. マネージメントからのクラスターの削除	32
7.4. ベアメタルでのクラスターの作成	33
7.4.1. 前提条件	33
7.4.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成	34
7.4.3. クラスターへのアクセス	35
7.4.4. マネージメントからのクラスターの削除	36
第8章 ハブクラスターへのターゲットのマネージドクラスターのインポート	37
8.1. コンソールを使用した既存クラスターのインポート	37
8.1.1. 前提条件	37
8.1.2. クラスターのインポート	37
8.1.2.1. YAML パラメーターおよび説明	39
8.1.3. インポートされたクラスターの削除	40
8.2. CLI を使用したマネージドクラスターのインポート	40
8.2.1. 前提条件	41
8.2.2. サポート対象のアーキテクチャー	41
8.2.3. インポートの準備	41
8.2.4. klusterlet のインポート	42
8.3. クラスターの KLUSTRLET アドオン設定の変更	43
8.3.1. ハブクラスターのコンソールを使用した変更	44
8.3.2. ハブクラスターのコマンドラインを使用した変更	44
第9章 クラスターのアップグレード	45
9.1. 非接続クラスターのアップグレード	45
9.1.1. 前提条件	46
9.1.2. 非接続ミラーレジストリーの準備	46
9.1.3. OpenShift Update Service の Operator のデプロイ	47
9.1.4. グラフデータの init コンテナの構築	47
9.1.5. ミラーリングされたレジストリーの証明書の設定	48
9.1.6. OpenShift Update Service インスタンスのデプロイ	49
9.1.7. デフォルトレジストリーを上書きするためのポリシーのデプロイ (任意)	49
9.1.8. 非接続カタログソースをデプロイするためのポリシーのデプロイ	51
9.1.9. マネージドクラスターのパラメーターを変更するためのポリシーのデプロイ	53
9.1.10. 利用可能なアップグレードの表示	56
9.1.11. クラスターのアップグレード	56

第1章 RED HAT ADVANCED CLUSTER MANAGEMENT FOR KUBERNETES でのクラスターの管理

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用した、クラウドプロバイダー全体におけるクラスターの作成、インポート、管理の方法を説明します。

以下のトピックでは、クラウドプロバイダー全体でクラスターを管理する方法について説明します。

- [サポート対象のクラウド](#)
- [クラスターのサイズ調整](#)
- [プロバイダー接続の作成](#)
- [Red Hat Advanced Cluster Management for Kubernetes でのクラスターの作成](#)
- [ハブクラスターへのターゲットのマネージドクラスターのインポート](#)
- [クラスターのアップグレード](#)

第2章 サポート対象のクラウド

Red Hat Advanced Cluster Management for Kubernetes で利用可能なクラウドプロバイダーについて説明します。また、利用可能なマネージドプロバイダーに関するドキュメントも参照してください。

- [サポート対象のハブクラスタープロバイダー](#)
- [サポート対象のマネージドクラスタープロバイダー](#)
- [kubectl の設定](#)

ベストプラクティス: マネージドクラスターのプロバイダーには、最新版の Kubernetes を使用してください。

2.1. サポート対象のハブクラスタープロバイダー

ハブクラスターとしてサポートされるのは、Red Hat OpenShift Container Platform 4.3.18 以降、4.4.4 以降、4.5.2 以降です。

- [OpenShift on Amazon Web Services](#) を参照してください。

2.2. サポート対象のマネージドクラスタープロバイダー

マネージドクラスターとしてサポートされるのは、Red Hat OpenShift Container Platform 3.11.200 以降、4.3.18 以降、4.4.4 以降、4.5.2 以降です。

利用可能なマネージドクラスターのオプションおよびドキュメントは以下を参照してください。

- [OpenShift on Amazon Web Services](#) を参照してください。
- [Red Hat OpenShift on IBM Cloud](#) を参照してください。
- [Red Hat OpenShift Kubernetes Engine](#) を参照してください。
- [IBM Cloud Kubernetes Service 概説](#) を参照してください。
- [Google Kubernetes Engine](#) を参照してください。
- [Azure Kubernetes Service](#) を参照してください。
- [Amazon Elastic Container Service for Kubernetes](#) を参照してください。

2.3. KUBECTL の設定

前述したベンダーのドキュメントを参照し、**kubectl** の設定方法を確認してください。マネージドクラスターをハブクラスターにインポートする場合には **kubectl** をインストールしておく必要があります。詳細は、「[ハブクラスターへのターゲットのマネージドクラスターのインポート](#)」を参照してください。

第3章 クラスターのサイズ調整

仮想マシンのサイズやノード数などのマネージドクラスターの仕様をカスタマイズできます。利用可能なプロバイダーごとの推奨設定は以下の一覧を参照してください。ただし、詳細な情報については、ドキュメントも参照してください。

3.1. AMAZON WEB SERVICES

Amazon Web Services 環境で作成された Red Hat OpenShift Container Platform クラスターのノード数は、ハブクラスターの **MachineSet** パラメーターを修正することで変更できます。

注記: Red Hat Advanced Cluster Mangement for Kubernetes は Hive for OpenShift を使用してクラスター内のノード数を判断するので、**MachineSet** パラメーターを変更してノード数を変える必要があります。**MachineSets** パラメーターを変更せずにノードを削除または追加すると、このパラメーターの現在の値に一致するように、ノードが追加または削除されます。

お使いのバージョンに該当する OpenShift Container Platform ドキュメントの「[クラスターのスケーリングに関する推奨プラクティス](#)」および「[MachineSet の手動によるスケーリング](#)」を参照してください。

ヒント: Red Hat Advanced Cluster Management for Kubernetes コンソールを使用してクラスターを作成した場合には、これは OpenShift Container Platform クラスターになります。

インポートした Amazon EKS クラスターにあるノード数を変更する場合には、「[Cluster autoscaler](#)」でクラスターのスケーリングに関する情報を参照してください。

3.2. GOOGLE CLOUD PLATFORM

Google Cloud Platform 環境で作成された Red Hat OpenShift Container Platform クラスターのノード数は、ハブクラスターの **MachineSet** パラメーターを修正することで変更できます。

注記: Red Hat Advanced Cluster Mangement for Kubernetes は Hive for OpenShift を使用してクラスター内のノード数を判断するので、**MachineSet** パラメーターを変更してノード数を変える必要があります。**MachineSets** パラメーターを変更せずにノードを削除または追加すると、このパラメーターの現在の値に一致するように、ノードが追加または削除されます。

クラスターのスケーリングに関する詳細は、お使いのバージョンに該当する OpenShift Container Platform ドキュメントの「[クラスターのスケーリングに関する推奨プラクティス](#)」および「[MachineSet の手動によるスケーリング](#)」を参照してください。**ヒント:** Red Hat Advanced Cluster Management を使用してクラスターを作成した場合には、これは OpenShift Container Platform クラスターになります。

インポートした Google Kubernetes Engine クラスターにあるノード数を変更する場合には、「[Resizing a cluster](#)」でクラスターのスケーリングに関する情報を参照してください。

3.3. MICROSOFT AZURE

Microsoft Azure 環境で作成された Red Hat OpenShift Container Platform クラスターのノード数は、ハブクラスターの **MachineSet** パラメーターを修正することで変更できます。

注記: Red Hat Advanced Cluster Mangement for Kubernetes は Hive for OpenShift を使用してクラスター内のノード数を判断するので、**MachineSet** パラメーターを変更してノード数を変える必要があります。**MachineSets** パラメーターを変更せずにノードを削除または追加すると、このパラメーターの現在の値に一致するように、ノードが追加または削除されます。

お使いのバージョンに該当する OpenShift Container Platform ドキュメントの「[クラスターのスケールリングに関する推奨プラクティス](#)」および「[MachineSet の手動によるスケールリング](#)」を参照してください。ヒント: Red Hat Advanced Cluster Management for Kubernetes を使用してクラスターを作成した場合には、これは OpenShift Container Platform クラスターになります。

インポートした Azure Kubernetes Services クラスターにあるノード数を変更する場合には、「[Scaling a cluster](#)」でクラスターのスケールリングに関する情報を参照してください。

3.4. ベアメタルクラスター

ベアメタル環境で作成された Red Hat OpenShift Container Platform クラスターのノード数は、ハブクラスターの **MachineSet** パラメーターを修正することで変更できます。

注記: Red Hat Advanced Cluster Mangement for Kubernetes は Hive for OpenShift を使用してクラスター内のノード数を判断するので、**MachineSet** パラメーターを変更してノード数を変える必要があります。**MachineSets** パラメーターを変更せずにノードを削除または追加すると、このパラメーターの現在の値に一致するように、ノードが追加または削除されます。

お使いのバージョンに該当する OpenShift Container Platform ドキュメントの「[クラスターのスケールリングに関する推奨プラクティス](#)」および「[MachineSet の手動によるスケールリング](#)」を参照してください。ヒント: Red Hat Advanced Cluster Management for Kubernetes を使用してクラスターを作成した場合には、これは OpenShift Container Platform クラスターになります。

インポートしたベアメタルクラスターにあるノード数を変更する場合は、クラスターをスケールリングする方法についての詳細は、「[ネットワークのカスタマイズによるベアメタルへのクラスターのインストール](#)」でクラスターのスケールリングの情報を参照してください。

注記: ベアメタルクラスターは、ハブクラスターが OpenShift Container Platform バージョン 4.5 以降である場合にのみサポートされます。

3.5. IBM KUBERNETES SERVICE

インポートした IBM Kubernetes Service クラスターにあるノード数を変更する場合には、「[Adding worker nodes and zones to clusters](#)」でクラスターのスケールリングに関する情報を参照してください。

注記: Red Hat Advanced Cluster Mangement for Kubernetes は Hive for OpenShift を使用してクラスター内のノード数を判断するので、**MachineSet** パラメーターを変更してノード数を変える必要があります。**MachineSets** パラメーターを変更せずにノードを削除または追加すると、このパラメーターの現在の値に一致するように、ノードが追加または削除されます。

第4章 リリースイメージ

Red Hat Advanced Cluster Management for Kubernetes を使用してプロバイダーでクラスターを作成する場合は、新規クラスターに使用するリリースイメージを指定する必要があります。リリースイメージでは、クラスターのビルドに使用する Red Hat OpenShift Container Platform のバージョンを指定します。

acm-hive-openshift-releases GitHub リポジトリの **yaml** ファイルを使用して、リリースイメージを参照します。Red Hat Advanced Cluster Management for Kubernetes はこれらのファイルを使用して、コンソールで利用可能なリリースイメージの一覧を作成します。リポジトリには、**clusterImageSets** ディレクトリーと **subscription** ディレクトリーが含まれます。これらのディレクトリーは、リリースイメージの操作時に使用します。

clusterImageSets ディレクトリーには以下のディレクトリーが含まれます。

- **Fast**: サポート対象の各 OpenShift Container Platform バージョンのリリースイメージの内、最新のバージョン 2 つを参照するファイルが含まれます。
- **Releases**: サポート対象の各 OpenShift Container Platform バージョンのリリースイメージすべてを参照するファイルが含まれます。**注記**: これらのリリースすべてがテストされおらず、安定版とみなされているわけではありません。
- **Stable**: サポート対象の各 OpenShift Container Platform バージョンのリリースイメージの内、最新の安定版 2 つを参照するファイルが含まれます。このフォルダー内のリリースイメージはテストされ、検証されています。

subscription ディレクトリーには、リリースイメージの一覧がプルされる場所を指定するファイルが含まれます。Red Hat Advanced Cluster Management のデフォルトのリリースイメージは、Quay.io デフォルトで提供されます。これらのリリースイメージは、[acm-hive-openshift-releases GitHub repository](#) のファイルで参照されます。

4.1. 利用可能なリリースイメージの同期

リリースイメージは頻繁に更新されるため、リリースイメージの一覧を同期して、利用可能な最新バージョンを選択できるようにする必要があります。リリースイメージは、[acm-hive-openshift-releases](#) の GitHub リポジトリから入手できます。

リリースイメージの安定性には、以下の 3 つのレベルがあります。

表4.1 リリースイメージの安定性レベル

カテゴリー	説明
stable	完全にテストされたイメージで、クラスターを正常にインストールしてビルドできることが確認されています。
fast	部分的にテスト済みですが、stable バージョンよりも安定性が低い可能性があります。
candidate	テストはしていませんが、最新のイメージです。バグがある可能性もあります。

一覧を更新するには、以下の手順を実行します。

1. [acm-hive-openshift-releases GitHub repository](#) をクローンします。
2. 以下のコマンドを入力して、Red Hat Advanced Cluster Management for Kubernetes のハブクラスターに接続します。

```
oc apply -k subscription/
```

約1分後に、最新の **fast** エントリーが2つ利用可能になります。

3. **acm-hive-openshift-releases** GitHub リポジトリをクローンした後に **stable** リリースイメージの一覧を同期するには、以下のコマンドを入力して **stable** イメージを更新します。

```
make subscribe-stable
```

注記: この **make** コマンドは、Linux または MacOS のオペレーティングシステムを使用している場合のみ実行できます。Windows オペレーティングシステムを使用している場合は、以下のコマンドを実行して **stable** イメージを更新します。

```
oc apply -f subscription-stable
```

このコマンドを実行後に、利用可能な **stable** のリリースイメージの一覧が、現在利用可能なイメージに約1分ほどで更新されます。

- Fast リリースイメージを同期して表示するには、以下のコマンドを実行します。

```
make subscribe-fast
```

注記: この **make** コマンドは、Linux または MacOS のオペレーティングシステムを使用している場合のみ実行できます。Windows オペレーティングシステムを使用している場合は、以下のコマンドを実行して **fast** イメージを更新します。

```
oc apply -f subscription/subscription-fast.yaml
```

このコマンドを実行後に、利用可能な **stable** と **fast** のリリースイメージの一覧が、現在利用可能なイメージに約1分ほどで更新されます。

- **candidate** リリースイメージを同期して表示するには、以下のコマンドを実行します。

```
make subscribe-candidate
```

注記: この **make** コマンドは、Linux または MacOS のオペレーティングシステムを使用している場合のみ実行できます。Windows オペレーティングシステムを使用している場合は、以下のコマンドを実行して **candidate** イメージを更新します。

```
oc apply -f subscription/subscription-candidate.yaml
```

このコマンドを実行後に、利用可能な **stable**、**fast** および **candidate** のリリースイメージの一覧が、現在利用可能なイメージに約1分ほどで更新されます。

4. クラスターの作成時に、Red Hat Advanced Cluster Management コンソールで現在利用可能なリリースイメージの一覧を表示します。
5. 以下の形式でコマンドを入力して、これらのチャンネルのサブスクライブを解除して更新の表示を停止することができます。

```
oc delete -f subscription/subscription-stable
```

4.1.1. 接続時におけるリリースイメージのカスタム一覧の管理

すべてのクラスターに同じリリースイメージが使用されるようにします。クラスターの作成時に利用可能なリリースイメージのカスタム一覧を作成し、作業を簡素化します。利用可能なリリースイメージを管理するには、以下の手順を実行します。

1. [acm-hive-openshift-releases GitHub repository](#) をフォークします。
2. `./subscription/channel.yaml` ファイルを更新して、**open-cluster-management** ではなく、フォークしたリポジトリの GitHub 名にアクセスするように、**spec: pathname** を変更します。この手順では、ハブクラスターによるリリースイメージの取得先を指定します。更新後の内容は以下の例のようになります。

```
spec:
  type: GitHub
  pathname: https://github.com/<forked_content>/acm-hive-openshift-releases.git
```

forked_content はフォークしたリポジトリのへのパスに置き換えます。

3. Red Hat Advanced Cluster Management for Kubernetes を使用して、クラスターの作成時に使用できるように、イメージの **yaml** ファイルを `./clusterImageSets/stable/` または `./clusterImageSets/fast/` に追加します。**ヒント:** フォークしたリポジトリに変更をマージして、メインのリポジトリから利用可能な **yaml** ファイルを取得できます。
4. フォークしたリポジトリに変更をコミットし、マージします。
5. **acm-hive-openshift-releases** リポジトリをクローンした後に `stable` リリースイメージの一覧を同期するには、以下のコマンドを入力して `stable` イメージを更新します。

```
make subscribe-stable
```

注記: この **make** コマンドは、Linux または MacOS のオペレーティングシステムを使用している場合のみ実行できます。Windows オペレーティングシステムを使用している場合は、以下のコマンドを入力します。

```
oc apply -k subscription/
oc delete -f subscription/subscription-fast.yaml
oc apply -f subscription/subscription-stable.yaml
```

このコマンドを実行後に、利用可能な安定版のリリースイメージの一覧が、現在利用可能なイメージに約1分ほどで更新されます。

6. デフォルトでは、安定版のイメージのみが一覧表示されます。Fast リリースイメージを同期して表示するには、以下のコマンドを実行します。

```
make subscribe-fast
```

注記: この **make** コマンドは、Linux または MacOS のオペレーティングシステムを使用している場合のみ実行できます。Windows オペレーティングシステムを使用している場合は、以下のコマンドを入力します。

```
oc apply -k subscription/
oc apply -f subscription/subscription-fast.yaml
```

このコマンドを実行後に、利用可能な fast リリースイメージの一覧が、現在利用可能なイメージに約1分ほどで更新されます。

7. デフォルトではRed Hat Advanced Cluster Management は ClusterImageSets を複数事前に読み込みます。以下のコマンドを使用して、利用可能な ClusterImageSet を表示し、デフォルトの設定を削除します (任意)。

```
oc get clusterImageSets
oc delete clusterImageSet <clusterImageSet_NAME>
```

8. クラスターの作成時に、Red Hat Advanced Cluster Management コンソールで現在利用可能なリリースイメージの一覧を表示します。

4.1.2. 非接続時におけるリリースイメージのカスタム一覧の管理

ハブクラスターにインターネット接続がない場合は、リリースイメージのカスタムリストを管理しないといけない場合があります。クラスターの作成時に利用可能なリリースイメージのカスタム一覧を作成します。非接続時に、利用可能なリリースイメージを管理するには、以下の手順を実行します。

1. オンライン接続されているシステムを使用している場合は、[acm-hive-openshift-releases GitHub repository](#) に移動します。
2. **clusterImageSets** ディレクトリーを、非接続の Red Hat Advanced Cluster Management for Kubernetes ハブクラスターにアクセス可能なシステムにコピーします。
3. **clusterImageSet** yaml を手作業で追加して、Red Hat Advanced Cluster Management for Kubernetes コンソールを使用してクラスターを作成する時に利用できるようにイメージの **yaml** ファイルを追加します。
4. **clusterImageSets** コマンドを作成します。

```
oc create -f <clusterImageSet_FILE>
```

追加するリソースごとにこのコマンドを実行すると、利用可能なリリースイメージの一覧が使用できるようになります。

5. または Red Hat Advanced Cluster Management のクラスター作成のコンソールに直接イメージの URL を貼り付けることもできます。これにより、clusterImageSets が存在しない場合には、新しいものが作成されます。
6. クラスターの作成時に、Red Hat Advanced Cluster Management コンソールで現在利用可能なリリースイメージの一覧を表示します。

第5章 ベアメタルアセットの作成および変更

重要: ベアメタルクラスター機能はテクノロジープレビューであるため、実稼働環境では使用しないでください。

ベアメタルアセットとは、クラウドオペレーションを実行するように設定された仮想サーバーまたは物理サーバーのことです。Red Hat Advanced Cluster Management for Kubernetes は管理者が作成するベアメタルアセットに接続してクラスターを作成できます。

Red Hat Advanced Cluster Management for Kubernetes でベアメタルアセットを作成して、ベアメタルアセットでクラスターを作成する必要があります。以下の手順を使用して、Red Hat Advanced Cluster Management for Kubernetes が管理するクラスターをホストできるベアメタルアセットを作成します。

注記: ベアメタル機能は、テクノロジープレビュー機能としてのみ提供されます。ベアメタルのオプションは、デフォルトで機能フラグにより非表示になります。ベアメタルのオプションを表示するには、「前提条件」のセクションの手順を実行して、機能フラグを有効にする必要があります。

5.1. 前提条件

ベアメタルアセットを作成する前に、以下の前提条件を満たす必要があります。

- OpenShift Container Platform バージョン 4.5 以降に、Red Hat Advanced Cluster Management for Kubernetes ハブクラスターをデプロイしておく。
- Red Hat Advanced Cluster Management for Kubernetes ハブクラスターがベアメタルアセットに接続できるようにアクセスを設定しておく。
- ベアメタルアセットおよび、ベアメタルアセットへのログインまたは管理に必要なパーミッションを指定したログイン認証情報を設定しておく。**注記:** ベアメタルアセットへのログイン認証情報には、管理者が提供する以下のアセットの項目が含まれます。
 - ユーザー名
 - パスワード
 - ベースボード管理コントローラー (BMC) アドレス
 - ブート NIC MAC アドレス
- ベアメタルオプションを表示するために、ベアメタル機能フラグを有効にしておく。ベアメタルのオプションは、デフォルトで機能フラグにより非表示になります。機能フラグを有効にするには、以下の手順を実行します。
 - a. Red Hat OpenShift Container Platform コマンドラインインターフェースを起動します。
 - b. 以下のコマンドを入力して、**console-header** コンテナの **featureFlags_baremetal** 設定を **true** に指定します。

```
oc patch deploy console-header -n <namespace> -p '{"spec":{"template":{"spec":{"containers":[{"name":"console-header","env":[{"name":"featureFlags_baremetal","value":"true"}]}]}}}}'
```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新後、**consoleui** CRD の例は以下のようになります。

```
spec:
  ...
  template:
    ...
    spec:
      ...
      containers:
        - env:                # Search for env:
          - name: featureFlags_baremetal
            value: "true"
      ...
```

- c. **hmc-ui** の **featureFlags_baremetal** の値を **true** に設定します。

```
oc patch -n <namespace> $(oc get deploy -o name | grep consoleui) -p '{"spec":
{"template":{"spec":{"containers":[{"name":"hmc-ui","env": [{"name":
"featureFlags_baremetal","value":"true"}]}]}}}'
```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新の例を以下に示します。

```
spec:
  ...
  template:
    ...
    spec:
      ...
      containers:
        - env:                # Search for env:
          - name: featureFlags_baremetal
            value: "true"
      ...
```

- d. **console-chart-...-consoleui...** と **console-header-...** の Pod が実行されていることを確認します。

```
oc -n open-cluster-management get pods
```

- e. Pod を再度実行したら、Red Hat Advanced Cluster Management for Kubernetes コンソールからログアウトし、再度ログインします。ベアメタルのオプションがコンソールに含まれるようになりました。

5.2. コンソールを使用したベアメタルアセットの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用してベアメタルアセットを作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Bare metal assets** に移動します。
2. **Bare metal assets** ページで **Create bare metal asset** をクリックします。
3. クラスターの作成時に識別できるようにアセット名を入力します。

ヒント: コンソールに情報を入力する時に **yaml** コンテンツの更新内容を表示するには、**YAML** を **ON** に切り替えるように設定します。

4. ベアメタルアセットを作成する namespace を入力します。

注記: ベアメタルアセット、ベアメタルのマネージドクラスター、および関連シークレットは同じ namespace に配置する必要があります。

この namespace にアクセスできるユーザーは、クラスターの作成時にこのアセットをクラスターに関連付けることができます。

5. ベースボード管理コントローラー (BMC) アドレスを入力します。このコントローラーで、ホストとの通信が可能になります。以下のプロトコルがサポートされます。
 - IPMI。詳細は、[IPMI 2.0 Specification](#) を参照してください。
 - iDRAC。詳細は、[Support for Integrated Dell Remote Access Controller 9 \(iDRAC9\)](#) を参照してください。
 - iRMC。詳細は、[Data Sheet: FUJITSU Software ServerView Suite integrated Remote Management Controller - iRMC S5](#) を参照してください。
 - Redfish。詳細は、[Redfish specification](#) を参照してください。
6. ベアメタルのユーザー名とパスワードを入力します。
7. ベアメタルアセットのブート NIC MAC アドレスを追加します。これは、ネットワーク接続されたホストの NIC の MAC アドレスで、ベアメタルアセットにホストをプロビジョニングする時に使用します。

「[ベアメタルでのクラスターの作成](#)」に進んでください。

5.3. ベアメタルアセットの変更

ベアメタルアセットの設定を変更する必要がある場合は、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes コンソールのナビゲーションで、**Automate infrastructure** > **Bare metal assets** を選択します。
2. テーブルで変更するアセットのオプションメニューを選択します。
3. **Modify** を選択します。

5.4. ベアメタルアセットの削除

ベアメタルアセットがどのクラスターにも使用されなくなった場合は、利用可能なベアメタルアセット一覧から削除できます。使用されていないアセットを削除することで、利用可能なアセット一覧が簡素化されて、対象のアセットが誤って選択されないようにします。

ベアメタルアセットを削除するには、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes コンソールのナビゲーションで、**Automate infrastructure** > **Bare metal assets** を選択します。
2. テーブルで削除するアセットのオプションメニューを選択します。
3. **Delete** を選択します。

第6章 プロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes でクラウドサービスプロバイダーに Red Hat OpenShift Container Platform クラスターを作成するには、**プロバイダー接続** が必要です。

プロバイダー接続には、アクセス用の認証情報と、プロバイダーの設定情報が保存されます。プロバイダーアカウントごとに独自のプロバイダー接続が、プロバイダーごとにドメインが必要です。

以下のファイルには、サポートされる各プロバイダーの接続のドキュメントを作成するのに必要な情報がまとめられています。

- [Amazon Web Services のプロバイダー接続の作成](#)
- [Microsoft Azure のプロバイダー接続の作成](#)
- [Google Cloud Platform のプロバイダー接続の作成](#)
- [ベアメタルのプロバイダー接続の作成](#)

6.1. AMAZON WEB SERVICES のプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、Amazon Web Services (AWS) に OpenShift クラスターをデプロイして管理するには、プロバイダー接続が必要です。



注記

Red Hat Advanced Cluster Management for Kubernetes でクラスターを作成する前に、以下の手順を実行する必要があります。

6.1.1. 前提条件

プロバイダー接続を作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく。
- Amazon Web Services で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- アクセスキー ID およびシークレットアクセスキーなど、Amazon Web Services (AWS) のログイン認証情報。[Understanding and getting your AWS credentials](#) を参照してください。
- AWS でクラスターをインストールできるようにするアカウントのパーミッション。設定の方法は、「[AWS アカウントの設定](#)」を参照してください。

6.1.2. コンソールを使用したプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからプロバイダー接続を作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Clusters** ページで **Provider connections** タブを選択します。
既存のプロバイダー接続が表示されます。

3. **Add a connection** を選択します。
4. **Amazon Web Services** をプロバイダーとして選択します。
5. プロバイダー接続の名前を追加します。
6. 一覧からプロバイダー接続の namespace を選択します。

ヒント

便宜上およびセキュリティ強化のため、プロバイダー接続のホスト専用の namespace を作成します。

7. オプションで、プロバイダー接続の **ベース DNS ドメイン** を追加できます。ベース DNS ドメインをプロバイダー接続に追加した場合には、このプロバイダー接続でクラスターを作成すると、このベース DNS ドメインは自動的に正しいフィールドに設定されます。
8. Amazon Web Service アカウントの **AWS Access Key ID** を追加します。[AWS](#) にログインして、ID を検索します。
9. **AWS Secret Access Key ID** を追加します。
10. **Red Hat OpenShift Pull Secret** を入力します。[Pull secret](#) からプルシークレットをダウンロードします。
11. **SSH 秘密鍵** と **SSH 公開鍵** を追加し、クラスターに接続できるようにします。既存のキーペアを使用するか、キー生成プログラムで新しいキーを作成できます。キー生成の方法は、「[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](#)」を参照してください。
12. **Create** をクリックします。プロバイダー接続を作成すると、プロバイダー接続の一覧に追加されます。

「[Amazon Web Services でのクラスターの作成](#)」の手順を実行して、このプロバイダー接続を使用するクラスターを作成します。

6.1.3. プロバイダー接続の削除

プロバイダー接続を使用するクラスターを管理しなくなった場合には、そのプロバイダー接続を削除して、プロバイダー接続の情報を保護します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Provider connections** を選択します。
3. 削除するプロバイダー接続の横にあるオプションメニューを選択します。
4. **Delete connection** を選択します。

6.2. MICROSOFT AZURE のプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、Microsoft Azure で Red Hat OpenShift Container Platform クラスターを作成して管理するには、プロバイダー接続が必要です。



注記

以下の手順は、Red Hat Advanced Cluster Management for Kubernetes でクラスターを作成するための前提条件となっています。

6.2.1. 前提条件

プロバイダー接続を作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく。
- Azure で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- ベースドメインのリソースグループおよび Azure Service Principal JSON などの Azure ログイン認証情報。 azure.microsoft.com を参照してください。
- Azure でクラスターがインストールできるようにするアカウントのパーミッション。詳細は、 [How to configure Cloud Services](#) および「[Azure アカウントの設定](#)」を参照してください。

6.2.2. コンソールを使用したプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからプロバイダー接続を作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Clusters** ページで **Provider connections** タブを選択します。
既存のプロバイダー接続が表示されます。
3. **Add a connection** を選択します。
4. **Microsoft Azure** をプロバイダーとして選択します。
5. プロバイダー接続の名前を追加します。
6. 一覧からプロバイダー接続の namespace を選択します。

ヒント

便宜上およびセキュリティ上、プロバイダー接続のホスト専用の namespace を作成できません。

7. オプションで、プロバイダー接続の **ベース DNS ドメイン** を追加できます。ベース DNS ドメインをプロバイダー接続に追加した場合には、このプロバイダー接続でクラスターを作成すると、このベース DNS ドメインは自動的に正しいフィールドに設定されます。
8. Azure アカウントの **ベースドメインリソースグループ名** を追加します。このエントリは、Azure アカウントで作成したリソース名です。Azure インターフェースで **Home > DNS Zones** を選択することで、ベースドメインのリソースグループ名を検索できます。ベースドメインのリソースグループ名は、アカウントに適用するベース DNS ドメインが含まれるエントリーの **Resource Group** コラムにあります。

9. **Client ID** を追加します。この値は、以下のコマンドを使用してサービスプリンシパルを作成すると、**appld** プロパティとして設定されます。

```
az ad sp create-for-rbac --role Contributor --name <service_principal>
```

service_principal は、お使いのサービスプリンシパル名に置き換えます。

10. **Client Secret** を追加します。この値は、以下のコマンドを使用してサービスプリンシパルを作成すると、**password** プロパティとして設定されます。

```
az ad sp create-for-rbac --role Contributor --name <service_principal>
```

service_principal は、お使いのサービスプリンシパル名に置き換えます。

11. **Subscription ID** を追加します。以下のコマンドの出力では、この値は、**id** プロパティになります。

```
az account show
```

12. **Tenant ID** を追加します。以下のコマンドの出力では、この値は、**tenantId** プロパティになります。

```
az account show
```

13. **Red Hat OpenShift Pull Secret** を入力します。[Pull secret](#) からプルシークレットをダウンロードします。
14. クラスターへの接続に使用する **SSH 秘密鍵** と **SSH 公開鍵** を追加します。既存のキーペアを使用するか、キー生成プログラムで新しいキーを作成できます。キー生成の方法は、「[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](#)」を参照してください。
15. **Create** をクリックします。プロバイダー接続を作成すると、プロバイダー接続の一覧に追加されます。

「[Microsoft Azure でのクラスターの作成](#)」の手順を実行して、このプロバイダー接続を使用するクラスターを作成します。

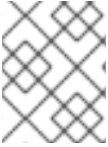
6.2.3. プロバイダー接続の削除

プロバイダー接続を使用するクラスターを管理しなくなった場合には、そのプロバイダー接続を削除して、プロバイダー接続の情報を保護します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Provider connections** を選択します。
3. 削除するプロバイダー接続のオプションメニューを選択します。
4. **Delete connection** を選択します。

6.3. GOOGLE CLOUD PLATFORM のプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、Google Cloud Platform (GCP) で Red Hat OpenShift Container Platform クラスターを作成して管理するには、プロバイダー接続が必要です。



注記

以下の手順は、Red Hat Advanced Cluster Management for Kubernetes でクラスターを作成するための前提条件となっています。

6.3.1. 前提条件

プロバイダー接続を作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく。
- Management for Kubernetes ハブクラスターでのインターネットアクセスを有効にし、GCP で Kubernetes クラスターを作成できるようにする。
- ユーザーの Google Cloud Platform プロジェクト ID および Google Cloud Platform サービスアカウント JSON キーなど、GCP ログインの認証情報。 [Creating and managing projects](#) を参照してください。
- GCP でクラスターがインストールできるようにするアカウントのパーミッション。アカウントの設定方法は、「[GCP プロジェクトの設定](#)」を参照してください。

6.3.2. コンソールを使用したプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからプロバイダー接続を作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. Clusters ページで **Provider connections** タブを選択します。
既存のプロバイダー接続が表示されます。
3. **Add a connection** を選択します。
4. **Google Cloud Platform** をプロバイダーとして選択します。
5. プロバイダー接続の名前を追加します。
6. 一覧からプロバイダー接続の namespace を選択します。

ヒント

便宜上およびセキュリティ上、プロバイダー接続のホスト専用の namespace を作成します。

7. オプションで、プロバイダー接続の **ベース DNS ドメイン** を追加できます。ベース DNS ドメインをプロバイダー接続に追加した場合には、このプロバイダー接続でクラスターを作成すると、このベース DNS ドメインは自動的に正しいフィールドに設定されます。
8. GCP アカウントの **Google Cloud Platform Project ID** を追加します。 [GCP](#) にログインして設定を取得します。
9. **Google Cloud Platform service account JSON key** を追加します。適切なパーミッションでキーを作成するには、以下の手順を実行します。
 - a. GCP のメインメニューで **IAM & Admin** を選択して、**Service Accounts applet** を起動します。

- b. **Create Service Account**を選択します。
 - c. サービスアカウントの **Name**、**Service account ID**および **Description** を指定します。
 - d. **Create** を選択してサービスアカウントを作成します。
 - e. **Owner** のロールを選択し、**Continue** をクリックします。
 - f. **Create Key** をクリックします。
 - g. **JSON** を選択して、**Create** をクリックします。
 - h. 作成されたファイルをコンピューターに保存します。
 - i. **Google Cloud Platform service account JSON key**のコンテンツを指定します。
10. **Red Hat OpenShift Pull Secret**を入力します。 [Pull secret](#) からプルシークレットをダウンロードします。
 11. クラスターにアクセスできるように **SSH 秘密鍵**と **SSH 公開鍵**を追加します。既存のキーペアを使用するか、キー生成プログラムで新しいキーを作成できます。キー生成の方法は、「[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](#)」を参照してください。
 12. **Create** をクリックします。プロバイダー接続を作成すると、プロバイダー接続の一覧に追加されます。

「[Google Cloud Platform でのクラスターの作成](#)」の手順を実行して、**クラスターの作成時に** このコネクションを使用できます。

6.3.3. プロバイダー接続の削除

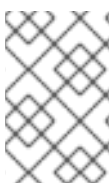
プロバイダー接続を使用するクラスターを管理しなくなった場合には、そのプロバイダー接続を削除して、プロバイダー接続の情報を保護します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Provider connections** を選択します。
3. 削除するプロバイダー接続の横にあるオプションメニューを選択します。
4. **Delete connection** を選択します。

6.4. ベアメタルのプロバイダー接続の作成

重要: ベアメタルクラスター機能はテクノロジープレビューであるため、実稼働環境では使用しないでください。

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、ベアメタル環境で Red Hat OpenShift Container Platform クラスターを作成して管理するには、プロバイダー接続が必要です。



注記

コンソールでのベアメタルのオプションはテクノロジープレビュー機能としてのみ提供され、デフォルトでは機能フラグにより非表示になっています。**前提条件** セクションで機能フラグを有効にする手順を参照してください。

6.4.1. 前提条件

プロバイダー接続を作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく。ベアメタルクラスターを管理する場合は、Red Hat OpenShift Container Platform バージョン 4.5 以降に、ハブクラスターをインストールする必要があります。
- ベアメタルサーバーで Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- libvirt URI、SSH、秘密鍵、SSH の既知のホスト一覧など、ベアメタルサーバーのログイン認証情報。「[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](#)」を参照してください。
- ベアメタルインフラストラクチャーでクラスターがインストールできるようにするアカウントのパーミッション。
- ベアメタルオプションを表示するために、ベアメタル機能フラグを有効にしておく。ベアメタルのオプションは、デフォルトで機能フラグにより非表示になります。機能フラグを有効にするには、以下の手順を実行します。
 - a. Red Hat OpenShift Container Platform コマンドラインインターフェースを起動します。
 - b. 以下のコマンドを入力して、**console-header** コンテナの **featureFlags_baremetal** 設定を **true** に指定します。

```
oc patch deploy console-header -n <namespace> -p '{"spec":{"template":{"spec":{"containers":[{"name":"console-header","env":[{"name":"featureFlags_baremetal","value":"true"}]}]}}}}'
```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新後、**consoleui** CRD の例は以下のようになります。

```
spec:
  ...
  template:
    ...
    spec:
      ...
      containers:
        - env:                # Search for env:
          - name: featureFlags_baremetal
            value: "true"
      ...
```

- c. **hmc-ui** の **featureFlags_baremetal** の値を **true** に設定します。

```
oc patch -n <namespace> $(oc get deploy -o name | grep consoleui) -p '{"spec":{"template":{"spec":{"containers":[{"name":"hmc-ui","env":[{"name":"featureFlags_baremetal","value":"true"}]}]}}}}'
```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新の例を以下に示します。

```
spec:
  ...
  template:
    ...
    spec:
      ...
      containers:
        - env:          # Search for env:
          - name: featureFlags_baremetal
            value: "true"
      ...
```

- d. **console-chart-...-consoleui...** と **console-header-...** の Pod が実行されていることを確認します。

```
oc -n open-cluster-management get pods
```

- e. Pod を再度実行したら、Red Hat Advanced Cluster Management for Kubernetes コンソールからログアウトし、再度ログインします。ベアメタルのオプションがコンソールに含まれるようになりました。

6.4.2. コンソールを使用したプロバイダー接続の作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからプロバイダー接続を作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Clusters** ページで **Provider connections** タブを選択します。
既存のプロバイダー接続が表示されます。
3. **Add connection** を選択します。
4. プロバイダーとして **Bare metal** を選択します。
5. プロバイダー接続の名前を追加します。
6. 一覧からプロバイダー接続の namespace を選択します。
ヒント: 便宜上およびセキュリティ上、プロバイダー接続のホスト専用の namespace を作成します。
7. オプションで、プロバイダー接続の **ベース DNS ドメイン** を追加できます。ベース DNS ドメインをプロバイダー接続に追加した場合には、このプロバイダー接続でクラスターを作成すると、このベース DNS ドメインは自動的に正しいフィールドに設定されます。
8. **libvirt URI** を追加します。詳細は、[Connection URIs](#) を参照してください。
9. **Red Hat OpenShift Pull Secret** を入力します。[Pull secret](#) からプルシークレットをダウンロードします。
10. クラスターにアクセスできるように **SSH 秘密鍵** と **SSH 公開鍵** を追加します。既存のキーを使用するか、キー生成プログラムを使用して新しいキーを作成できます。キー生成の方法は、「[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](#)」を参照してください。

11. SSH 既知のホスト一覧を追加します。
12. オフラインインストールのみ: **Configuration for disconnected installation**サブセクションのフィールドに必要な情報を入力します。
 - **Image Registry Mirror:** このオプションの値には、オフラインのレジストリーパスを含みます。このパスには、オフラインインストールに使用する全インストールイメージのホスト名、ポート、レジストリーパスが含まれます。例: **repository.com:5000/openshift/ocp-release**。
 - **Bootstrap OS Image:** この値には、ブートストラップマシンに使用するイメージの URL が含まれます。
 - **Cluster OS Image:** この値には、Red Hat OpenShift Container Platform クラスタマシンに使用するイメージの URL が含まれます。
 - **Additional Trust Bundle:** この値で、ミラーレジストリーへのアクセスに必要な証明書ファイルのコンテンツを指定します。
13. **Create** をクリックします。プロバイダー接続を作成すると、プロバイダー接続の一覧に追加されます。

「[ベアメタルでのクラスタの作成](#)」の手順を実行して、このプロバイダー接続を使用するクラスタを作成します。

6.4.3. プロバイダー接続の削除

プロバイダー接続を使用するクラスタを管理しなくなった場合には、そのプロバイダー接続を削除して、プロバイダー接続の情報を保護します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Provider connections** を選択します。
3. 削除するプロバイダー接続の横にあるオプションメニューを選択します。
4. **Delete connection** を選択します。

第7章 RED HAT ADVANCED CLUSTER MANAGEMENT FOR KUBERNETES でのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes を使用した、クラウドプロバイダー全体にクラスターを作成する方法を説明します。

- [Amazon Web Services でのクラスターの作成](#)
- [Google Cloud Platform でのクラスターの作成](#)
- [Microsoft Azure でのクラスターの作成](#)
- [ベアメタルでのクラスターの作成](#)（Red Hat OpenShift Container Platform バージョン 4.4 以降が必要）

7.1. AMAZON WEB SERVICES でのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、Amazon Web Services (AWS) で Red Hat OpenShift Container Platform クラスターを作成できます。

7.1.1. 前提条件

AWS でクラスターを作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしている。
- Amazon Web Services で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- AWS プロバイダー接続。詳細は、「[Amazon Web Services のプロバイダー接続の作成](#)」を参照してください。
- AWS で設定されたドメイン。ドメインの設定方法は、「[AWS アカウントの設定](#)」を参照してください。
- ユーザー名、パスワード、アクセスキー ID およびシークレットアクセスキーなど、Amazon Web Services (AWS) のログイン認証情報。[Understanding and Getting Your Security Credentials](#) を参照してください。
- Red Hat OpenShift イメージプルシークレット。「[イメージプルシークレットの使用](#)」を参照してください。

注記: クラウドプロバイダーのアクセスキーを変更した場合に、プロビジョニングしたクラスターアクセスキーを手動で更新する必要があります。詳細は、既知の問題「[プロビジョニングしたクラスターのシークレットの自動更新はサポートされない](#)」を参照してください。

7.1.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからクラスターを作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。

2. クラスターページで **Add Cluster** をクリックします。

3. **Create a cluster** を選択します。



注記

この手順では、クラスターを作成します。既存のクラスターをインポートする場合には、「[ハブクラスターへのターゲットのマネージドクラスターのインポート](#)」の手順を参照してください。

4. クラスターの名前を入力します。この名前はクラスターのホスト名で使用されます。

ヒント

コンソールに情報を入力する時に **yaml** コンテンツの更新内容を表示するには、**YAML** を **ON** に切り替えるように設定します。

5. インフラストラクチャプラットフォーム向けの **Amazon Web Services** を選択します。

6. クラスターに使用する **リリースイメージ** を指定します。このリリースイメージで、クラスターの作成に使用される Red Hat OpenShift Container Platform イメージのバージョンを特定します。使用するバージョンが利用可能な場合は、イメージの一覧からイメージを選択できます。標準イメージ以外のイメージを使用する場合は、使用するイメージへの url を入力できます。リリース [イメージ](#) の詳細は、「リリースイメージ」を参照してください。

7. 利用可能な接続一覧から、お使いのプロバイダー接続を選択します。プロバイダー接続が設定されていない場合や、新規のプロバイダー接続を設定する場合には、**Add connection** を参照してください。[プロバイダー接続の作成に関する詳細は](#)、「[Amazon Web Services](#) のプロバイダー接続の作成」を参照してください。

8. AWS アカウントに設定したベースドメイン情報を入力します。選択したプロバイダー接続にベースドメインが紐付けされている場合には、その値がこのフィールドに設定されます。値を上書きすると変更できます。詳細は、「[AWS アカウントの設定](#)」を参照してください。この名前はクラスターのホスト名で使用されます。

9. クラスターに関連付ける **Label** を追加します。これらのラベルは、クラスターを特定し、検索結果を絞り込むのに役立ちます。

10. クラスターの **ノードプール** を設定します。

ノードプールは、クラスターに使用されるノードの場所とサイズを定義します。

Region は、ノードの地理的な配置場所を指定します。リージョンが近くにある場合にはパフォーマンスの速度が向上しますが、リージョンの距離が離れると、より分散されます。

- **マスタープール**: マスタープールには、クラスター向けに作成されたマスターノードが3つあります。マスターノードは、クラスターアクティビティの管理を共有します。より分散されているマスターノードグループでは、リージョンで複数のゾーンを選択できます。インスタンスの作成後にインスタンスのタイプやサイズを変更できますが、このセクションで指定することもできます。デフォルト値は、ルートストレージ 500 GiB の **mx4.xlarge - 4 vCPU, 16 GiB RAM - General Purpose** です。
- **ワーカープール**: ワーカープールに1つまたは複数のワーカーノードを作成し、クラスターのコンテナワークロードを実行できます。ワーカーノードは、ワーカープール1つに所属することも、複数のワーカープールに分散させることもできます。

11. **オプション**: クラスターネットワークオプションを設定します。
12. **オプション**: クラスターのラベルを設定します。
13. **Create** をクリックします。作成およびインポートプロセスを完了すると、クラスターの詳細を表示できます。



注記

クラスターのインポートには、クラスターの詳細で提示された **kubectl** コマンドを実行する必要はありません。クラスターを作成すると、Red Hat Advanced Cluster Management for Kubernetes で管理されるように自動的に設定されます。

7.1.3. クラスターへのアクセス

Red Hat Advanced Cluster Management for Kubernetes で管理されるクラスターにアクセスするには、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes ナビゲーションメニューで **Automate infrastructure** > **Clusters** に移動します。
2. 作成したクラスターまたはアクセスするクラスターの名前を選択します。クラスターの詳細が表示されます。
3. **Reveal credentials** を選択し、クラスターのユーザー名およびパスワードを表示します。クラスターにログインする際に使用するため、この値を書き留めてください。
4. クラスターにリンクする **Console URL** を選択します。
5. 手順 3 で確認したユーザー ID およびパスワードを使用して、クラスターにログインします。
6. アクセスするクラスターの **Actions** メニューを選択します。
7. **Launch to cluster** を選択します。

ヒント

すでにログイン情報が分かっている場合には、クラスターの **Actions** メニューにアクセスして、**Launch to cluster** を選択します。

7.1.4. マネージメントからのクラスターの削除

Red Hat Advanced Cluster Management for Kubernetes で作成したマネージメントから、Red Hat OpenShift Container Platform クラスターを削除すると、このクラスターを**デタッチ**するか、**破棄**できます。

クラスターをデタッチするとマネージメントから削除されますが、完全に削除されません。管理下に戻す場合には、もう一度インポートし直すことができます。このオプションは、クラスターが **Ready** 状態にある場合にだけ利用できます。

クラスターを破棄すると、マネージメントから削除され、クラスターのコンポーネントが削除されます。こちらは、完全に削除されるので、削除後に管理下に戻すことはできません。

1. ナビゲーションメニューから **Automate infrastructure** > **Clusters** に移動します。

2. 削除するクラスターの横にあるオプションメニューを選択します。
3. **Destroy cluster** または **Detach cluster** を選択します。

ヒント

複数のクラスターをデタッチまたは破棄するには、デタッチまたは破棄するクラスターのチェックボックスを選択します。次に、**Detach** または **Destroy** を選択します。

7.2. MICROSOFT AZURE でのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、Microsoft Azure で Red Hat OpenShift Container Platform クラスターを作成できます。

7.2.1. 前提条件

Azure でクラスターを作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしている。
- Azure で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- Azure プロバイダー接続。詳細は、「[Microsoft Azure のプロバイダー接続の作成](#)」を参照してください。
- Azure で設定されたドメイン。ドメイン設定の方法は、[Configuring a custom domain name for an Azure cloud service](#) を参照してください。
- ユーザー名とパスワードなどの Azure ログイン認証情報。[azure.microsoft.com](#) を参照してください。
- **clientId**、**clientSecret** および **tenantId** などの Azure サービスプリンシパル。[azure.microsoft.com](#) を参照してください。
- Red Hat OpenShift イメージプルシークレット。「[イメージプルシークレットの使用](#)」を参照してください。

注記: クラウドプロバイダーのアクセスキーを変更した場合に、プロビジョニングしたクラスターアクセスキーを手動で更新する必要があります。詳細は、既知の問題「[プロビジョニングしたクラスターのシークレットの自動更新はサポートされない](#)」を参照してください。

7.2.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからクラスターを作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Clusters** ページで、**Add cluster** をクリックします。
3. **Create a cluster** を選択します。



注記

この手順では、クラスターを作成します。既存のクラスターをインポートする場合には、「[ハブクラスターへのターゲットのマネージドクラスターのインポート](#)」の手順を参照してください。

4. クラスターの名前を入力します。この名前はクラスターのホスト名で使用されます。

ヒント

コンソールに情報を入力する時に **yaml** コンテンツの更新内容を表示するには、**YAML** を **ON** に切り替えるように設定します。

5. インフラストラクチャプラットフォームに **Microsoft Azure** を選択します。
6. クラスターに使用する **リリースイメージ** を指定します。このリリースイメージで、クラスターの作成に使用される Red Hat OpenShift Container Platform イメージのバージョンを特定します。使用するバージョンが利用可能な場合は、イメージの一覧からイメージを選択できます。標準イメージ以外のイメージを使用する場合は、使用するイメージへの url を入力できます。リリース [イメージ](#) の詳細は、「リリースイメージ」を参照してください。
7. 利用可能な接続一覧から、お使いのプロバイダー接続を選択します。プロバイダー接続が設定されていない場合や、新規のプロバイダー接続を設定する場合には、**Add connection** を参照してください。[プロバイダー接続の作成に関する詳細は](#)、「[Microsoft Azure のプロバイダー接続の作成](#)」を参照してください。
8. Azure アカウントに設定したベースドメイン情報を入力します。選択したプロバイダー接続にベースドメインが紐付けされている場合には、その値がこのフィールドに設定されます。値を上書きすると変更できます。詳細は、[Configuring a custom domain name for an Azure cloud service](#) を参照してください。この名前はクラスターのホスト名で使用されます。
9. クラスターに関連付ける **Label** を追加します。これらのラベルは、クラスターを特定し、検索結果を絞り込むのに役立ちます。
10. クラスターの **ノードプール** を設定します。
ノードプールは、クラスターに使用されるノードの場所とサイズを定義します。

Region は、ノードの地理的な配置場所を指定します。リージョンが近くにある場合にはパフォーマンスの速度が向上しますが、リージョンの距離が離れると、より分散されます。

- **マスタープール:** マスタープールには、クラスター向けに作成されたマスターノードが3つあります。マスターノードは、クラスターアクティビティの管理を共有します。より分散されているマスターノードグループでは、リージョンで複数のゾーンを選択できます。インスタンスの作成後にインスタンスのタイプやサイズを変更できますが、このセクションで指定することもできます。デフォルト値は、ルートストレージ 512 GiB の **Standard_D2s_v3 - 2 vCPU, 8 GiB RAM - General Purpose** です。
- **ワーカープール:** ワーカープールに1つまたは複数のワーカーノードを作成し、クラスターのコンテナワークロードを実行できます。ワーカーノードは、ワーカープール1つに所属することも、複数のワーカープールに分散させることもできます。

11. **オプション:** クラスターネットワークオプションを設定します。
12. **オプション:** クラスターのラベルを設定します。

13. **Create** をクリックします。作成およびインポートプロセスを完了すると、クラスターの詳細を表示できます。



注記

クラスターのインポートには、クラスターの詳細で提示された **kubectli** コマンドを実行する必要はありません。クラスターを作成すると、Red Hat Advanced Cluster Management for Kubernetes で管理されるように自動的に設定されます。

7.2.3. クラスターへのアクセス

Red Hat Advanced Cluster Management for Kubernetes で管理されるクラスターにアクセスするには、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes ナビゲーションメニューで **Automate infrastructure** > **Clusters** に移動します。
2. 作成したクラスターまたはアクセスするクラスターの名前を選択します。クラスターの詳細が表示されます。
3. **Reveal credentials** を選択し、クラスターのユーザー名およびパスワードを表示します。クラスターにログインする際に使用するため、この値を書き留めてください。
4. クラスターにリンクする **Console URL** を選択します。
5. 手順3で確認したユーザー ID およびパスワードを使用して、クラスターにログインします。
6. アクセスするクラスターの **Actions** メニューを選択します。
7. **Launch to cluster** を選択します。

ヒント

すでにログイン情報が分かっている場合には、クラスターの **Actions** メニューにアクセスして、**Launch to cluster** を選択します。

7.2.4. マネージメントからのクラスターの削除

Red Hat Advanced Cluster Management for Kubernetes で作成したマネージメントから、Red Hat OpenShift Container Platform クラスターを削除すると、このクラスターを**デタッチ**するか、**破棄**できます。

クラスターをデタッチするとマネージメントから削除されますが、完全に削除されません。管理下に戻す場合には、もう一度インポートし直すことができます。このオプションは、クラスターが **Ready** 状態にある場合にだけ利用できます。

クラスターを破棄すると、マネージメントから削除され、クラスターのコンポーネントが削除されます。こちらは、完全に削除されるので、削除後に管理下に戻すことはできません。

1. ナビゲーションメニューから **Automate infrastructure** > **Clusters** に移動します。
2. 削除するクラスターの横にあるオプションメニューを選択します。
3. **Destroy cluster** または **Detach cluster** を選択します。

ヒント

複数のクラスターをデタッチまたは破棄するには、デタッチまたは破棄するクラスターのチェックボックスを選択します。次に、**Detach** または **Destroy** を選択します。

7.3. GOOGLE CLOUD PLATFORM でのクラスターの作成

Google Cloud Platform (GCP) で Red Hat OpenShift Container Platform クラスターを作成する手順に従います。Google Cloud Platform の詳細は、[Google Cloud Platform](#) を参照してください。

7.3.1. 前提条件

GCP でクラスターを作成する前に、以下の前提条件を満たす必要があります。

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしている。
- GCP で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセスがある。
- GCP プロバイダー接続。詳細は、「[Google Cloud Platform のプロバイダー接続の作成](#)」を参照してください。
- GCP に設定されたドメイン。ドメインの設定方法は、[Setting up a custom domain](#) を参照してください。
- ユーザー名とパスワードなどの GCP ログイン認証情報。
- Red Hat OpenShift イメージプルシークレット。「[イメージプルシークレットの使用](#)」を参照してください。

注記: クラウドプロバイダーのアクセスキーを変更した場合に、プロビジョニングしたクラスターアクセスキーを手動で更新する必要があります。詳細は、既知の問題「[プロビジョニングしたクラスターのシークレットの自動更新はサポートされない](#)」を参照してください。

7.3.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからクラスターを作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. **Clusters** ページで、**Add cluster** をクリックします。
3. **Create a cluster** を選択します。



注記

この手順では、クラスターを作成します。既存のクラスターをインポートする場合には、「[ハブクラスターへのターゲットのマネージドクラスターのインポート](#)」の手順を参照してください。

4. クラスターの名前を入力します。GCP クラスターの命名に適用される制限がいくつかあります。この制限には、名前を **goog** で開始しないことや、名前に **google** に類似する文字および

数字のグループが含まれないことなどがあります。制限の完全な一覧は、「[Bucket naming guidelines](#)」を参照してください。

この名前はクラスターのホスト名で使用されます。

ヒント

コンソールに情報を入力する時に **yaml** コンテンツの更新内容を表示するには、**YAML** を **ON** に切り替えるように設定します。

5. インフラストラクチャプラットフォームに **Google Cloud** を選択します。
6. クラスターに使用する **リリースイメージ** を指定します。このリリースイメージで、クラスターの作成に使用される Red Hat OpenShift Container Platform イメージのバージョンを特定します。使用するバージョンが利用可能な場合は、イメージの一覧からイメージを選択できます。標準イメージ以外のイメージを使用する場合は、使用するイメージへの url を入力できます。リリース **イメージ** の詳細は、「リリースイメージ」を参照してください。
7. 利用可能な接続一覧から、お使いのプロバイダー接続を選択します。プロバイダー接続が設定されていない場合や、新規のプロバイダー接続を設定する場合には、**Add connection** を参照してください。プロバイダー接続の作成に関する詳細は、「[Google Cloud Platform のプロバイダー接続の作成](#)」を参照してください。
8. Google Cloud Platform アカウントに設定したベースドメイン情報を入力します。選択したプロバイダー接続にベースドメインが紐付けされている場合には、その値がこのフィールドに設定されます。値を上書きすると変更できます。詳細は、[Setting up a custom domain](#) を参照してください。この名前はクラスターのホスト名で使用されます。
9. クラスターに関連付ける **Label** を追加します。これらのラベルは、クラスターを特定し、検索結果を絞り込むのに役立ちます。
10. クラスターの **ノードプール** を設定します。
ノードプールは、クラスターに使用されるノードの場所とサイズを定義します。

Region は、ノードの地理的な配置場所を指定します。リージョンが近くにある場合にはパフォーマンスの速度が向上しますが、リージョンの距離が離れると、より分散されます。

- **マスタープール**: マスタープールには、クラスター向けに作成されたマスターノードが3つあります。マスターノードは、クラスターアクティビティの管理を共有します。より分散されているマスターノードグループでは、リージョンで複数のゾーンを選択できます。インスタンスの作成後にインスタンスのタイプやサイズを変更できますが、このセクションで指定することもできます。デフォルト値は、ルートストレージ 500 GiB の **n1-standard-1 - n1-standard-11 vCPU - General Purpose** です。
 - **ワーカープール**: ワーカープールに1つまたは複数のワーカーノードを作成し、クラスターのコンテナワークロードを実行できます。ワーカーノードは、ワーカープール1つに所属することも、複数のワーカープールに分散させることもできます。
11. **オプション**: クラスターネットワークオプションを設定します。
 12. **オプション**: クラスターのラベルを設定します。
 13. **Create** をクリックします。

作成およびインポートプロセスを完了すると、クラスターの詳細を表示できます。

+ 注記: クラスターのインポートには、クラスターの詳細で提示された **kubectrl** コマンドを実行する必要はありません。クラスターを作成すると、Red Hat Advanced Cluster Management for Kubernetes で管理されるように自動的に設定されます。

7.3.3. クラスターへのアクセス

Red Hat Advanced Cluster Management for Kubernetes で管理されるクラスターにアクセスするには、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes ナビゲーションメニューで **Automate infrastructure** > **Clusters** に移動します。
2. 作成したクラスターまたはアクセスするクラスターの名前を選択します。クラスターの詳細が表示されます。
3. **Reveal credentials** を選択し、クラスターのユーザー名およびパスワードを表示します。クラスターにログインする際に使用するため、この値を書き留めてください。
4. クラスターにリンクする **Console URL** を選択します。
5. 手順 3 で確認したユーザー ID およびパスワードを使用して、クラスターにログインします。
6. アクセスするクラスターの **Actions** メニューを選択します。
7. **Launch to cluster** を選択します。

ヒント

すでにログイン情報が分かっている場合には、クラスターの **Actions** メニューにアクセスして、**Launch to cluster** を選択します。

7.3.4. マネージメントからのクラスターの削除

Red Hat Advanced Cluster Management for Kubernetes で作成したマネージメントから、Red Hat OpenShift Container Platform クラスターを削除すると、このクラスターを**デタッチ**するか、**破棄**できます。

クラスターをデタッチするとマネージメントから削除されますが、完全に削除されません。管理下に戻す場合には、もう一度インポートし直すことができます。このオプションは、クラスターが **Ready** 状態にある場合にだけ利用できます。

クラスターを破棄すると、マネージメントから削除され、クラスターのコンポーネントが削除されます。こちらは、完全に削除されるので、削除後に管理下に戻すことはできません。

1. ナビゲーションメニューから **Automate infrastructure** > **Clusters** に移動します。
2. 削除するクラスターの横にあるオプションメニューを選択します。
3. **Destroy cluster** または **Detach cluster** を選択します。

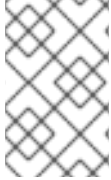
ヒント

複数のクラスターをデタッチまたは破棄するには、デタッチまたは破棄するクラスターのチェックボックスを選択します。次に、**Detach** または **Destroy** を選択します。

7.4. ベアメタルでのクラスターの作成

重要: ベアメタルクラスター機能はテクノロジープレビューであるため、実稼働環境では使用しないでください。

Red Hat Advanced Cluster Management for Kubernetes コンソールを使用して、ベアメタル環境で Red Hat OpenShift Container Platform クラスターを作成できます。



注記

コンソールでのベアメタルのオプションはテクノロジープレビュー機能としてのみ提供され、デフォルトでは機能フラグにより非表示になっています。**前提条件** セクションで機能フラグを有効にする手順を参照してください。

7.4.1. 前提条件

ベアメタル環境にクラスターを作成する前に、以下の前提条件を満たす必要があります。

- OpenShift Container Platform バージョン 4.5 以降に、Red Hat Advanced Cluster Management for Kubernetes ハブクラスターをデプロイしておく。
- ベアメタル環境で Kubernetes クラスターを作成できるようにする Red Hat Advanced Cluster Management for Kubernetes ハブクラスターでのインターネットアクセス
- ベアメタルのプロバイダー接続。詳細は、「[ベアメタルのプロバイダー接続の作成](#)」を参照してください。
- ユーザー名、パスワード、ベースボード管理コントローラー (BMC) アドレスなどのベアメタル環境のログイン認証情報。
- Red Hat OpenShift Container Platform イメージプルシークレット。「[イメージプルシークレットの使用](#)」を参照してください。
注記: ベアメタルアセット、ベアメタルのマネージドクラスター、および関連シークレットは同じ namespace に配置する必要があります。
- ベアメタルオプションを表示するために、ベアメタル機能フラグを有効にしておく。ベアメタルのオプションは、デフォルトで機能フラグにより非表示になります。機能フラグを有効にするには、以下の手順を実行します。

- a. Red Hat OpenShift Container Platform コマンドラインインターフェースを起動します。
- b. 以下のコマンドを入力して、**console-header** コンテナの **featureFlags_baremetal** 設定を **true** に指定します。

```
oc patch deploy console-header -n <namespace> -p '{"spec":{"template":{"spec":{"containers":[{"name":"console-header","env":[{"name":"featureFlags_baremetal","value":"true"}]}]}}}}'
```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新後、**consoleui** CRD の例は以下のようになります。

```
spec:
  ...
```

```

template:
...
spec:
...
containers:
- env:                # Search for env:
  - name: featureFlags_baremetal
    value: "true"
...

```

- c. **hmc-ui** の **featureFlags_baremetal** の値を **true** に設定します。

```

oc patch -n <namespace> $(oc get deploy -o name | grep consoleui) -p '{"spec":
{"template":{"spec":{"containers":[{"name":"hmc-ui","env":[{"name":
"featureFlags_baremetal","value":"true"}]}]}}}'

```

<namespace> は Red Hat Advanced Cluster Management プロジェクトの namespace に置き換えます。

更新の例を以下に示します。

```

spec:
...
template:
...
spec:
...
containers:
- env:                # Search for env:
  - name: featureFlags_baremetal
    value: "true"
...

```

- d. **console-chart-...-consoleui...** と **console-header-...** の Pod が実行されていることを確認します。

```

oc -n open-cluster-management get pods

```

- e. Pod を再度実行したら、Red Hat Advanced Cluster Management for Kubernetes コンソールからログアウトし、再度ログインします。ベアメタルのオプションがコンソールに含まれるようになりました。

注記: クラウドプロバイダーのアクセスキーを変更する場合は、プロビジョニングしたクラスターアクセスキーを手動で更新する必要があります。詳細は、既知の問題「[プロビジョニングしたクラスターのシークレットの自動更新はサポートされない](#)」を参照してください。

7.4.2. Red Hat Advanced Cluster Management for Kubernetes コンソールでのクラスターの作成

Red Hat Advanced Cluster Management for Kubernetes コンソールからクラスターを作成するには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。

2. クラスターページで **Add Cluster** をクリックします。
3. **Create a cluster** を選択します。
注記: この手順では、クラスターを作成します。既存のクラスターをインポートする場合には、「[ハブクラスターへのターゲットのマネージドクラスターのインポート](#)」の手順を参照してください。
4. クラスターの名前を入力します。この名前はクラスターのホスト名で使用されます。
ヒント: コンソールに情報を入力する時に **yaml** コンテンツの更新内容を表示するには、**YAML** を **ON** に切り替えるように設定します。
5. インフラストラクチャプラットフォームに **Bare Metal** を選択します。
6. クラスターに使用する **リリースイメージ** を指定します。このリリースイメージで、クラスターの作成に使用される Red Hat OpenShift Container Platform イメージのバージョンを特定します。使用するバージョンが利用可能な場合は、イメージの一覧からイメージを選択できます。標準イメージ以外のイメージを使用する場合は、使用するイメージへの url を入力できます。リリース [イメージ](#) の詳細は、「リリースイメージ」を参照してください。
7. 利用可能な接続一覧から、お使いのプロバイダー接続を選択します。プロバイダー接続が設定されていない場合や、新規のプロバイダー接続を設定する場合には、**Add provider** を選択してください。[プロバイダー接続の作成に関する詳細は、「ベアメタルのプロバイダー接続の作成」](#)を参照してください。
8. ベアメタル環境に設定したベースドメイン情報を入力します。選択したプロバイダー接続にベースドメインが紐付けされている場合には、その値がこのフィールドに設定されます。値を上書きすると変更できます。この名前はクラスターのホスト名で使用されます。
9. プロバイダー接続に関連付けられたホスト一覧から、お使いのホストを選択します。ハイパーバイザーと同じブリッジネットワークにあるアセットを3つ以上選択します。
10. **オプション:** クラスターネットワークオプションを設定します。
11. **オプション:** クラスターのラベルを設定します。
12. **オプション:** Configmap の追加設定を変更する場合には、詳細設定を更新します。
13. **Create** をクリックします。作成およびインポートプロセスを完了すると、クラスターの詳細を表示できます。
注記: クラスターのインポートには、クラスターの詳細で提示された **kubectrl** コマンドを実行する必要はありません。クラスターを作成すると、Red Hat Advanced Cluster Management for Kubernetes で管理されるように自動的に設定されます。

7.4.3. クラスターへのアクセス

Red Hat Advanced Cluster Management for Kubernetes で管理されるクラスターにアクセスするには、以下の手順を実行します。

1. Red Hat Advanced Cluster Management for Kubernetes ナビゲーションメニューで **Automate infrastructure** > **Clusters** に移動します。
2. 作成したクラスターまたはアクセスするクラスターの名前を選択します。クラスターの詳細が表示されます。
3. **Reveal credentials** を選択し、クラスターのユーザー名およびパスワードを表示します。クラスターにログインする際に使用するため、この値を書き留めてください。

4. クラスターにリンクする **Console URL** を選択します。
5. 手順 3 で確認したユーザー ID およびパスワードを使用して、クラスターにログインします。
6. アクセスするクラスターの **Actions** メニューを選択します。
7. **Launch to cluster** を選択します。
ヒント: すでにログイン情報が分かっている場合には、クラスターの **Actions** メニューにアクセスして、**Launch to cluster** を選択します。

7.4.4. マネージメントからのクラスターの削除

Red Hat Advanced Cluster Management for Kubernetes で作成したマネージメントから、Red Hat OpenShift Container Platform クラスターを削除すると、このクラスターを**デタッチ**するか、**破棄**できます。

クラスターをデタッチするとマネージメントから削除されますが、完全に削除されません。管理下に戻す場合には、もう一度インポートし直すことができます。このオプションは、クラスターが **Ready** 状態にある場合にだけ利用できます。

クラスターを破棄すると、マネージメントから削除され、クラスターのコンポーネントが削除されます。こちらは、完全に削除されるので、削除後に管理下に戻すことはできません。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。
2. 削除するクラスターの横にあるオプションメニューを選択します。
3. **Destroy cluster** または **Detach cluster** を選択します。
ヒント: 複数のクラスターをデタッチまたは破棄するには、デタッチまたは破棄するクラスターのチェックボックスを選択します。次に、**Detach** または **Destroy** を選択します。

第8章 ハブクラスターへのターゲットのマネージドクラスターのインポート

別の Kubernetes クラウドプロバイダーからクラスターをインポートできます。インポートすると、ターゲットクラスターは Red Hat Advanced Cluster Management for Kubernetes ハブクラスターのマネージドクラスターになります。指定されていない場合には、ハブクラスターとターゲットのマネージドクラスターにアクセスできる場所で、インポートタスクを実行します。



注記

ハブクラスターは、他のハブクラスターを管理 **できない** ので、既存のクラスターをインポートする必要があります。

コンソールまたは CLI からのマネージドクラスターの設定は、以下の手順から選択します。

必要なユーザータイプまたはアクセスレベル: クラスター管理者

- [コンソールを使用した既存クラスターのインポート](#)
- [CLI を使用したマネージドクラスターのインポート](#)
- [クラスターの klusterlet アドオン設定の変更](#)

8.1. コンソールを使用した既存クラスターのインポート

Red Hat Advanced Cluster Management for Kubernetes をインストールすると、管理するクラスターをインポートする準備が整います。コンソールと CLI の両方からインポートできます。コンソールからインポートするには、以下の手順に従います。この手順では、認証用にターミナルが必要です。

- [前提条件](#)
- [クラスターのインポート](#)
- [クラスターの削除](#)

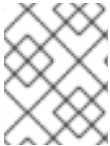
8.1.1. 前提条件

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく必要がある。ベアメタルクラスターをインポートする場合には、ハブクラスターを Red Hat OpenShift Container Platform バージョン 4.4 以降にインストールする必要があります。
- 管理するクラスターとインターネット接続が必要である。
- **kubectl** をインストールしておく必要がある。**kubectl**のインストール手順は、[Kubernetes ドキュメント](#) の [Install and Set Up kubectl](#)を参照してください。
- **Base64** コマンドラインツールが必要である。

必要なユーザータイプまたはアクセスレベル: クラスター管理者

8.1.2. クラスターのインポート

利用可能なクラウドプロバイダーごとに、Red Hat Advanced Cluster Management for Kubernetes コンソールから既存のクラスターをインポートできます。



注記

ハブクラスターは、他のハブクラスターを管理 **できない** ので、既存のクラスターをインポートする必要があります。

1. ナビゲーションメニューで **Automate infrastructure** にマウスをかざし、**Clusters** をクリックします。
2. **Add cluster** をクリックします。
3. **Import an existing cluster** をクリックします。
4. クラスター名を指定します。デフォルトで、namespace はクラスター名と同じ値に設定されます。**ベストプラクティス:** namespace の値はそのままにして、編集しないでください。
5. **オプション:** クリックして **Edit cluster import YAML file** を展開し、エンドポイント設定を変更します。
各パラメーターの詳細は、[表 1 YAML ファイルパラメーターおよび説明](#) を参照してください。
6. **オプション:** インポート後に、**Configure advanced parameters** をクリックしてラベルを追加し、検索時にこのラベルを使用します。
7. **オプション:** **MANAGED CLUSTER URLS** を設定します。**MANAGED CLUSTER URLS** を設定すると、**oc get managedcluster** コマンドを実行時に、この URL がテーブルに表示されます。
 - a. まだオンになっていない場合には、Web コンソールのスイッチを使用して **YAML** コンテンツをオンにし、コンテンツを表示できるようにします。
 - b. 以下の例のように、**import.yaml** ファイルの **ManagedCluster** 仕様に **manageClusterClientConfigs** のセクションを追加します。

```
apiVersion: cluster.open-cluster-management.io/v1
kind: ManagedCluster
metadata:
  labels:
    cloud: auto-detect
    vendor: auto-detect
    name: cluster-test
    name: cluster-test
spec:
  hubAcceptsClient: true
  managedClusterClientConfigs:
    - url: https://multicloud-console.apps.new-managed.dev.redhat.com
  ---
apiVersion: agent.open-cluster-management.io/v1
...
```

URL の値は、マネージドクラスターの外部アクセス URL アドレスに置き換えます。

8. **Generate Command** をクリックし、コマンドを取得して、**open-cluster-management-agent-addon** をデプロイします。
9. **Import an existing cluster** ウィンドウで **Copy command** アイコンにマウスをかざしてクリックし、インポートコマンドと指定のトークンをコピーします。適切なコピーを取得するには、**Copy** アイコンをクリックする必要があります。**重要:** コマンドには、インポートした各クラスターにコピーされるプルシークレット情報が含まれます。インポートしたクラスターにア

クセスできるユーザーであれば誰でも、プルシークレット情報を表示できます。<https://cloud.redhat.com/> で 2 つ目のプルシークレットを作成するか、サービスアカウントを作成して、個人の認証情報が不正アクセスされないようにします。詳細は、「[イメージプルシークレットの使用](#)」または「[サービスアカウントの概要および作成](#)」を参照してください。

10. ターミナルから、マネージドクラスターへの認証を行います。ターゲットのマネージドクラスターの **kubectl** を設定します。
kubectl の設定方法は、[サポート対象のクラウド](#) を参照してください。
11. マネージドクラスターに **open-cluster-management-agent-addon** をデプロイするには、[手順 8](#) で生成してコピーしたコマンドを実行します。
12. **View cluster** をクリックして **Overview** ページのクラスターの概要を表示します。

注記 続けて他のクラスターもインポートできます。**Import another** をクリックして、このプロセスを繰り返します。

8.1.2.1. YAML パラメーターおよび説明

表 1: YAML ファイルで利用可能なパラメーターおよび説明を以下の表に示します。

パラメーター	説明	デフォルト値
clusterLabels	クラスターラベルを指定します。ラベルをファイルに追加できます。	なし
clusterLabels.cloud	クラスターのプロバイダーラベル	auto-detect
clusterLabels.vendor	クラスターの Kubernetes ベンダーラベル	auto-detect
clusterLabels.environment	クラスターの環境ラベル	なし
clusterLabels.region	クラスターが設定されているリージョン	なし
applicationManager.enabled	マルチクラスターマネージャーのアプリケーションデプロイメントを有効化して、サブスクリプションコントローラーと deployable コントローラーをデプロイします。	true
searchCollector.enabled	検索コレクションおよびインデックス化を有効にします。	true
policyController.enabled	ガバナンスおよびリスクダッシュボードのポリシー機能を有効にします。	true、updateInterval: 15

パラメーター	説明	デフォルト値
certPolicyController.enabled	分散ポリシーに基づいて証明書の有効期限を監視します	true
iamPolicyController	分散ポリシーに基づくアイデンティティ管理を監視します	true
serviceRegistry.enabled	アプリケーションの deployable によりマネージドクラスター全体にデプロイされたサービスを検出するのに使用するサービスレジストリー。	false
serviceRegistry.dnsSuffix	レジストリーの DNS 名のサフィックス。これは、ターゲットクラスターの DNS ドメイン名の末尾に追加されます。	mcm.svc
serviceRegistry.plugins	有効化したプラグインのコンマ区切りの一覧。サポートされるプラグイン: kube-service , kube-ingress および istio	kube-service
version	open-cluster-management-agent-addon のバージョン	2.0

8.1.3. インポートされたクラスターの削除

以下の手順を実行して、インポートされたクラスターと、マネージドクラスターで作成された **open-cluster-management-agent-addon** を削除します。

1. **Clusters** ページの表から、インポートされたクラスターを見つけます。
2. **Options > Detach cluster** をクリックしてマネージメントからクラスターを削除します。

8.2. CLI を使用したマネージドクラスターのインポート

Red Hat Advanced Cluster Management for Kubernetes をインストールすると、管理するクラスターをインポートする準備が整います。コンソールと CLI の両方からインポートできます。以下の手順に従って、CLI からインポートします。

- [前提条件](#)
- [サポート対象のアーキテクチャー](#)
- [klusterlet のインポート](#)

**注記**

ハブクラスターは別のハブクラスターを管理できません。

8.2.1. 前提条件

- Red Hat Advanced Cluster Management for Kubernetes のハブクラスターをデプロイしておく必要がある。ベアメタルクラスターをインポートする場合には、ハブクラスターを Red Hat OpenShift Container Platform バージョン 4.4 以降にインストールする必要があります。**重要:** ベアメタル機能はテクノロジープレビューであるため、実稼働環境では使用しないでください。
- 管理予定の別のクラスターとインターネット接続が必要です。
- **oc** コマンドを実行するには、Red Hat OpenShift Container Platform の CLI バージョン 4.3 以降が必要である。Red Hat OpenShift CLI (**oc**) のインストールおよび設定の詳細は、「[CLI の使用方法](#)」を参照してください。
- Kubernetes CLI (**kubectl**) をインストールする必要がある。**kubectl**のインストール手順は、[Kubernetes ドキュメント](#) の **Install and Set Up kubectl**を参照してください。

**注記**

コンソールから CLI ツールのインストールファイルをダウンロードします。

8.2.2. サポート対象のアーキテクチャー

- Linux
- macOS

8.2.3. インポートの準備

1. **ハブクラスター** にログインします。以下のコマンドを実行します。

```
oc login
```

2. ハブクラスターで以下のコマンドを実行して namespace を作成します。**注記:** **<cluster_name>** で定義したクラスター名は、**.yaml** ファイルおよびコマンドでクラスターの namespace としても使用します。

```
oc new-project ${CLUSTER_NAME}
oc label namespace ${CLUSTER_NAME} cluster.open-cluster-
management.io/managedCluster=${CLUSTER_NAME}
```

3. 以下の YAML 例のように、ManagedCluster の例を編集します。

```
apiVersion: cluster.open-cluster-management.io/v1
kind: ManagedCluster
metadata:
  name: <cluster_name>
spec:
  hubAcceptsClient: true
```

4. ファイルは **managed-cluster.yaml** として保存します。
5. 以下のコマンドを使用して、YAML ファイルを適用します。

```
oc apply -f managed-cluster.yaml
```

6. klusterlet のアドオン設定ファイルを作成します。以下の YAML の例を入力します。

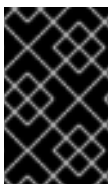
```
apiVersion: agent.open-cluster-management.io/v1
kind: KlusterletAddonConfig
metadata:
  name: <cluster_name>
  namespace: <cluster_name>
spec:
  clusterName: <cluster_name>
  clusterNamespace: <cluster_name>
  applicationManager:
    enabled: true
  certPolicyController:
    enabled: true
  clusterLabels:
    cloud: auto-detect
    vendor: auto-detect
  iamPolicyController:
    enabled: true
  policyController:
    enabled: true
  searchCollector:
    enabled: true
  version: 2.0.0
```

7. ファイルは **klusterlet-addon-config.yaml** として保存します。
8. YAML を適用します。以下のコマンドを実行します。

```
oc apply -f klusterlet-addon-config.yaml
```

ManagedCluster-Import-Controller は **\$(CLUSTER_NAME)-import** という名前のシークレットを生成します。**\$(CLUSTER_NAME)-import** シークレットには、**import.yaml** が含まれており、このファイルをユーザーがマネージドクラスターに適用して klusterlet をインストールします。

8.2.4. klusterlet のインポート



重要

import コマンドには、インポートした各クラスターにコピーされるプルシークレット情報が含まれます。インポートしたクラスターにアクセスできるユーザーであれば誰でも、プルシークレット情報を表示できます。

1. マネージドクラスターのインポートコントローラーによって生成された **klusterlet-crd.yaml** を取得します。
以下のコマンドを実行します。

```
oc get secret ${CLUSTER_NAME}-import -n ${CLUSTER_NAME} -o jsonpath=
{.data.crd$\yaml} | base64 --decode > klusterlet-crd.yaml
```

2. マネージドクラスターのインポートコントローラーによって生成された **import.yaml** を取得します。以下のコマンドを実行します。

```
oc get secret ${CLUSTER_NAME}-import -n ${CLUSTER_NAME} -o jsonpath=
{.data.import$\yaml} | base64 --decode > import.yaml
```

3. ターゲットの マネージド クラスターにログインします。
4. 手順1で生成した **klusterlet-crd.yaml** を適用します。以下のコマンドを実行します。

```
kubectl apply -f klusterlet-crd.yaml
```

5. 手順2で生成した **import.yaml** ファイルを適用します。以下のコマンドを実行します。

```
kubectl apply -f import.yaml
```

6. ターゲットのマネージドクラスターで Pod のステータスを検証します。以下のコマンドを実行します。

```
kubectl get pod -n open-cluster-management-agent
```

7. インポートしたクラスターのステータス (**JOINED** および **AVAILABLE**) を確認します。ハブ クラスターから以下のコマンドを実行します。

```
kubectl get managedcluster -n ${CLUSTER_NAME}
```

8. アドオンは、マネージドクラスターが **AVAILABLE** になってからインストールされます。ターゲットのマネージドクラスターでアドオンの Pod ステータスを確認します。以下のコマンドを実行します。

```
kubectl get pod -n open-cluster-management-agent-addon
```

8.3. クラスターの KLUSTERLET アドオン設定の変更

ハブクラスターを使用して設定を変更するには、**klusterlet addon** の設定を変更します。

klusterlet addon コントローラーは、**klusterletaddonconfigs.agent.open-cluster-management.io** Kubernetes リソースの設定に合わせて有効化/無効化される機能を管理します。

以下の設定は、**klusterletaddonconfigs.agent.open-cluster-management.io** の Kubernetes リソースで更新できます。

設定名	値
applicationmanager	true または false
policyController	true または false

設定名	値
searchCollector	true または false
certPolicyController	true または false
iamPolicyController	true または false

8.3.1. ハブクラスターのコンソールを使用した変更

ハブクラスターを使用して、**klusterletaddonconfigs.agent.open-cluster-management.io** リソースの設定を変更できます。設定の変更には、以下の手順を実行します。

1. ハブクラスターの Red Hat Advanced Cluster Management for Kubernetes コンソールへの認証を行います。
2. ハブクラスターのメインメニューから **Search** を選択します。
3. 検索パラメーターに、**kind:klusterletaddonconfigs** の値を入力します。
4. 更新するエンドポイントリソースを選択します。
5. **spec** セクションから、**Edit** を選択してコンテンツを編集します。
6. 設定を変更します。
7. **Save** を選択して変更を適用します。

8.3.2. ハブクラスターのコマンドラインを使用した変更

ハブクラスターを使用して設定を変更するには、<cluster-name> namespace へのアクセス権が必要です。以下の手順を実行します。

1. ハブクラスターへの認証を行います。
2. 以下のコマンドを入力してリソースを編集します。

```
kubectl edit klusterletaddonconfigs.agent.open-cluster-management.io <cluster-name> -n <cluster-name>
```

3. **spec** セクションを検索します。
4. 必要に応じて設定を変更します。

第9章 クラスターのアップグレード

Red Hat Advanced Cluster Management for Kubernetes で管理するクラスターを作成したら、Red Hat Advanced Cluster Management コンソールを使用して、マネージドクラスターが使用するバージョンチャンネルで利用可能な最新のマイナーバージョンに、これらのクラスターをアップグレードできます。

メジャーバージョンへのアップグレードには、そのバージョンへのアップグレードの前提条件をすべて満たしていることを確認する必要があります。コンソールでクラスターをアップグレードする前に、マネージドクラスターのバージョンチャンネルを更新する必要があります。マネージドクラスターのバージョンチャンネルの更新後に、Red Hat Advanced Cluster Management コンソールに、アップグレードに利用可能な最新版が表示されます。

注記: Red Hat OpenShift Kubernetes Service クラスターは、Red Hat Advanced Cluster Management for Kubernetes コンソールでアップグレードできません。

このアップグレードの手法は、ステータスが **Ready** の Red Hat OpenShift Container Platform クラスターに対してだけ、使用できます。

クラスターをアップグレードするには、以下の手順を実行します。

1. ナビゲーションメニューから **Automate infrastructure > Clusters** に移動します。アップグレードが利用可能な場合には、**Distribution version** の列に表示されます。
2. アップグレードするクラスターを選択します。**注記:** クラスターは **Ready** の状態であること、かつ、コンソールを使用して OpenShift Container Platform クラスターをアップグレードする必要があります。
3. **Upgrade** を選択します。
4. 各クラスターの新しいバージョンを選択します。
5. **Upgrade** を選択します。

9.1. 非接続クラスターのアップグレード

Red Hat OpenShift Update Service を Red Hat Advanced Cluster Management for Kubernetes で使用すると、非接続環境でクラスターをアップグレードできます。

重要: Red Hat OpenShift Update Service は、Red Hat OpenShift Container Platform Operator で、OpenShift Container Platform 4.4 のテクノロジープレビュー機能として提供されます。実稼働環境での使用を目的としていません。

セキュリティ上の理由で、クラスターがインターネットに直接接続できない場合があります。このような場合は、アップグレードが利用可能なタイミングや、これらのアップグレードの処理方法を把握するのが困難になります。OpenShift Update Service を設定すると便利です。

OpenShift Update Service は、個別の Operator および オペランドで、非接続環境で利用可能なマネージドクラスターを監視して、クラスターのアップグレードで利用できるようにします。OpenShift Update Service の設定後に、以下のアクションを実行できます。

1. オフラインのクラスター向けにいつアップグレードが利用できるかを監視します。
2. グラフデータファイルを使用してアップグレード用にどの更新がローカルサイトにミラーリングされているかを特定します。

- Red Hat Advanced Cluster Management コンソールを使用して、クラスターのアップグレードが利用可能であることを通知します。

9.1.1. 前提条件

OpenShift Update Service を使用して非接続クラスターをアップグレードするには、以下の前提条件を満たす必要があります。

- Red Hat OpenShift Container Platform 4.5 以降に、制限付きの OLM を設定して Red Hat Advanced Cluster Management ハブクラスターをデプロイしておく。制限付きの OLM の設定方法については、「[ネットワークが制限された環境での Operator Lifecycle Manager の使用](#)」を参照してください。

ヒント: 制限付きの OLM の設定時に、カタログソースイメージをメモします。

- Red Hat Advanced Cluster Management ハブクラスターが管理する OpenShift Container Platform クラスター。
- クラスターイメージをミラーリング可能なローカルレジストリーにアクセスするための認証情報。このリポジトリの作成方法については、「[ネットワークが制限された環境でのインストール用のミラーレジストリーの作成](#)」を参照してください。

注記: アップグレードするクラスターの現行バージョンのイメージは、ミラーリングされたイメージの1つとして常に利用可能でなければなりません。アップグレードに失敗すると、クラスターはアップグレード試行時のクラスターのバージョンに戻ります。

9.1.2. 非接続ミラーレジストリーの準備

ローカルのミラーリングレジストリーに、アップグレード前の現行のイメージと、アップグレード後のイメージの移行をミラーリングする必要があります。イメージをミラーリングするには以下の手順を実行します。

- 以下の例のような内容を含むスクリプトファイルを作成します。

```
UPSTREAM_REGISTRY=quay.io
PRODUCT_REPO=openshift-release-dev
RELEASE_NAME=ocp-release
OCP_RELEASE=4.5.2-x86_64
LOCAL_REGISTRY=$(hostname):5000
LOCAL_SECRET_JSON=/path/to/pull/secret

oc adm -a ${LOCAL_SECRET_JSON} release mirror \
--
from=${UPSTREAM_REGISTRY}/${PRODUCT_REPO}/${RELEASE_NAME}:${OCP_RELEASE} \
--to=${LOCAL_REGISTRY}/ocp4 \
--to-release-image=${LOCAL_REGISTRY}/ocp4/release:${OCP_RELEASE}
```

path-to-pull-secret は、OpenShift Container Platform のプルシークレットへのパスに置き換えます。

- スクリプトを実行して、イメージのミラーリング、設定の構成、リリースイメージとリリースコンテンツの分離を行います。

ヒント: **ImageContentSourcePolicy** の作成時に、このスクリプトの最後の行にある出力を使用できます。

9.1.3. OpenShift Update Service の Operator のデプロイ

OpenShift Container Platform 環境で OpenShift Update Service の Operator をデプロイするには、以下の手順を実行します。

1. ハブクラスターで、OpenShift Container Platform Operator のハブにアクセスします。
2. **Red Hat OpenShift Update Service Operator** を選択して Operator をデプロイします。必要に応じてデフォルト値を更新します。Operator をデプロイすると、**openshift-cincinnati** という名前の新規プロジェクトが作成されます。
3. Operator のインストールが完了するまで待ちます。
ヒント: OpenShift Container Platform コマンドラインで **oc get pods** コマンドを入力して、インストールのステータスを確認できます。Operator の状態が **running** であることを確認します。

9.1.4. グラフデータの init コンテナの構築

OpenShift Update Service はグラフデータ情報を使用して、利用可能なアップグレードを判別します。オンライン環境では、OpenShift Update Service は [Cincinnati グラフデータの GitHub リポジトリ](#) から直接利用可能なアップグレードがないか、グラフデータ情報をプルします。非接続環境を設定しているため、**init container** を使用してローカルリポジトリでグラフデータを利用できるようにする必要があります。以下の手順を実行して、グラフデータの **init container** を作成します。

1. 以下のコマンドを入力して、**グラフデータ** Git リポジトリのクローンを作成します。

```
git clone https://github.com/openshift/cincinnati-graph-data
```

2. グラフデータの **init** の情報が含まれるファイルを作成します。このサンプル [Dockerfile](#) は、**cincinnati-operator** GitHub リポジトリにあります。ファイルの内容は以下の例のようになります。

```
FROM registry.access.redhat.com/ubi8/ubi:8.1

RUN curl -L -o cincinnati-graph-data.tar.gz https://github.com/openshift/cincinnati-graph-data/archive/master.tar.gz

RUN mkdir -p /var/lib/cincinnati/graph-data/

CMD exec /bin/bash -c "tar xvzf cincinnati-graph-data.tar.gz -C /var/lib/cincinnati/graph-data/ --strip-components=1"
```

この例では、以下のように設定されています。

- **FROM** 値は、OpenShift Update Service がイメージを検索する先の外部レジストリーに置き換えます。
 - **RUN** コマンドはディレクトリーを作成し、アップグレードファイルをパッケージ化します。
 - **CMD** コマンドは、パッケージファイルをローカルリポジトリにコピーして、ファイルを展開してアップグレードします。
3. 以下のコマンドを実行して、**graph data init container** をビルドします。

```
podman build -f <path_to_Dockerfile> -t
${DISCONNECTED_REGISTRY}/cincinnati/cincinnati-graph-data-container:latest
podman push ${DISCONNECTED_REGISTRY}/cincinnati/cincinnati-graph-data-
container:latest --authfile=/path/to/pull_secret.json
```

`path_to_Dockerfile` は、直前の手順で作成したファイルへのパスに置き換えます。

`${DISCONNECTED_REGISTRY}/cincinnati/cincinnati-graph-data-container` は、ローカルグラフデータ init container へのパスに置き換えます。

`/path/to/pull_secret` は、プルシークレットへのパスに置き換えます。

注記: `podman` がインストールされていない場合は、コマンドの `podman` を `docker` に置き換えることもできます。

9.1.5. ミラーリングされたレジストリーの証明書の設定

セキュアな外部コンテナレジストリーを使用してミラーリングされた OpenShift Container Platform リリースイメージを保存する場合は、アップグレードグラフをビルドするために OpenShift Update Service からこのレジストリーへのアクセス権が必要です。OpenShift Update Service Pod と連携するように CA 証明書を設定するには、以下の手順を実行します。

1. **image.config.openshift.io** にある OpenShift Container Platform 外部レジストリー API を検索します。これは、外部レジストリーの CA 証明書の保存先です。
詳細は、OpenShift Container Platform ドキュメントの「[OpenShift Container Platform のイメージレジストリー Operator](#)」を参照してください。
2. **openshift-config** namespace に ConfigMap を作成します。
3. CA 証明書をキーの **cincinnati-registry** に追加します。OpenShift Update Service はこの設定を使用して、証明書を特定します。

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: trusted-ca
data:
  cincinnati-registry: |
    -----BEGIN CERTIFICATE-----
    ...
    -----END CERTIFICATE-----
```

4. **image.config.openshift.io** API の **cluster** リソースを編集して、**additionalTrustedCA** フィールドを作成した ConfigMap 名に設定します。

```
oc patch image.config.openshift.io cluster -p '{"spec":{"additionalTrustedCA":
{"name":"trusted-ca"}}}' --type merge
```

trusted-ca は、新しい ConfigMap へのパスに置き換えます。

OpenShift Update Service Operator は、変更がないか、**image.config.openshift.io** API と、**openshift-config** namespace に作成した ConfigMap を監視し、CA 証明書が変更された場合にはデプロイメントを再起動します。

9.1.6. OpenShift Update Service インスタンスのデプロイ

ハブクラスターへの OpenShift Update Service インスタンスのデプロイが完了したら、このインスタンスは、クラスターのアップグレードのイメージをミラーリングして非接続マネージドクラスターに提供する場所に配置されます。インスタンスをデプロイするには、以下の手順を実行します。

1. デフォルトの Operator の namespace (**openshift-cincinnati**) を使用しない場合は、お使いの OpenShift Update Service インスタンスの namespace を作成します。
 - a. OpenShift Container Platform ハブクラスターコンソールのナビゲーションメニューで、**Administration > Namespaces** を選択します。
 - b. **Create Namespace** を選択します。
 - c. namespace 名と、namespace のその他の情報を追加します。
 - d. **Create** を選択して namespace を作成します。
2. OpenShift Container Platform コンソールの **Installed Operators** セクションで、**Red Hat OpenShift Update Service Operator** を選択します。
3. メニューから **Create Instance** を選択します。
4. OpenShift Update Service インスタンスからコンテンツを貼り付けます。YAML ファイルは以下のマニフェストのようになります。

```
apiVersion: cincinnati.openshift.io/v1beta1
kind: Cincinnati
metadata:
  name: openshift-update-service-instance
  namespace: openshift-cincinnati
spec:
  registry: <registry_host_name>:<port>
  replicas: 1
  repository: ${LOCAL_REGISTRY}/ocp4/release
  graphDataImage: '<host_name>:<port>/cincinnati-graph-data-container'
```

spec.registry の値は、イメージの非接続環境にあるローカルレジストリーへのパスに置き換えます。

spec.graphDataImage の値は、グラフデータ init container へのパスに置き換えます。**ヒント:** これは、**podman push** コマンドを使用して、グラフデータ init container をプッシュする時に使用した値と同じです。

5. **Create** を選択してインスタンスを作成します。
6. ハブクラスター CLI で **oc get pods** コマンドを入力し、インスタンス作成のステータスを表示します。時間がかかる場合がありますが、コマンド結果でインスタンスと Operator が実行中である旨が表示されたらプロセスは完了です。

9.1.7. デフォルトレジストリーを上書きするためのポリシーのデプロイ (任意)

注記: 本セクションの手順は、ミラーレジストリーにリリースをミラーリングした場合にのみ該当します。

OpenShift Container Platform にはイメージレジストリーのデフォルト値があり、この値でアップグレードパッケージの検索先を指定します。非接続環境では、リリースイメージをミラーリングするローカルイメージレジストリーへのパスに値を置き換えるポリシーを作成してください。

これらの手順では、ポリシーの名前に **ImageContentSourcePolicy** を指定します。ポリシーを作成するには、以下の手順を実行します。

1. ハブクラスターの OpenShift Container Platform 環境にログインします。
2. OpenShift Container Platform ナビゲーションから **Administration > Custom Resource Definitions** を選択します。
3. **Instances** タブを選択します。
4. コンテンツが表示されるように非接続 OLM を設定する時に作成した **ImageContentSourcePolicy** の名前を選択します。
5. **YAML** タブを選択して、**YAML** 形式でコンテンツを表示します。
6. ImageContentSourcePolicy の内容全体をコピーします。
7. Red Hat Advanced Cluster Management コンソールで、**Govern risk > Create policy** を選択します。
8. **YAML** スイッチを **On** に設定して、ポリシーの YAML バージョンを表示します。
9. **YAML** コードのコンテンツをすべて削除します。
10. 以下の **YAML** コンテンツをウィンドウに貼り付け、カスタムポリシーを作成します。

```
apiVersion: policy.open-cluster-management.io/v1
kind: Policy
metadata:
  name: policy-pod
  namespace: default
  annotations:
    policy.open-cluster-management.io/standards:
    policy.open-cluster-management.io/categories:
    policy.open-cluster-management.io/controls:
spec:
  disabled: false
  policy-templates:
  - objectDefinition:
      apiVersion: policy.open-cluster-management.io/v1
      kind: ConfigurationPolicy
      metadata:
        name: policy-pod-sample-nginx-pod
      spec:
        object-templates:
        - complianceType: musthave
          objectDefinition:
            apiVersion: v1
            kind: Pod
            metadata:
              name: sample-nginx-pod
              namespace: default
            status:
```

```

        phase: Running
        remediationAction: inform
        severity: low
        remediationAction: enforce
    ---
    apiVersion: policy.open-cluster-management.io/v1
    kind: PlacementBinding
    metadata:
      name: binding-policy-pod
      namespace: default
    placementRef:
      name: placement-policy-pod
      kind: PlacementRule
      apiGroup: apps.open-cluster-management.io
    subjects:
    - name: policy-pod
      kind: Policy
      apiGroup: policy.open-cluster-management.io
    ---
    apiVersion: apps.open-cluster-management.io/v1
    kind: PlacementRule
    metadata:
      name: placement-policy-pod
      namespace: default
    spec:
      clusterConditions:
      - status: "True"
        type: ManagedClusterConditionAvailable
      clusterSelector:
        matchExpressions:
        [] # selects all clusters if not specified

```

11. テンプレートの **objectDefinition** セクション内のコンテンツは、ImageContentSourcePolicy の設定を追加する以下の内容に置き換えます。

```

apiVersion: operator.openshift.io/v1alpha1
kind: ImageContentSourcePolicy
metadata:
  name: ImageContentSourcePolicy
spec:
  repositoryDigestMirrors:
  - mirrors:
    - <path-to-local-mirror>
    source: registry.redhat.io

```

- **path-to-local-mirror** は、ローカルミラーリポジトリへのパスに置き換えます。
- ヒント: **oc adm release mirror** コマンドを入力すると、ローカルミラーへのパスが分かります。

12. **Enforce if supported** のボックスを選択します。

13. **Create** を選択してポリシーを作成します。

9.1.8. 非接続カタログソースをデプロイするためのポリシーのデプロイ

マネージドクラスターに **Catalogsource** ポリシーをプッシュして、接続環境がある場所から非接続のローカルレジストリーにデフォルトの場所を変更します。

1. Red Hat Advanced Cluster Management コンソールで **Automate infrastructure > Clusters** を選択します。
2. クラスター一覧でポリシーを受信するマネージドクラスターを検索します。
3. マネージドクラスターの **name** ラベルの値をメモします。ラベルの形式は **name=managed-cluster-name** です。この値は、ポリシーのプッシュ時に使用します。
4. Red Hat Advanced Cluster Management コンソールメニューで、**Govern risk > Create policy** を選択します。
5. **YAML** スイッチを **On** に設定して、ポリシーの YAML バージョンを表示します。
6. **YAML** コードのコンテンツをすべて削除します。
7. 以下の **YAML** コンテンツをウィンドウに貼り付け、カスタムポリシーを作成します。
8. 以下の **YAML** コンテンツをウィンドウに貼り付け、カスタムポリシーを作成します。

```
apiVersion: policy.open-cluster-management.io/v1
kind: Policy
metadata:
  name: policy-pod
  namespace: default
  annotations:
    policy.open-cluster-management.io/standards:
    policy.open-cluster-management.io/categories:
    policy.open-cluster-management.io/controls:
spec:
  disabled: false
  policy-templates:
  - objectDefinition:
      apiVersion: policy.open-cluster-management.io/v1
      kind: ConfigurationPolicy
      metadata:
        name: policy-pod-sample-nginx-pod
      spec:
        object-templates:
        - complianceType: musthave
          objectDefinition:
            apiVersion: v1
            kind: Pod
            metadata:
              name: sample-nginx-pod
              namespace: default
            status:
              phase: Running
            remediationAction: inform
            severity: low
          remediationAction: enforce
    ---
    apiVersion: policy.open-cluster-management.io/v1
    kind: PlacementBinding
    metadata:
```

```

    name: binding-policy-pod
    namespace: default
  placementRef:
    name: placement-policy-pod
    kind: PlacementRule
    apiGroup: apps.open-cluster-management.io
  subjects:
  - name: policy-pod
    kind: Policy
    apiGroup: policy.open-cluster-management.io
---
apiVersion: apps.open-cluster-management.io/v1
kind: PlacementRule
metadata:
  name: placement-policy-pod
  namespace: default
spec:
  clusterConditions:
  - status: "True"
    type: ManagedClusterConditionAvailable
  clusterSelector:
    matchExpressions:
    [] # selects all clusters if not specified

```

9. 以下の内容をポリシーに追加します。

```

apiVersion: config.openshift.io/v1
kind: OperatorHub
metadata:
  name: cluster
spec:
  disableAllDefaultSources: true

```

10. 以下の内容を追加します。

```

apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: my-operator-catalog
  namespace: openshift-marketplace
spec:
  sourceType: grpc
  image: <registry_host_name>:<port>/olm/redhat-operators:v1
  displayName: My Operator Catalog
  publisher: grpc

```

`spec.image` の値は、ローカルの制約付きのカatalogソースイメージへのパスに置き換えます。

11. Red Hat Advanced Cluster Management コンソールのナビゲーションで、**Automate infrastructure > Clusters** を選択して、マネージドクラスターのステータスを確認します。ポリシーが適用されると、クラスターのステータスは **Ready** になります。

9.1.9. マネージドクラスターのパラメーターを変更するためのポリシーのデプロイ

ClusterVersion ポリシーをマネージドクラスターにプッシュし、アップグレード取得先のデフォルトの場所を変更します。

1. マネージドクラスターから、以下のコマンドを入力して **ClusterVersion** アップストリームパラメーターがデフォルトの OpenShift Update Service オペランドであることを確認します。

```
oc get clusterversion -o yaml
```

返される内容は以下のようになります。

```
apiVersion: v1
items:
- apiVersion: config.openshift.io/v1
  kind: ClusterVersion
  [...]
  spec:
    channel: stable-4.4
    upstream: https://api.openshift.com/api/upgrades_info/v1/graph
```

2. ハブクラスターから、**oc get routes** というコマンドを入力して OpenShift Update Service オペランドへのルート URL を特定します。
ヒント: 今後の手順でできるようにこの値をメモします。
3. ハブクラスターの Red Hat Advanced Cluster Management コンソールメニューで、**Govern risk > Create a policy** を選択します。
4. **YAML** スイッチを **On** に設定して、ポリシーの YAML バージョンを表示します。
5. **YAML** コードのコンテンツをすべて削除します。
6. 以下の **YAML** コンテンツをウィンドウに貼り付け、カスタムポリシーを作成します。

```
apiVersion: policy.open-cluster-management.io/v1
kind: Policy
metadata:
  name: policy-pod
  namespace: default
  annotations:
    policy.open-cluster-management.io/standards:
    policy.open-cluster-management.io/categories:
    policy.open-cluster-management.io/controls:
spec:
  disabled: false
  policy-templates:
  - objectDefinition:
      apiVersion: policy.open-cluster-management.io/v1
      kind: ConfigurationPolicy
      metadata:
        name: policy-pod-sample-nginx-pod
      spec:
        object-templates:
        - complianceType: musthave
          objectDefinition:
            apiVersion: v1
            kind: Pod
            metadata:
```

```

        name: sample-nginx-pod
        namespace: default
      status:
        phase: Running
      remediationAction: inform
      severity: low
    remediationAction: enforce
  ---
  apiVersion: policy.open-cluster-management.io/v1
  kind: PlacementBinding
  metadata:
    name: binding-policy-pod
    namespace: default
  placementRef:
    name: placement-policy-pod
    kind: PlacementRule
    apiGroup: apps.open-cluster-management.io
  subjects:
  - name: policy-pod
    kind: Policy
    apiGroup: policy.open-cluster-management.io
  ---
  apiVersion: apps.open-cluster-management.io/v1
  kind: PlacementRule
  metadata:
    name: placement-policy-pod
    namespace: default
  spec:
    clusterConditions:
    - status: "True"
      type: ManagedClusterConditionAvailable
    clusterSelector:
      matchExpressions:
      [] # selects all clusters if not specified

```

7. **policy** セクションの **policy.spec** に以下の内容を追加します。

```

  apiVersion: config.openshift.io/v1
  kind: ClusterVersion
  metadata:
    name: version
  spec:
    channel: stable-4.4
    upstream: https://example-cincinnati-policy-engine-uri/api/upgrades_info/v1/graph

```

spec.upstream の値は、ハブクラスター OpenShift Update Service オペランドへのパスに置き換えます。

ヒント: 以下の手順を実行すると、オペランドへのパスを確認できます。

- a. ハブクラスターで **oc get routes -A** コマンドを実行します。
- b. **cincinnati** へのルートを見つけます。+ オペランドへのパスは、**HOST/PORT** フィールドの値です。

- マネージドクラスター CLI で、**ClusterVersion** のアップストリームパラメーターがローカルハブクラスター OpenShift Update Service URL に更新されていることを確認します。これには以下のコマンドを入力します。

```
oc get clusterversion -o yaml
```

結果は、以下の内容のようになります。

```
apiVersion: v1
items:
- apiVersion: config.openshift.io/v1
  kind: ClusterVersion
  [..]
spec:
  channel: stable-4.4
  upstream: https://<hub-cincinnati-uri>/api/upgrades_info/v1/graph
```

9.1.10. 利用可能なアップグレードの表示

以下の手順を実行して、マネージドクラスターで利用可能なアップグレード一覧を確認します。

- Red Hat Advanced Cluster Management コンソールにログインします。
- ナビゲーションメニューで **Automate infrastructure > Clusters** を選択します。
- 状態が **Ready** のクラスターを選択します。
- Actions** メニューから **Upgrade cluster** を選択します。
- オプションのアップグレードパスが利用可能であることを確認します。
注記: 現行バージョンがローカルのイメージリポジトリにミラーリングされていない場合には、利用可能なアップグレードバージョンは表示されません。

9.1.11. クラスターのアップグレード

非接続レジストリーの設定後に、Red Hat Advanced Cluster Management および OpenShift Update Service は非接続レジストリーを使用して、アップグレードが利用可能かどうかを判断します。利用可能なアップグレードが表示されない場合は、クラスターの現行のリリースイメージと、1つ後のイメージがローカルリポジトリにミラーリングされていることを確認します。クラスターの現行バージョンのリリースイメージが利用できないと、アップグレードは利用できません。

以下の手順を実行してアップグレードします。

- Red Hat Advanced Cluster Management コンソールで **Automate infrastructure > Clusters** を選択します。
- そのクラスターの内、利用可能なアップグレードがあるかどうかを判断するクラスターを特定します。
- 利用可能なアップグレードがある場合は、クラスターの **Distribution version** コラムで、アップグレードが利用可能であることが表示されます。
- クラスターの **Options** メニュー、**Upgrade cluster** の順に選択します。
- アップグレードのターゲットバージョン、**Upgrade** の順に選択します。

マネージドクラスターは、選択したバージョンに更新されます。