



Red Hat OpenShift Service on AWS 4

ROSA with HCP クラスターのインストール

Red Hat OpenShift Service on AWS (ROSA) クラスターのインストール、アクセス、
および削除

Red Hat OpenShift Service on AWS 4 ROSA with HCP クラスターのインストール

Red Hat OpenShift Service on AWS (ROSA) クラスターのインストール、アクセス、および削除

法律上の通知

Copyright © 2024 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.

概要

このドキュメントでは、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane クラスタをインストールする方法を説明します。

目次

第1章 デフォルトのオプションを使用した ROSA WITH HCP クラスターの作成	3
自動作成モードに関する考慮事項	3
1.1. デフォルトのクラスター仕様の概要	4
1.2. ROSA WITH HCP の前提条件	5
1.3. CLI を使用した ROSA WITH HCP クラスターの作成	14
1.4. 次のステップ	16
1.5. 関連情報	16
第2章 カスタム AWS KMS 暗号鍵を使用した ROSA WITH HCP クラスターの作成	17
2.1. ROSA WITH HCP の前提条件	17
2.2. 次のステップ	27
2.3. 関連情報	27
第3章 ROSA WITH HCP でのプライベートクラスターの作成	29
3.1. AWS プライベートクラスターの作成	29
3.2. API にアクセスするための AWS セキュリティーグループの設定	30
3.3. 次のステップ	31
3.4. 関連情報	31
第4章 ROSA WITH HCP クラスターでの NODE TUNING OPERATOR の使用	32
目的	32
4.1. カスタムチューニング仕様	33
4.2. ROSA WITH HCP のノードチューニング設定の作成	37
4.3. ROSA WITH HCP のノードチューニング設定の変更	39
4.4. ROSA WITH HCP のノードチューニング設定の削除	41

第1章 デフォルトのオプションを使用した ROSA WITH HCP クラスターの作成



注記

ROSA Classic のクイックスタートガイドをお探しの場合は、[Red Hat OpenShift Service on AWS クイックスタートガイド](#) を参照してください。

Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane は、Red Hat OpenShift Service on AWS (ROSA) クラスターを作成するためのより効率的で信頼性の高いアーキテクチャーを提供します。ROSA with HCP では、各クラスターに ROSA サービスアカウントで分離された専用のコントロールプレーンがあります。

デフォルトのオプションと AWS Identity and Access Management (IAM) リソースの自動作成を使用して、ROSA with HCP クラスターをすばやく作成します。ROSA CLI (**rosa**) を使用してクラスターをデプロイできます。



重要

既存の ROSA クラスターを Hosted Control Plane アーキテクチャーにアップグレードまたは変換することはできないため、ROSA with HCP の機能を使用するには新しいクラスターを作成する必要があります。



注記

ROSA with HCP クラスターは、AWS Security Token Service (STS) 認証のみをサポートします。

関連資料

- ROSA with HCP と ROSA Classic の比較は、[アーキテクチャーモデルの比較](#) のドキュメントを参照してください。
- [ROSA CLI を使用して自動モードで ROSA with HCP の使用を開始する方法](#) については、AWS ドキュメントを参照してください。

関連情報

サポートされている証明書の完全なリストは、「Red Hat OpenShift Service on AWS のプロセスとセキュリティについて」の [コンプライアンス](#) セクションを参照してください。

自動作成モードに関する考慮事項

このドキュメントの手順では、ROSA CLI の **auto** モードを使用して、現在の AWS アカウントを使用して必要な IAM リソースを即座に作成します。必要なリソースには、アカウント全体の IAM ロールおよびポリシー、クラスター固有の Operator ロール、ならびに OpenID Connect (OIDC) ID プロバイダーが含まれます。

または、IAM リソースを自動的にデプロイする代わりに、IAM リソースの作成に必要な **aws** コマンドを出力する **manual** モードを使用することもできます。**manual** モードを使用するか、カスタマイズを使用して ROSA with HCP クラスターをデプロイする手順は、[カスタマイズを使用したクラスターの作成](#) を参照してください。

次のステップ


- [AWS の前提条件](#) を満たしていることを確認してください。

1.1. デフォルトのクラスター仕様の概要

デフォルトのインストールオプションを使用すると、AWS Security Token Service (STS) を使用して ROSA with HCP クラスターをすばやく作成できます。次の要約では、デフォルトのクラスター仕様について説明します。

表1.1 ROSA with HCP クラスターのデフォルトの仕様

コンポーネント	デフォルトの仕様
アカウントおよびロール	<ul style="list-style-type: none"> • デフォルトの IAM ロールの接頭辞: ManagedOpenShift • クラスター管理者ロールは作成されない
クラスター設定	<ul style="list-style-type: none"> • デフォルトのクラスターバージョン: 最新 • ROSA CLI (rosa) を使用したインストールのデフォルトの AWS リージョン: aws CLI 設定によって定義されます。 • デフォルトの EC2 IMDS エンドポイント (v1 と v2 の両方) が有効になっています • 可用性: データプレーンの単一ゾーン • ユーザー定義プロジェクトの監視: 有効
暗号化	<ul style="list-style-type: none"> • クラウドストレージは保存時に暗号化されます。 • 追加の etcd 暗号化が有効になっていません。 • デフォルトの AWS Key Management Service (KMS) キーは、永続データの暗号化キーとして使用されます。
コンピューターードマシンプール	<ul style="list-style-type: none"> • コンピューターードインスタンスタイプ: m5.xlarge (4 vCPU 16, GiB RAM) • コンピューターード数: 2 • 自動スケーリング: 無効 • 追加のノードラベルなし
ネットワーク設定	<ul style="list-style-type: none"> • クラスターのプライバシー: パブリック • 独自の Virtual Private Cloud (VPC) を設定しておく必要があります。 • クラスター全体のプロキシは設定されていません。

コンポーネント	デフォルトの仕様
Classless Inter-Domain Routing (CIDR) の範囲	<ul style="list-style-type: none"> ● Machine CIDR: 10.0.0.0/16 ● Service CIDR: 172.30.0.0/16 ● Pod CIDR: 10.128.0.0/16 ● Host prefix: /23 <div style="display: flex; align-items: flex-start; margin-top: 10px;">  <div> <p>注記</p> <p>ROSA with HCP を使用する場合、静的 IP アドレス 172.20.0.1 は内部 Kubernetes API アドレス用に予約されています。マシン、Pod、およびサービスの CIDR 範囲は、この IP アドレスと競合してはなりません。</p> </div> </div>
クラスターのロールおよびポリシー	<ul style="list-style-type: none"> ● Operator ロールおよび OpenID Connect (OIDC) プロバイダーの作成に使用されるモード: auto <div style="display: flex; align-items: flex-start; margin-top: 10px;">  <div> <p>注記</p> <p>Hybrid Cloud Console で OpenShift Cluster Manager を使用するインストールの場合、auto モードには管理者権限が割り当てられた OpenShift Cluster Manager ロールが必要です。</p> </div> </div> <ul style="list-style-type: none"> ● デフォルトの Operator ロールの接頭辞: <cluster_name>-<4_digit_random_string>
クラスター更新戦略	<ul style="list-style-type: none"> ● 個別の更新 ● ノードドレインの1時間の猶予期間

1.2. ROSA WITH HCP の前提条件

ROSA with HCP クラスターを作成するには、次のものがが必要です。

- 設定された仮想プライベートクラウド (VPC)
- アカウント全体のロール
- OIDC 設定
- オペレーターのロール

1.2.1. ROSA with HCP クラスター用の仮想プライベートクラウドの作成

ROSA with HCP クラスターを作成するには、Virtual Private Cloud (VPC) が必要です。次の方法を使用して VPC を作成できます。

- Terraform テンプレートを使用して VPC を作成する
- AWS コンソールで VPC リソースを手動で作成する



注記

Terraform の手順はテストとデモンストレーションを目的としています。独自のインストールでは、独自に使用するために VPC にいくつかの変更を加える必要があります。また、この Terraform スクリプトを使用するときは、クラスターをインストールする予定のリージョンと同じリージョンにあることを確認する必要があります。これらの例では、**us-east-2** を使用します。

Terraform を使用した Virtual Private Cloud の作成

Terraform は、確立されたテンプレートを使用してさまざまなリソースを作成できるツールです。次のプロセスでは、必要に応じてデフォルトのオプションを使用して、ROSA with HCP クラスターを作成します。Terraform の使用の詳細は、関連情報を参照してください。

前提条件

- マシンに Terraform バージョン 1.4.0 以降がインストールされている。
- マシンに Git がインストールされている。

手順

1. シェルプロンプトを開き、次のコマンドを実行して Terraform VPC リポジトリのクローンを作成します。

```
$ git clone https://github.com/openshift-cs/terraform-vpc-example
```

2. 次のコマンドを実行して、作成したディレクトリーに移動します。

```
$ cd terraform-vpc-example
```

3. 次のコマンドを実行して、Terraform ファイルを開始します。

```
$ terraform init
```

このプロセスが完了すると、初期化を確認するメッセージが表示されます。

4. 既存の Terraform テンプレートに基づいて VPC Terraform プランを構築するには、**plan** コマンドを実行します。AWS リージョンを含める必要があります。クラスター名の指定を選択できます。**terraform plan** が完了すると、**rosa.tfplan** ファイルが **hypershift-tf** ディレクトリーに追加されます。オプションの詳細は、[Terraform VPC リポジトリの README ファイル](#) を参照してください。

```
$ terraform plan -out rosa.tfplan -var region=<region> [-var cluster_name=<cluster_name>]
```

5. 次のコマンドを実行して、このプランファイルを適用して VPC を構築します。

```
$ terraform apply rosa.tfplan
```

6. 任意: 次のコマンドを実行して、Terraform でプロビジョニングされたプライベート、パブリック、およびマシンプールのサブネット ID の値を環境変数としてキャプチャーし、ROSA with HCP クラスターを作成するときに使用できます。

```
$ export SUBNET_IDS=$(terraform output -raw cluster-subnets-string)
```

検証

- 次のコマンドを使用して、変数が正しく設定されたことを確認できます。

```
$ echo $SUBNET_IDS
```

出力例

```
$ subnet-0a6a57e0f784171aa,subnet-078e84e5b10ecf5b0
```

関連情報

- ニーズに合わせて VPC をカスタマイズするときに使用できるすべてのオプションの詳細なリストは、[Terraform VPC](#) リポジトリを参照してください。

Virtual Private Cloud を手動で作成する

Terraform を使用する代わりに Virtual Private Cloud (VPC) を手動で作成することを選択した場合は、[AWS コンソールの VPC ページ](#) に移動します。VPC は、次の表に示す要件を満たしている必要があります。

表1.2 VPC の要件

要件	詳細
VPC 名	クラスターを作成するときは、特定の VPC 名と ID が必要です。
CIDR 範囲	VPC CIDR 範囲はマシンの CIDR と一致する必要があります。
アベイラビリティゾーン	単一ゾーンの場合は1つの可用性ゾーンが必要で、複数ゾーンの場合は3つの可用性ゾーンが必要です。
パブリックサブネット	パブリッククラスターには、NAT ゲートウェイを備えたパブリックサブネットが1つ必要です。プライベートクラスターにはパブリックサブネットは必要ありません。
DNS ホスト名と解決	DNS ホスト名と解決が有効になっていることを確認する必要があります。

サブネットへのタグ付け

VPC を使用して ROSA with HCP クラスターを作成する前に、VPC サブネットにタグを付ける必要があります。自動サービスのプリフライトチェックでは、これらのリソースを使用する前に、これらのリソースが正しくタグ付けされていることを確認します。次の表は、リソースを次のようにタグ付ける

方法を示しています。

リソース	キー	値
パブリックサブネット	kubernetes.io/role/elb	1 または値なし
プライベートサブネット	kubernetes.io/role/internal-elb	1 または値なし



注記

少なくとも1つのプライベートサブネットと、該当する場合は1つのパブリックサブネットにタグを付ける必要があります。

前提条件

- VPC を作成している。
- **aws** CLI をインストールしている。

手順

1. 次のコマンドを実行して、サブネットに現在あるタグを確認します。

```
$ aws ec2 describe-tags --filters "Name=resource-id,Values=<subnet-id>"
```

出力例

```
TAGS Name <subnet-id> subnet <prefix>-subnet-public1-us-east-1a
```

2. 次のコマンドを実行して、ターミナルでリソースにタグを付けます。
 - a. パブリックサブネットの場合は、以下を実行します。

```
$ aws ec2 create-tags --resources <public-subnet-id> --tags
Key=kubernetes.io/role/elb,Value=1
```

- b. プライベートサブネットの場合は、以下を実行します。

```
$ aws ec2 create-tags --resources <private-subnet-id> --tags
Key=kubernetes.io/role/internal-elb,Value=1
```

検証

1. 次のコマンドを実行して、タグが正しく適用されていることを確認します。

```
$ aws ec2 describe-tags --filters "Name=resource-id,Values=<subnet_id>"
```

出力例

```

TAGS  Name                <subnet-id>    subnet <prefix>-subnet-public1-us-east-1a
TAGS  kubernetes.io/role/elb <subnet-id>    subnet 1

```

関連情報

- [Get Started with Amazon VPC](#)
- [HashiCorp Terraform ドキュメント](#)
- [サブネットの自動検出](#)

1.2.2. アカウント全体の STS ロールおよびポリシーの作成

Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) を使用して、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane クラスターを作成する前に、Operator ポリシーを含む、必要なアカウント全体のロールとポリシーを作成します。



注記

ROSA with HCP クラスターには、AWS 管理ポリシーがアタッチされたアカウントと Operator ロールが必要です。顧客管理のポリシーはサポートされていません。ROSA with HCP クラスターの AWS 管理ポリシーの詳細は、[AWS managed policies for ROSA account roles](#) を参照してください。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- 利用可能な AWS サービスクォータがある。
- AWS コンソールで ROSA サービスを有効にしている。
- インストールホストに、最新の ROSA CLI (**rosa**) をインストールして設定している。
- ROSA CLI を使用して Red Hat アカウントにログインしている。

手順

- AWS アカウントに存在しない場合は、次のコマンドを実行して、必要なアカウント全体の STS ロールを作成し、ポリシーをアタッチします。

```
$ rosa create account-roles --hosted-cp --mode auto --yes
```

- オプション: 次のコマンドを実行して、接頭辞を環境変数として設定します。

```
$ export ACCOUNT_ROLES_PREFIX=<account_role_prefix>
```

- 次のコマンドを実行して、変数の値を表示します。

```
$ echo $ACCOUNT_ROLES_PREFIX
```

出力例

ManagedOpenShift

ROSA の AWS 管理 IAM ポリシーの詳細は、[AWS managed IAM policies for ROSA](#) を参照してください。

1.2.3. OpenID Connect 設定の作成

ROSA with HCP クラスターを使用する場合は、クラスターを作成する前に OpenID Connect (OIDC) 設定を作成する必要があります。この設定は、OpenShift Cluster Manager で使用するために登録されています。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- Red Hat OpenShift Service on AWS の AWS 前提条件を完了している。
- インストールホストに、最新の Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) をインストールして設定している。

手順

- AWS リソースと一緒に OIDC 設定を作成するには、次のコマンドを実行します。

```
$ rosa create oidc-config --mode=auto --yes
```

このコマンドは次の情報を返します。

出力例

```
? Would you like to create a Managed (Red Hat hosted) OIDC Configuration Yes
!: Setting up managed OIDC configuration
!: To create Operator Roles for this OIDC Configuration, run the following command and
remember to replace <user-defined> with a prefix of your choice:
  rosa create operator-roles --prefix <user-defined> --oidc-config-id 13cdr6b
!: If you are going to create a Hosted Control Plane cluster please include '--hosted-cp'
!: Creating OIDC provider using 'arn:aws:iam::4540112244:user/userName'
? Create the OIDC provider? Yes
!: Created OIDC provider with ARN 'arn:aws:iam::4540112244:oidc-
provider/dvbwgdztaeq9o.cloudfront.net/13cdr6b'
```

クラスターを作成するときは、OIDC 設定 ID を指定する必要があります。CLI 出力では、**--mode auto** のこの値が提供されます。それ以外の場合は、**--mode manual** の **aws** CLI 出力に基づいてこれらの値を決定する必要があります。

- オプション: OIDC 設定 ID を変数として保存して、後で使用できます。次のコマンドを実行して変数を保存します。

```
$ export OIDC_ID=<oidc_config_id> 1
```

1 上記の出力例では、OIDC 設定 ID は 13cdr6b です。

- 次のコマンドを実行して、変数の値を表示します。

```
$ echo $OIDC_ID
```

出力例

```
13cdr6b
```

検証

- ユーザー組織に関連付けられているクラスターで使用できる可能な OIDC 設定をリストできません。以下のコマンドを実行します。

```
$ rosa list oidc-config
```

出力例

```
ID                MANAGED ISSUER URL
SECRET ARN
2330dbs0n8m3chkk25gkkcd8pnj3lk2 true
https://dvbwgdztaeq9o.cloudfront.net/2330dbs0n8m3chkk25gkkcd8pnj3lk2
233hvnjrjoqu14jltk6lhbhf2tj11f8un false https://oidc-r7u1.s3.us-east-1.amazonaws.com
aws:secretsmanager:us-east-1:242819244:secret:rosa-private-key-oidc-r7u1-tM3MDN
```

1.2.4. Operator のロールとポリシーの作成

ROSA with HCP クラスターを使用する場合は、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane (HCP) デプロイメントに必要な Operator IAM ロールを作成する必要があります。クラスター Operator は、Operator のロールを使用して、バックエンドストレージ、クラウドプロバイダーの認証情報、クラスターへの外部アクセスの管理など、クラスター操作を実行するために必要な一時的なアクセス許可を取得します。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- インストールホストに、最新の Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) をインストールして設定している。
- アカウント全体の AWS ロールを作成している。

手順

1. 次のコマンドを使用して、接頭辞名を環境変数に設定します。

```
$ export OPERATOR_ROLES_PREFIX=<prefix_name>
```

2. Operator ロールを作成するには、次のコマンドを実行します。

```
$ rosa create operator-roles --hosted-cp --prefix=$OPERATOR_ROLES_PREFIX --oidc-config-id=$OIDC_ID --installer-role-arn arn:aws:iam::${AWS_ACCOUNT_ID}:role/${ACCOUNT_ROLES_PREFIX}-HCP-ROSA-Installer-Role
```

次の内訳は、Operator ロール作成のオプションを示しています。

```
$ rosa create operator-roles --hosted-cp
--prefix=$OPERATOR_ROLES_PREFIX ❶
--oidc-config-id=$OIDC_ID ❷
--installer-role-arn
arn:aws:iam::${AWS_ACCOUNT_ID}:role/${ACCOUNT_ROLES_PREFIX}-HCP-ROSA-
Installer-Role ❸
```

- ❶ これらの Operator ロールを作成するときは、接頭辞を指定する必要があります。そうしないとエラーが発生します。演算子接頭辞は、このセクションの関連情報を参照してください。
- ❷ この値は、ROSA with HCP クラスター用に作成した OIDC 設定 ID です。
- ❸ この値は、ROSA アカウントロールの作成時に作成したインストーラーロールの ARN です。

ROSA with HCP クラスター用の正しいロールを作成するには、**--hosted-cp** パラメーターを含める必要があります。このコマンドは次の情報を返します。

出力例

```
? Role creation mode: auto
? Operator roles prefix: <pre-filled_prefix> ❶
? OIDC Configuration ID: 23soa2bgvpek9kmes9s7os0a39i13qm4 |
https://dvbwgdztaeq9o.cloudfront.net/23soa2bgvpek9kmes9s7os0a39i13qm4 ❷
? Create hosted control plane operator roles: Yes
W: More than one Installer role found
? Installer role ARN: arn:aws:iam::4540112244:role/<prefix>-HCP-ROSA-Installer-Role
? Permissions boundary ARN (optional):
I: Reusable OIDC Configuration detected. Validating trusted relationships to operator roles:
I: Creating roles using 'arn:aws:iam::4540112244:user/<userName>'
I: Created role '<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials'
I: Created role '<prefix>-openshift-cloud-network-config-controller-cloud-credenti' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-cloud-network-config-controller-cloud-
credenti'
I: Created role '<prefix>-kube-system-kube-controller-manager' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-kube-controller-manager'
I: Created role '<prefix>-kube-system-capac-controller-manager' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-capac-controller-manager'
I: Created role '<prefix>-kube-system-control-plane-operator' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-control-plane-operator'
I: Created role '<prefix>-kube-system-kms-provider' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-kms-provider'
I: Created role '<prefix>-openshift-image-registry-installer-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-image-registry-installer-cloud-credentials'
I: Created role '<prefix>-openshift-ingress-operator-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-ingress-operator-cloud-credentials'
I: To create a cluster with these roles, run the following command:
  rosa create cluster --sts --oidc-config-id 23soa2bgvpek9kmes9s7os0a39i13qm4 --operator-
  roles-prefix <prefix> --hosted-cp
```


- 1 このフィールドには、最初の作成コマンドで設定した接頭辞が事前に入力されます。
- 2 このフィールドでは、ROSA with HCP クラスター用に作成した OIDC 設定を選択する必要があります。

これで、Operator ロールが作成され、ROSA with HCP クラスターの作成に使用できるようになりました。

検証

- ROSA アカウントに関連付けられている Operator ロールをリスト表示できます。以下のコマンドを実行します。

```
$ rosa list operator-roles
```

出力例

```
I: Fetching operator roles
ROLE PREFIX AMOUNT IN BUNDLE
<prefix> 8
? Would you like to detail a specific prefix Yes 1
? Operator Role Prefix: <prefix>
ROLE NAME ROLE ARN
VERSION MANAGED
<prefix>-kube-system-cap-a-controller-manager
arn:aws:iam::4540112244:role/<prefix>-kube-system-cap-a-controller-manager
4.13 No
<prefix>-kube-system-control-plane-operator
arn:aws:iam::4540112244:role/<prefix>-kube-system-control-plane-operator
4.13 No
<prefix>-kube-system-kms-provider
arn:aws:iam::4540112244:role/<prefix>-kube-system-kms-provider 4.13
No
<prefix>-kube-system-kube-controller-manager
arn:aws:iam::4540112244:role/<prefix>-kube-system-kube-controller-manager
4.13 No
<prefix>-openshift-cloud-network-config-controller-cloud-credenti
arn:aws:iam::4540112244:role/<prefix>-openshift-cloud-network-config-controller-cloud-credenti 4.13 No
<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials
4.13 No
<prefix>-openshift-image-registry-installer-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-image-registry-installer-cloud-credentials
4.13 No
<prefix>-openshift-ingress-operator-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-ingress-operator-cloud-credentials
4.13 No
```

- 1 コマンドを実行すると、AWS アカウントに関連付けられているすべての接頭辞が表示され、この接頭辞に関連付けられているロールの数が記録されます。これらのロールとその詳細をすべて表示する必要がある場合は、詳細プロンプトで "Yes" と入力すると、これらのロールが詳細とともにリストされます。

関連情報

- オペレーター接頭辞については、[カスタム Operator IAM ロール接頭辞](#) を参照してください。

1.3. CLI を使用した ROSA WITH HCP クラスターの作成

Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) を使用してクラスターを作成する場合は、デフォルトのオプションを選択してクラスターを迅速に作成できます。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- 利用可能な AWS サービスクォータがある。
- AWS コンソールで ROSA サービスを有効にしている。
- インストールホストに、最新の ROSA CLI (**rosa**) をインストールして設定している。**rosa version** を実行して、現在インストールされている ROSA CLI のバージョンを確認します。新しいバージョンが利用可能な場合、CLI はこのアップグレードをダウンロードするためのリンクを提供します。
- ROSA CLI を使用して Red Hat アカウントにログインしている。
- OIDC 設定が作成されている。
- AWS Elastic Load Balancing (ELB) サービスロールが AWS アカウントに存在することを確認している。

手順

1. ROSA with HCP クラスターを作成するには、次のいずれかのコマンドを使用します。



注記

ROSA with HCP クラスターを作成する場合、デフォルトのマシン Classless Inter-Domain Routing (CIDR) は **10.0.0.0/16** です。これが VPC サブネットの CIDR 範囲に対応していない場合は、次のコマンドに **--machine-cidr <address_block>** を追加します。Red Hat OpenShift Service on AWS のデフォルト CIDR 範囲について、詳細は [CIDR 範囲の定義](#) を参照してください。

- 環境変数を設定していない場合は、以下のコマンドを実行します。

```
$ rosa create cluster --cluster-name=<cluster_name> \ <.>
--mode=auto --hosted-cp [--private] \ <.>
--operator-roles-prefix <operator-role-prefix> \ <.>
--oidc-config-id <id-of-oidc-configuration> \
--subnet-ids=<public-subnet-id>,<private-subnet-id>
```

<.> クラスターの名前を指定します。クラスター名が 15 文字を超える場合、openshiftapps.com でプロビジョニングされたクラスターのサブドメインとして自動生成されたドメイン接頭辞が含まれます。サブドメインをカスタマイズするには、**--domain-prefix** フラグを使用します。ドメイン接頭辞は 15 文字を超えてはならず、一意である必要があります。クラスターの作成後に変更できません。<.> オプション: **--private** 引数は、プライベート ROSA with HCP クラスターを作成するために使用されます。この引数を使用する場

合は、**--subnet-ids** にプライベートサブネット ID のみを使用するようにしてください。<> デフォルトでは、クラスター固有の Operator のロール名には、クラスター名とランダムな 4 桁のハッシュが接頭辞として付けられます。オプションで、ロール名の **<cluster_name>-<hash>** を置き換えるカスタム接頭辞を指定できます。接頭辞は、クラスター固有の Operator IAM ロールを作成するときに適用されます。接頭辞の詳細は、**カスタム Operator IAM ロール接頭辞について**を参照してください。



注記

関連するアカウント全体のロールを作成したときにカスタム ARN パスを指定した場合、カスタムパスは自動的に検出されます。カスタムパスは、後のステップで作成するときに、クラスター固有の Operator ロールに適用されます。

- 環境変数を設定する場合は、次のコマンドを実行して、公開または非公開で利用可能な API と Ingress を使用して、初期マシンプールが1つ含まれるクラスターを作成します。

```
$ rosa create cluster --private --cluster-name=<cluster_name> \
  --mode=auto --hosted-cp --operator-roles-prefix=$OPERATOR_ROLES_PREFIX \
  --oidc-config-id=$ODIC_CONFIG --subnet-ids=$SUBNET_IDS
```

- 環境変数を設定する場合は、次のコマンドを実行して、単一の初期マシンプール、公開されている API、および公開されている Ingress が含まれるクラスターを作成します。

```
$ rosa create cluster --cluster-name=<cluster_name> --mode=auto --hosted-cp --
operator-roles-prefix=$OPERATOR_ROLES_PREFIX --oidc-config-id=$ODIC_CONFIG
--subnet-ids=$SUBNET_IDS
```

- 次のコマンドを実行して、クラスターのステータスを確認します。

```
$ rosa describe cluster --cluster=<cluster_name>
```

以下の **State** フィールドの変更は、クラスターインストールの進捗として出力に表示されません。

- **pending (Preparing account)**
- **installing (DNS setup in progress)**
- **installing**
- **ready**



注記

インストールが失敗した場合や、**State** フィールドが 10 分以上 **ready** に変わらない場合は、インストールのトラブルシューティングのドキュメントで詳細を確認してください。詳細は、**インストールのトラブルシューティング**を参照してください。Red Hat サポートにサポートを依頼する手順は、**Red Hat OpenShift Service on AWS のサポートを受ける**を参照してください。

- Red Hat OpenShift Service on AWS インストールプログラムのログを監視して、クラスター作成の進行状況を追跡します。ログを確認するには、次のコマンドを実行します。

```
$ rosa logs install --cluster=<cluster_name> --watch \<.>
```

<.> オプション: インストールの進行中に新しいログメッセージを監視するには、**--watch** 引数を使用します。

1.4. 次のステップ

- [ROSA クラスターへのアクセス](#)

1.5. 関連情報

- 手動モードを使用して ROSA クラスターをデプロイする手順は、[カスタマイズを使用したクラスターの作成](#) を参照してください。
- STS を使用する Red Hat OpenShift Service on AWS をデプロイするのに必要な AWS Identity Access Management (IAM) リソースの詳細は、[STS を使用するクラスターの IAM リソースについて](#) を参照してください。
- セキュリティグループの要件については、[追加のカスタムセキュリティグループ](#) を参照してください。
- オプションで Operator ロール名接頭辞を設定する方法の詳細は、[カスタム Operator IAM ロール接頭辞について](#) を参照してください。
- STS を使用する ROSA をインストールするための前提条件の詳細は、[STS を使用する ROSA の AWS の前提条件](#) を参照してください。
- **auto** モードと **manual** モードを使用して必要な STS リソースを作成する方法の詳細は、[自動デプロイメントモードと手動デプロイメントモードについて](#) を参照してください。
- AWS IAM で OpenID Connect (OIDC) アイデンティティプロバイダーの使用に関する詳細は、AWS ドキュメントの [Creating OpenID Connect \(OIDC\) identity providers](#) を参照してください。
- ROSA クラスターのインストールのトラブルシューティングの詳細は、[インストールのトラブルシューティング](#) を参照してください。
- Red Hat サポートにサポートを依頼する手順は、[Red Hat OpenShift Service on AWS のサポートを受ける](#) を参照してください。

第2章 カスタム AWS KMS 暗号鍵を使用した ROSA WITH HCP クラスターの作成

カスタムの AWS Key Management Service (KMS) キーを使用して、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane クラスターを作成します。

2.1. ROSA WITH HCP の前提条件

ROSA with HCP クラスターを作成するには、次のものがが必要です。

- 設定された仮想プライベートクラウド (VPC)
- アカウント全体のロール
- OIDC 設定
- オペレーターのロール

2.1.1. ROSA with HCP クラスター用の仮想プライベートクラウドの作成

ROSA with HCP クラスターを作成するには、Virtual Private Cloud (VPC) が必要です。次の方法を使用して VPC を作成できます。

- Terraform テンプレートを使用して VPC を作成する
- AWS コンソールで VPC リソースを手動で作成する



注記

Terraform の手順はテストとデモンストレーションを目的としています。独自のインストールでは、独自に使用するために VPC にいくつかの変更を加える必要があります。また、この Terraform スクリプトを使用するときは、クラスターをインストールする予定のリージョンと同じリージョンにあることを確認する必要があります。これらの例では、**us-east-2** を使用します。

Terraform を使用した Virtual Private Cloud の作成

Terraform は、確立されたテンプレートを使用してさまざまなリソースを作成できるツールです。次のプロセスでは、必要に応じてデフォルトのオプションを使用して、ROSA with HCP クラスターを作成します。Terraform の使用の詳細は、関連情報を参照してください。

前提条件

- マシンに Terraform バージョン 1.4.0 以降がインストールされている。
- マシンに Git がインストールされている。

手順

1. シェルプロンプトを開き、次のコマンドを実行して Terraform VPC リポジトリのクローンを作成します。

```
$ git clone https://github.com/openshift-cs/terraform-vpc-example
```

2. 次のコマンドを実行して、作成したディレクトリーに移動します。

```
$ cd terraform-vpc-example
```

3. 次のコマンドを実行して、Terraform ファイルを開始します。

```
$ terraform init
```

このプロセスが完了すると、初期化を確認するメッセージが表示されます。

4. 既存の Terraform テンプレートに基づいて VPC Terraform プランを構築するには、**plan** コマンドを実行します。AWS リージョンを含める必要があります。クラスター名の指定を選択できます。**terraform plan** が完了すると、**rosa.tfplan** ファイルが **hypershift-tf** ディレクトリーに追加されます。オプションの詳細は、[Terraform VPC リポジトリーの README ファイル](#) を参照してください。

```
$ terraform plan -out rosa.tfplan -var region=<region> [-var cluster_name=<cluster_name>]
```

5. 次のコマンドを実行して、このプランファイルを適用して VPC を構築します。

```
$ terraform apply rosa.tfplan
```

6. 任意: 次のコマンドを実行して、Terraform でプロビジョニングされたプライベート、パブリック、およびマシンプールのサブネット ID の値を環境変数としてキャプチャーし、ROSA with HCP クラスターを作成するときに使用できます。

```
$ export SUBNET_IDS=$(terraform output -raw cluster-subnets-string)
```

検証

- 次のコマンドを使用して、変数が正しく設定されたことを確認できます。

```
$ echo $SUBNET_IDS
```

出力例

```
$ subnet-0a6a57e0f784171aa,subnet-078e84e5b10ecf5b0
```

関連情報

- ニーズに合わせて VPC をカスタマイズするときに使用できるすべてのオプションの詳細なリストは、[Terraform VPC](#) リポジトリーを参照してください。

Virtual Private Cloud を手動で作成する

Terraform を使用する代わりに Virtual Private Cloud (VPC) を手動で作成することを選択した場合は、[AWS コンソールの VPC ページ](#) に移動します。VPC は、次の表に示す要件を満たしている必要があります。

表2.1 VPC の要件

要件	詳細
VPC 名	クラスターを作成するときは、特定の VPC 名と ID が必要です。
CIDR 範囲	VPC CIDR 範囲はマシンの CIDR と一致する必要があります。
アベイラビリティゾーン	単一ゾーンの場合は1つの可用性ゾーンが必要で、複数ゾーンの場合は3つの可用性ゾーンが必要です。
パブリックサブネット	パブリッククラスターには、NAT ゲートウェイを備えたパブリックサブネットが1つ必要です。プライベートクラスターにはパブリックサブネットは必要ありません。
DNS ホスト名と解決	DNS ホスト名と解決が有効になっていることを確認する必要があります。

関連情報

- [Get Started with Amazon VPC](#)
- [HashiCorp Terraform ドキュメント](#)

2.1.2. アカウント全体の STS ロールおよびポリシーの作成

Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) を使用して、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane クラスターを作成する前に、Operator ポリシーを含む、必要なアカウント全体のロールとポリシーを作成します。



注記

ROSA with HCP クラスターには、AWS 管理ポリシーがアタッチされたアカウントと Operator ロールが必要です。顧客管理のポリシーはサポートされていません。ROSA with HCP クラスターの AWS 管理ポリシーの詳細は、[AWS managed policies for ROSA account roles](#) を参照してください。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- 利用可能な AWS サービスクォータがある。
- AWS コンソールで ROSA サービスを有効にしている。
- インストールホストに、最新の ROSA CLI (**rosa**) をインストールして設定している。
- ROSA CLI を使用して Red Hat アカウントにログインしている。

手順

- AWS アカウントに存在しない場合は、次のコマンドを実行して、必要なアカウント全体の STS ロールを作成し、ポリシーをアタッチします。

```
$ rosa create account-roles --hosted-cp --mode auto --yes
```

- オプション: 次のコマンドを実行して、接頭辞を環境変数として設定します。

```
$ export ACCOUNT_ROLES_PREFIX=<account_role_prefix>
```

- 次のコマンドを実行して、変数の値を表示します。

```
$ echo $ACCOUNT_ROLES_PREFIX
```

出力例

```
ManagedOpenShift
```

ROSA の AWS 管理 IAM ポリシーの詳細は、[AWS managed IAM policies for ROSA](#) を参照してください。

2.1.3. OpenID Connect 設定の作成

ROSA with HCP クラスターを使用する場合は、クラスターを作成する前に OpenID Connect (OIDC) 設定を作成する必要があります。この設定は、OpenShift Cluster Manager で使用するために登録されています。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。
- Red Hat OpenShift Service on AWS の AWS 前提条件を完了している。
- インストールホストに、最新の Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) をインストールして設定している。

手順

- AWS リソースと一緒に OIDC 設定を作成するには、次のコマンドを実行します。

```
$ rosa create oidc-config --mode=auto --yes
```

このコマンドは次の情報を返します。

出力例

```
? Would you like to create a Managed (Red Hat hosted) OIDC Configuration Yes
I: Setting up managed OIDC configuration
I: To create Operator Roles for this OIDC Configuration, run the following command and
remember to replace <user-defined> with a prefix of your choice:
  rosa create operator-roles --prefix <user-defined> --oidc-config-id 13cdr6b
If you are going to create a Hosted Control Plane cluster please include '--hosted-cp'
I: Creating OIDC provider using 'arn:aws:iam::4540112244:user/userName'
```



```
? Create the OIDC provider? Yes
```

```
l: Created OIDC provider with ARN 'arn:aws:iam::4540112244:oidc-provider/dvbwgdztaeq9o.cloudfront.net/13cdr6b'
```

クラスターを作成するときは、OIDC 設定 ID を指定する必要があります。CLI 出力では、`--mode auto` のこの値が提供されます。それ以外の場合は、`--mode manual` の `aws` CLI 出力に基づいてこれらの値を決定する必要があります。

- オプション: OIDC 設定 ID を変数として保存して、後で使用できます。次のコマンドを実行して変数を保存します。

```
$ export OIDC_ID=<oidc_config_id> ①
```

- ① 上記の出力例では、OIDC 設定 ID は 13cdr6b です。

- 次のコマンドを実行して、変数の値を表示します。

```
$ echo $OIDC_ID
```

出力例

```
13cdr6b
```

検証

- ユーザー組織に関連付けられているクラスターで使用できる可能な OIDC 設定をリストできます。以下のコマンドを実行します。

```
$ rosa list oidc-config
```

出力例

```
ID                MANAGED ISSUER URL
SECRET ARN
2330dbs0n8m3chkk25gkkcd8pnj3lk2 true
https://dvwgdztaeq9o.cloudfront.net/2330dbs0n8m3chkk25gkkcd8pnj3lk2
233hvnjrjoqu14jltk6lhbhf2tj11f8un false https://oidc-r7u1.s3.us-east-1.amazonaws.com
aws:secretsmanager:us-east-1:242819244:secret:rosa-private-key-oidc-r7u1-tM3MDN
```

2.1.4. Operator のロールとポリシーの作成

ROSA with HCP クラスターを使用する場合は、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane (HCP) デプロイメントに必要な Operator IAM ロールを作成する必要があります。クラスター Operator は、Operator のロールを使用して、バックエンドストレージ、クラウドプロバイダーの認証情報、クラスターへの外部アクセスの管理など、クラスター操作を実行するために必要な一時的なアクセス許可を取得します。

前提条件

- ROSA with HCP の AWS の前提条件を完了している。

- インストールホストに、最新の Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) をインストールして設定している。
- アカウント全体の AWS ロールを作成している。

手順

1. 次のコマンドを使用して、接頭辞名を環境変数に設定します。

```
$ export OPERATOR_ROLES_PREFIX=<prefix_name>
```

2. Operator ロールを作成するには、次のコマンドを実行します。

```
$ rosa create operator-roles --hosted-cp --prefix=$OPERATOR_ROLES_PREFIX --oidc-
config-id=$OIDC_ID --installer-role-arn
arn:aws:iam::${AWS_ACCOUNT_ID}:role/${ACCOUNT_ROLES_PREFIX}-HCP-ROSA-
Installer-Role
```

次の内訳は、Operator ロール作成のオプションを示しています。

```
$ rosa create operator-roles --hosted-cp
--prefix=$OPERATOR_ROLES_PREFIX 1
--oidc-config-id=$OIDC_ID 2
--installer-role-arn
arn:aws:iam::${AWS_ACCOUNT_ID}:role/${ACCOUNT_ROLES_PREFIX}-HCP-ROSA-
Installer-Role 3
```

- 1** これらの Operator ロールを作成するときは、接頭辞を指定する必要があります。そうしないとエラーが発生します。演算子接頭辞は、このセクションの関連情報を参照してください。
- 2** この値は、ROSA with HCP クラスター用に作成した OIDC 設定 ID です。
- 3** この値は、ROSA アカウントロールの作成時に作成したインストーラーロールの ARN です。

ROSA with HCP クラスター用の正しいロールを作成するには、**--hosted-cp** パラメーターを含める必要があります。このコマンドは次の情報を返します。

出力例

```
? Role creation mode: auto
? Operator roles prefix: <pre-filled_prefix> 1
? OIDC Configuration ID: 23soa2bgvpek9kmes9s7os0a39i13qm4 |
https://dvbwgdztaeq9o.cloudfront.net/23soa2bgvpek9kmes9s7os0a39i13qm4 2
? Create hosted control plane operator roles: Yes
W: More than one Installer role found
? Installer role ARN: arn:aws:iam::4540112244:role/<prefix>-HCP-ROSA-Installer-Role
? Permissions boundary ARN (optional):
I: Reusable OIDC Configuration detected. Validating trusted relationships to operator roles:
I: Creating roles using 'arn:aws:iam::4540112244:user/<userName>'
I: Created role '<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials'
I: Created role '<prefix>-openshift-cloud-network-config-controller-cloud-credenti' with ARN
```

```
'arn:aws:iam::4540112244:role/<prefix>-openshift-cloud-network-config-controller-cloud-
credenti'
I: Created role '<prefix>-kube-system-kube-controller-manager' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-kube-controller-manager'
I: Created role '<prefix>-kube-system-capac-controller-manager' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-capac-controller-manager'
I: Created role '<prefix>-kube-system-control-plane-operator' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-control-plane-operator'
I: Created role '<prefix>-kube-system-kms-provider' with ARN
'arn:aws:iam::4540112244:role/<prefix>-kube-system-kms-provider'
I: Created role '<prefix>-openshift-image-registry-installer-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-image-registry-installer-cloud-credentials'
I: Created role '<prefix>-openshift-ingress-operator-cloud-credentials' with ARN
'arn:aws:iam::4540112244:role/<prefix>-openshift-ingress-operator-cloud-credentials'
I: To create a cluster with these roles, run the following command:
rosa create cluster --sts --oidc-config-id 23soa2bgvpek9kmes9s7os0a39i13qm4 --operator-
roles-prefix <prefix> --hosted-cp
```

- ❶ このフィールドには、最初の作成コマンドで設定した接頭辞が事前に入力されます。
- ❷ このフィールドでは、ROSA with HCP クラスター用に作成した OIDC 設定を選択する必要があります。

これで、Operator ロールが作成され、ROSA with HCP クラスターの作成に使用できるようになりました。

検証

- ROSA アカウントに関連付けられている Operator ロールをリスト表示できます。以下のコマンドを実行します。

```
$ rosa list operator-roles
```

出力例

```
I: Fetching operator roles
ROLE PREFIX AMOUNT IN BUNDLE
<prefix> 8
? Would you like to detail a specific prefix Yes ❶
? Operator Role Prefix: <prefix>
ROLE NAME ROLE ARN
VERSION MANAGED
<prefix>-kube-system-capac-controller-manager
arn:aws:iam::4540112244:role/<prefix>-kube-system-capac-controller-manager
4.13 No
<prefix>-kube-system-control-plane-operator
arn:aws:iam::4540112244:role/<prefix>-kube-system-control-plane-operator
4.13 No
<prefix>-kube-system-kms-provider
arn:aws:iam::4540112244:role/<prefix>-kube-system-kms-provider 4.13
No
<prefix>-kube-system-kube-controller-manager
arn:aws:iam::4540112244:role/<prefix>-kube-system-kube-controller-manager
4.13 No
```

```

<prefix>-openshift-cloud-network-config-controller-cloud-credenti
arn:aws:iam::4540112244:role/<prefix>-openshift-cloud-network-config-controller-cloud-
credenti 4.13 No
<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-cluster-csi-drivers-ebs-cloud-credentials
4.13 No
<prefix>-openshift-image-registry-installer-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-image-registry-installer-cloud-credentials
4.13 No
<prefix>-openshift-ingress-operator-cloud-credentials
arn:aws:iam::4540112244:role/<prefix>-openshift-ingress-operator-cloud-credentials
4.13 No

```

- 1 コマンドを実行すると、AWS アカウントに関連付けられているすべての接頭辞が表示され、この接頭辞に関連付けられているロールの数が記録されます。これらのロールとその詳細をすべて表示する必要がある場合は、詳細プロンプトで "Yes" と入力すると、これらのロールが詳細とともにリストされます。

2.1.5. カスタム AWS KMS キーを使用した ROSA クラスターの作成

ノードのルートボリューム、etcd データベース、またはその両方の暗号化に使用するお客様提供の KMS キーを使用して、Red Hat OpenShift Service on AWS (ROSA) クラスターを作成できます。両方にそれぞれ異なる KMS キー ARN を指定できます。



注記

ROSA with HCP は、お客様提供の KMS キーを使用して永続ボリュームを暗号化するように **default** ストレージクラスを自動設定しません。これは、インストール後にクラスター内で設定できるものです。

手順

1. 次のコマンドを実行して、AWS の顧客管理のカスタム KMS キーを作成します。

```

$ KMS_ARN=$(aws kms create-key --region $AWS_REGION --description 'Custom ROSA
Encryption Key' --tags TagKey=red-hat,TagValue=true --query KeyMetadata.Arn --output
text)

```

このコマンドで、後の手順のために、このカスタムキーの Amazon リソースネーム (ARN) 出力が保存されます。



注記

お客様は、お客様の KMS キーに必要な **--tags TagKey=red-hat,TagValue=true** 引数を指定する必要があります。

2. 次のコマンドを実行して、KMS キーが作成されたことを確認します。

```

$ echo $KMS_ARN

```

3. AWS アカウント ID を環境変数に設定します。

```
$ AWS_ACCOUNT_ID=<aws_account_id>
```

4. 前述の手順で作成したアカウント全体のインストーラーロールと Operator ロールの ARN を、ファイルの **Statement.Principal.AWS** セクションに追加します。次の例では、デフォルトの **ManagedOpenShift-HCP-ROSA-Installer-Role** ロールの ARN が追加されます。

```
{
  "Version": "2012-10-17",
  "Id": "key-rosa-policy-1",
  "Statement": [
    {
      "Sid": "Enable IAM User Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::${AWS_ACCOUNT_ID}:root"
      },
      "Action": "kms:*",
      "Resource": "*"
    },
    {
      "Sid": "Installer Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::${AWS_ACCOUNT_ID}:role/ManagedOpenShift-HCP-ROSA-Installer-Role"
      },
      "Action": [
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:GenerateDataKeyWithoutPlaintext"
      ],
      "Resource": "*"
    },
    {
      "Sid": "ROSA KubeControllerManager Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::${AWS_ACCOUNT_ID}:role/<operator_role_prefix>-kubernetes-kube-controller-manager"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "ROSA KMS Provider Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::${AWS_ACCOUNT_ID}:role/<operator_role_prefix>-kubernetes-kms-provider"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:DescribeKey"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "*"
  },
  {
    "Sid": "ROSA NodeManager Permissions",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::${AWS_ACCOUNT_ID}:role/<operator_role_prefix>-kube-
system-capac-controller-manager"
    },
    "Action": [
      "kms:DescribeKey",
      "kms:GenerateDataKeyWithoutPlaintext",
      "kms:CreateGrant"
    ],
    "Resource": "*"
  }
]
}

```

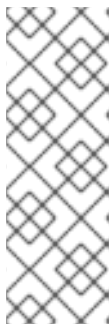
5. 次のコマンドを実行して、作成されたポリシーファイルの詳細を確認します。

```
$ cat rosa-key-policy.json
```

6. 次のコマンドを実行して、新しく生成されたキーポリシーをカスタム KMS キーに適用します。

```
$ aws kms put-key-policy --key-id $KMS_ARN \
--policy file://rosa-key-policy.json \
--policy-name default
```

7. 次のコマンドを実行してクラスターを作成します。



注記

クラスター名が 15 文字を超える場合、***.openshiftapps.com** にプロビジョニングされたクラスターのサブドメインとして自動生成されたドメイン接頭辞が含まれます。

サブドメインをカスタマイズするには、**--domain-prefix** フラグを使用します。ドメイン接頭辞は 15 文字を超えてはならず、一意である必要があり、クラスターの作成後に変更できません。

```
$ rosa create cluster --cluster-name <cluster_name> \
--subnet-ids <private_subnet_id>,<public_subnet_id> \
--sts \
--mode auto \
--machine-cidr 10.0.0.0/16 \
--compute-machine-type m5.xlarge \
--hosted-cp \
--region <aws_region> \
--oidc-config-id $OIDC_ID \
--kms-key-arn $KMS_ARN \ ①
--etcd-encryption-kms-arn $KMS_ARN \ ②
--operator-roles-prefix $OPERATOR_ROLES_PREFIX
```

-

- ① この KMS キー ARN は、すべてのワーカーノードのルートボリュームを暗号化するために使用します。etcd データベースの暗号化のみが必要な場合は必要ありません。
- ② この KMS キー ARN は、etcd データベースの暗号化に使用します。etcd データベースは、デフォルトでは常に AES 暗号ブロックを使用して暗号化されますが、代わりに KMS キーを使用して暗号化することもできます。ノードのルートボリュームの暗号化のみが必要な場合は必要ありません。

検証

[OpenShift Cluster Manager](#) を使用して、KMS キーが機能することを確認できます。

1. [OpenShift Cluster Manager](#) に移動し、**Instances** を選択します。
2. インスタンスを選択します。
3. **Storage** タブをクリックします。
4. KMS key ID をコピーします。
5. **Key Management Service** を検索して選択します。
6. コピーした KMS key ID を **Filter** フィールドに入力します。

2.2. 次のステップ

- [ROSA クラスターへのアクセス](#)

2.3. 関連情報

- CLI を使用してクラスターを作成する方法については、[CLI を使用した ROSA with HCP クラスターの作成](#) を参照してください。
- 手動モードを使用して ROSA クラスターをデプロイする手順は、[カスタマイズを使用したクラスターの作成](#) を参照してください。
- STS を使用する Red Hat OpenShift Service on AWS をデプロイするのに必要な AWS Identity Access Management (IAM) リソースの詳細は、[STS を使用するクラスターの IAM リソースについて](#) を参照してください。
- オプションで Operator ロール名接頭辞を設定する方法の詳細は、[カスタム Operator IAM ロール接頭辞について](#) を参照してください。
- STS を使用する ROSA をインストールするための前提条件の詳細は、[STS を使用する ROSA の AWS の前提条件](#) を参照してください。
- **auto** モードと **manual** モードを使用して必要な STS リソースを作成する方法の詳細は、[自動デプロイメントモードと手動デプロイメントモードについて](#) を参照してください。
- AWS IAM での OpenID Connect (OIDC) ID プロバイダーの使用の詳細は、[Creating OpenID Connect \(OIDC\) identity providers](#) を参照してください。
- ROSA クラスターのインストールのトラブルシューティングの詳細は、[インストールのトラブルシューティング](#) を参照してください。

- Red Hat サポートにサポートを依頼する手順は、[Red Hat OpenShift Service on AWS のサポートを受ける](#) を参照してください。

第3章 ROSA WITH HCP でのプライベートクラスターの作成

このドキュメントでは、Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane (HCP) のプライベートクラスターを作成する方法について説明します。

3.1. AWS プライベートクラスターの作成

ROSA コマンドラインインターフェイス (CLI) **rosa** を使用して、ROSA with HCP に複数のアベイラビリティゾーン (Multi-AZ) を持つプライベートクラスターを作成できます。

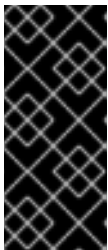
前提条件

- 利用可能な AWS サービスクォータがある。
- AWS コンソールで ROSA サービスを有効にしている。
- インストールホストに、最新バージョンの ROSA CLI をインストールして設定している。

手順

Hosted Control Plane を使用したクラスターの作成には、10 分ほどかかる場合があります。

1. 少なくとも1つのプライベートサブネットを持つ VPC を作成します。マシンの Classless Inter-Domain Routing (CIDR) が、仮想プライベートクラウドの CIDR と一致していることを確認します。詳細は、[独自の VPC を使用するための要件](#) および [VPC 検証](#) を参照してください。



重要

ファイアウォールを使用する場合は、ROSA が機能するのに必要なサイトにアクセスできるようにファイアウォールを設定する必要があります。

詳細は、「AWS PrivateLink ファイアウォールの前提条件」セクションを参照してください。

2. 次のコマンドを実行して、アカウント全体の IAM ロールを作成します。

```
$ rosa create account-roles --hosted-cp
```

3. 次のコマンドを実行して、OIDC 設定を作成します。

```
$ rosa create oidc-config --mode=auto --yes
```

OIDC 設定の ID を保存します。Operator ロールの作成に必要なためです。

出力例

```
I: Setting up managed OIDC configuration
I: To create Operator Roles for this OIDC Configuration, run the following command and
remember to replace <user-defined> with a prefix of your choice:
  rosa create operator-roles --prefix <user-defined> --oidc-config-id
  28s4avcdt2l318r1jbk3ifmimkurk384
If you are going to create a Hosted Control Plane cluster please include '--hosted-cp'
```

```
I: Creating OIDC provider using 'arn:aws:iam::46545644412:user/user'
I: Created OIDC provider with ARN 'arn:aws:iam::46545644412:oidc-provider/oidc.op1.openshiftapps.com/28s4avcdt2l318r1jbk3ifmimkurk384'
```

4. 次のコマンドを実行して、Operator ロールを作成します。

```
$ rosa create operator-roles --hosted-cp --prefix <operator_roles_prefix> --oidc-config-id
<oidc_config_id> --installer-role-arn
arn:aws:iam::<account_roles_prefix>:role/<account_roles_prefix>-HCP-ROSA-Installer-
Role
```

5. 次のコマンドを実行して、ROSA with HCP プライベートクラスターを作成します。

```
$ rosa create cluster --private --cluster-name=<cluster-name> --sts --mode=auto --hosted-cp
--operator-roles-prefix <operator_role_prefix> --oidc-config-id <oidc_config_id> [--machine-
cidr=<VPC CIDR>/16] --subnet-ids=<private-subnet-id1>[,<private-subnet-id2>,<private-
subnet-id3>]
```

6. 以下のコマンドを実行して Pod のステータスを確認します。クラスターの作成中、出力の **State** フィールドは **pending** から **installing** に移行し、最後に **ready** に移行します。

```
$ rosa describe cluster --cluster=<cluster_name>
```



注記

インストールが失敗する場合や、10 分経っても **State** フィールドが **ready** に変わらない場合は、「関連情報」セクションの「Red Hat OpenShift Service on AWS のインストールのトラブルシューティング」ドキュメントを参照してください。

7. 以下のコマンドを実行して、OpenShift インストーラーのログでクラスターの進捗を追跡します。

```
$ rosa logs install --cluster=<cluster_name> --watch
```

3.2. API にアクセスするための AWS セキュリティーグループの設定

ROSA with HCP プライベートクラスターでは、お客様の VPC で公開される AWS PrivateLink エンドポイントにデフォルトのセキュリティーグループがあります。このセキュリティーグループは、VPC 内に存在するリソースまたは VPC CIDR 範囲に関連付けられた IP アドレスを持つリソースにのみ制限されている PrivateLink エンドポイントにアクセスできます。VPC ピアリングとトランジットゲートウェイを介した VPC 外部のエンティティーへのアクセスを許可するには、別のセキュリティーグループを作成して PrivateLink エンドポイントに割り当てて、必要なアクセス権を付与する必要があります。

前提条件

- 企業ネットワークまたは他の VPC に接続性がある。
- VPC 内でセキュリティーグループを作成して割り当てる権限を持っている。

手順

1. 次のコマンドを実行して、クラスター名を環境変数として設定します。

```
$ export CLUSTER_NAME=<cluster_name>
```

次のコマンドを実行すると、変数が設定されたことを確認できます。

```
$ echo $CLUSTER_NAME
```

出力例

```
hcp-private
```

2. 次のコマンドを実行して、VPC エンドポイント (VPCE) ID と VPC ID を見つけます。

```
$ read -r VPCE_ID VPC_ID <<< $(aws ec2 describe-vpc-endpoints --filters  
"Name=tag:api.openshift.com/id,Values=$(rosa describe cluster -c ${CLUSTER_NAME} -o  
yaml | grep '^id: ' | cut -d' ' -f2)" --query 'VpcEndpoints[][VpcEndpointId,VpcId]' --output text)
```

3. 次のコマンドを実行して、セキュリティーグループを作成します。

```
$ export SG_ID=$(aws ec2 create-security-group --description "Granting API access to  
${CLUSTER_NAME} from outside of VPC" --group-name "${CLUSTER_NAME}-api-sg" --  
vpc-id $VPC_ID --output text)
```

4. 次のコマンドを実行して、セキュリティーグループに Ingress ルールを追加します。

```
$ aws ec2 authorize-security-group-ingress --group-id $SG_ID --ip-permissions  
FromPort=443,ToPort=443,IpProtocol=tcp,IpRanges=[{CidrIp=0.0.0.0/0}]
```

5. 次のコマンドを実行して、新しいセキュリティーグループを VPCE に追加します。

```
$ aws ec2 modify-vpc-endpoint --vpc-endpoint-id $VPCE_ID --add-security-group-ids  
$SG_ID
```

これで、ROSA with HCP プライベートクラスターを使用して API にアクセスできるようになりました。

3.3. 次のステップ

[アイデンティティプロバイダーの設定](#)

3.4. 関連情報

- [AWS PrivateLink ファイアウォールの前提条件](#)
- [STS を使用する ROSA のデプロイメントワークフローの概要](#)
- [ROSA クラスターの削除](#)
- [アーキテクチャーモデル](#)
- [Red Hat OpenShift Service on AWS のインストールのトラブルシューティング](#)

第4章 ROSA WITH HCP クラスターでの NODE TUNING OPERATOR の使用

Red Hat OpenShift Service on AWS (ROSA) with Hosted Control Plane (HCP) は、ROSA with HCP クラスター上のノードのパフォーマンスを向上させる Node Tuning Operator をサポートしています。ノード調整設定を作成する前に、カスタム調整仕様を作成する必要があります。

目的

Node Tuning Operator は、TuneD デーモンを調整することでノードレベルのチューニングを管理し、パフォーマンスプロファイルコントローラーを使用して低レイテンシーのパフォーマンスを実現するのに役立ちます。ほとんどの高パフォーマンスアプリケーションでは、一定レベルのカーネルのチューニングが必要です。Node Tuning Operator は、ノードレベルの `sysctl` の統一された管理インターフェイスをユーザーに提供し、ユーザーが指定するカスタムチューニングを追加できるよう柔軟性を提供します。

Operator は、コンテナ化された Red Hat OpenShift Service on AWS の TuneD デーモンを Kubernetes デーモンセットとして管理します。これにより、カスタムチューニング仕様が、デーモンが認識する形式でクラスターで実行されるすべてのコンテナ化された TuneD デーモンに渡されます。デーモンは、ノードごとに1つずつ、クラスターのすべてのノードで実行されます。

コンテナ化された TuneD デーモンによって適用されるノードレベルの設定は、プロファイルの変更をトリガーするイベントで、または終了シグナルの受信および処理によってコンテナ化された TuneD デーモンが正常に終了する際にロールバックされます。

Node Tuning Operator は、Performance Profile コントローラーを使用して自動チューニングを実装し、Red Hat OpenShift Service on AWS アプリケーションの低レイテンシーパフォーマンスを実現します。

クラスター管理者は、以下のようなノードレベルの設定を定義するパフォーマンスプロファイルを設定します。

- カーネルを `kernel-rt` に更新します。
- ハウスキーピング用の CPU を選択します。
- 実行中のワークロード用の CPU を選択します。



注記

現在、CPU 負荷分散の無効化は `cgroup v2` ではサポートされていません。その結果、`cgroup v2` が有効になっている場合は、パフォーマンスプロファイルから望ましい動作が得られない可能性があります。パフォーマンスプロファイルを使用している場合は、`cgroup v2` を有効にすることは推奨されません。

Node Tuning Operator は、バージョン 4.1 以降の標準の Red Hat OpenShift Service on AWS インストールに含まれています。



注記

Red Hat OpenShift Service on AWS の以前のバージョンでは、OpenShift アプリケーションの低レイテンシーパフォーマンスを実現する自動チューニングを実装するために Performance Addon Operator が使用されていました。Red Hat OpenShift Service on AWS 4.11 以降では、この機能は Node Tuning Operator の一部です。

4.1. カスタムチューニング仕様

Operator のカスタムリソース (CR) には 2 つの重要なセクションがあります。1 つ目のセクションの **profile:** は TuneD プロファイルおよびそれらの名前のリストです。2 つ目の **recommend:** は、プロファイル選択ロジックを定義します。

複数のカスタムチューニング仕様は、Operator の namespace に複数の CR として共存できます。新規 CR の存在または古い CR の削除は Operator によって検出されます。既存のカスタムチューニング仕様はすべてマージされ、コンテナ化された TuneD デーモンの適切なオブジェクトは更新されます。

管理状態

Operator 管理の状態は、デフォルトの Tuned CR を調整して設定されます。デフォルトで、Operator は Managed 状態であり、**spec.managementState** フィールドはデフォルトの Tuned CR に表示されません。Operator Management 状態の有効な値は以下のとおりです。

- Managed: Operator は設定リソースが更新されるとそのオペランドを更新します。
- Unmanaged: Operator は設定リソースへの変更を無視します。
- Removed: Operator は Operator がプロビジョニングしたオペランドおよびリソースを削除します。

プロファイルデータ

profile: セクションは、TuneD プロファイルおよびそれらの名前をリスト表示します。

```
{
  "profile": [
    {
      "name": "tuned_profile_1",
      "data": "# TuneD profile specification\n[main]\nsummary=Description of tuned_profile_1\nprofile\n\n[sysctl]\nnet.ipv4.ip_forward=1\n# ... other sysctl's or other TuneD daemon plugins\nsupported by the containerized TuneD\n"
    },
    {
      "name": "tuned_profile_n",
      "data": "# TuneD profile specification\n[main]\nsummary=Description of tuned_profile_n\nprofile\n\n# tuned_profile_n profile settings\n"
    }
  ]
}
```

推奨プロファイル

profile: 選択ロジックは、CR の **recommend:** セクションによって定義されます。**recommend:** セクションは、選択基準に基づくプロファイルの推奨項目のリストです。

```
"recommend": [
  {
    "recommend-item-1": details_of_recommendation,
    # ...
    "recommend-item-n": details_of_recommendation,
  }
]
```

リストの個別項目:

```
{
  "profile": [
    {
      # ...
    }
  ],
  "recommend": [
    {
      "profile": <tuned_profile_name>, ❶
      "priority": { <priority>, ❷
    },
    "match": [ ❸
      {
        "label": <label_information> ❹
      },
    ]
  },
]
```

- ❶ 一致に適用する TuneD プロファイル。たとえば、**tuned_profile_1** です。
- ❷ プロファイルの順序付けの優先度。数値が小さいほど優先度が高くなります (**0** が最も高い優先度になります)。
- ❸ 省略した場合、優先度の高いプロファイルが先に一致しない限り、プロファイル一致とみなされません。
- ❹ プロファイル一致アイテムのラベル。

<match> は、以下のように再帰的に定義されるオプションの一覧です。

```
"match": [
  {
    "label": ❶
  },
]
```

- ❶ ノードまたは Pod のラベル名。

<match> が省略されない場合、ネストされたすべての <match> セクションが **true** に評価される必要もあります。そうでない場合には **false** が想定され、それぞれの <match> セクションのあるプロファイルは適用されず、推奨されません。そのため、ネスト化 (子の <match> セクション) は論理 AND 演算子として機能します。これとは逆に、<match> 一覧のいずれかの項目が一致する場合は、<match> の一覧全体が **true** に評価されます。そのため、リストは論理 OR 演算子として機能します。

例: ノードまたは Pod のラベルベースのマッチング

```
[
  {
    "match": [
```

```

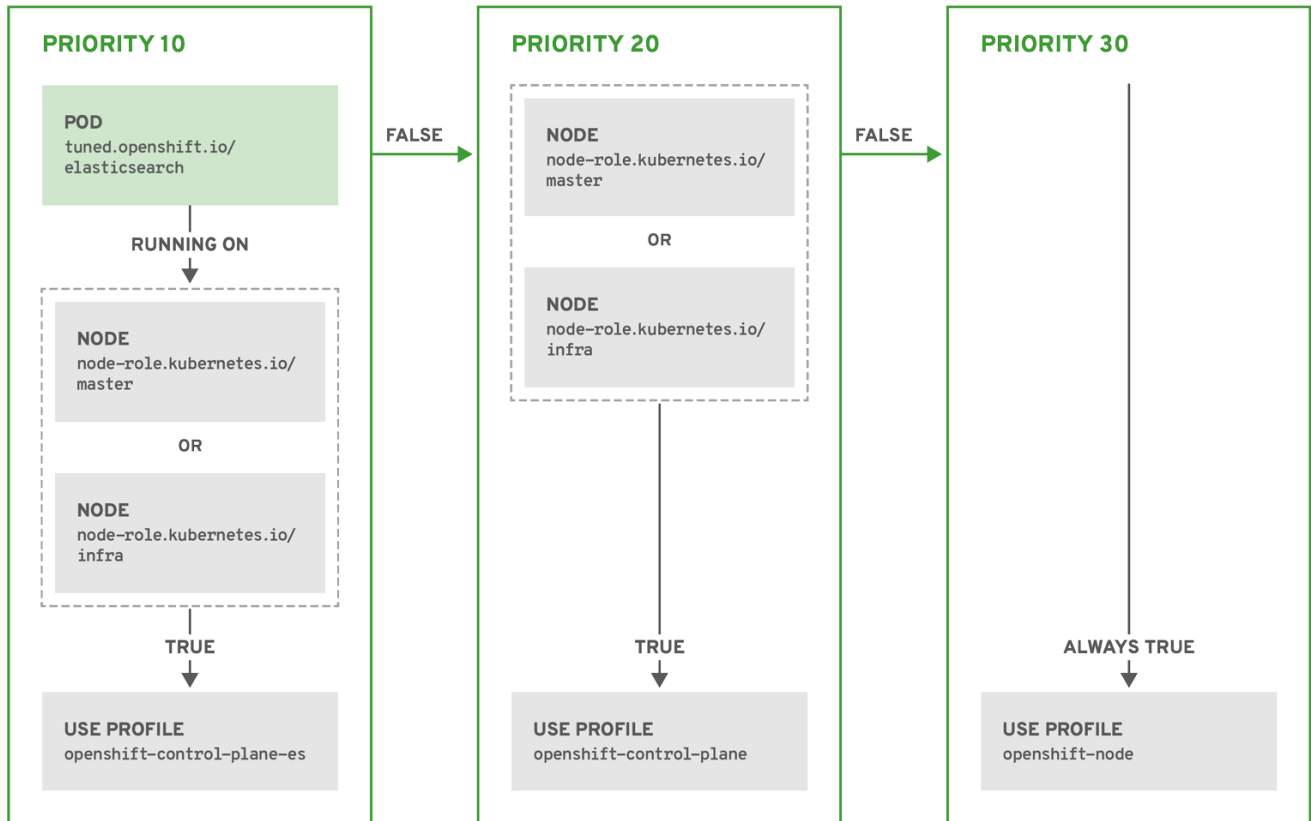
    {
      "label": "tuned.openshift.io/elasticsearch",
      "match": [
        {
          "label": "node-role.kubernetes.io/master"
        },
        {
          "label": "node-role.kubernetes.io/infra"
        }
      ],
      "type": "pod"
    }
  ],
  "priority": 10,
  "profile": "openshift-control-plane-es"
},
{
  "match": [
    {
      "label": "node-role.kubernetes.io/master"
    },
    {
      "label": "node-role.kubernetes.io/infra"
    }
  ],
  "priority": 20,
  "profile": "openshift-control-plane"
},
{
  "priority": 30,
  "profile": "openshift-node"
}
]

```

上記のコンテナ化された TuneD デーモンの CR は、プロファイルの優先順位に基づいてその **recommend.conf** ファイルに変換されます。最も高い優先順位 (**10**) を持つプロファイルは **openshift-control-plane-es** であるため、これが最初に考慮されます。指定されたノードで実行されるコンテナ化された TuneD デーモンは、同じノードに **tuned.openshift.io/elasticsearch** ラベルが設定された Pod が実行されているかどうかを確認します。これがない場合は、**<match>** セクション全体が **false** として評価されます。このラベルを持つこのような Pod がある場合に、**<match>** セクションが **true** に評価されるようにするには、ノードラベルを **node-role.kubernetes.io/master** または **node-role.kubernetes.io/infra** にする必要もあります。

優先順位が **10** のプロファイルのラベルが一致した場合は、**openshift-control-plane-es** プロファイルが適用され、その他のプロファイルは考慮されません。ノード/Pod ラベルの組み合わせが一致しない場合は、2 番目に高い優先順位プロファイル (**openshift-control-plane**) が考慮されます。このプロファイルは、コンテナ化された TuneD Pod が **node-role.kubernetes.io/master** または **node-role.kubernetes.io/infra** ラベルを持つノードで実行される場合に適用されます。

最後に、プロファイル **openshift-node** には最低の優先順位である **30** が設定されます。これには **<match>** セクションがないため、常に一致します。これは、より高い優先順位の他のプロファイルが指定されたノードで一致しない場合に **openshift-node** プロファイルを設定するために、最低の優先順位のノードが適用される汎用的な (catch-all) プロファイルとして機能します。



OPENSHIFT_10_0319

例: マシンプールベースのマッチング

```

{
  "apiVersion": "tuned.openshift.io/v1",
  "kind": "Tuned",
  "metadata": {
    "name": "openshift-node-custom",
    "namespace": "openshift-cluster-node-tuning-operator"
  },
  "spec": {
    "profile": [
      {
        "data": "[main]\nsummary=Custom OpenShift node profile with an additional kernel\nparameter\ninclude=openshift-\nnode\n[bootloader]\ncmdline_openshift_node_custom=+skew_tick=1\n",
        "name": "openshift-node-custom"
      }
    ],
    "recommend": [
      {
        "priority": 20,
        "profile": "openshift-node-custom"
      }
    ]
  }
}

```

クラウドプロバイダー固有の TuneD プロファイル

この機能により、すべてのクラウドプロバイダー固有のノードに、Red Hat OpenShift Service on AWS クラスター上の特定のクラウドプロバイダーに合わせて特別に調整された TuneD プロファイルを簡単に割り当てることができます。別のノードラベルを追加したり、ノードをマシンプールにグループ化したりする必要はありません。

この機能は、`<cloud-provider>://<cloud-provider-specific-id>` の形式で `spec.providerID` ノードオブジェクト値を利用して、NTO オペランドコンテナの `<cloud-provider>` の値で `/var/lib/tuned/provider` ファイルを書き込みます。その後、このファイルのコンテンツは TuneD により、プロバイダー `provider-<cloud-provider>` プロファイル (存在する場合) を読み込むために使用されます。

`openshift-control-plane` および `openshift-node` プロファイルの両方の設定を継承する `openshift` プロファイルは、条件付きプロファイルの読み込みを使用してこの機能を使用するよう更新されるようになりました。現時点で、NTO や TuneD にクラウドプロバイダー固有のプロファイルは含まれていません。ただし、すべてのクラウドプロバイダー固有のクラスターノードに適用されるカスタムプロファイル `provider-<cloud-provider>` を作成できます。

GCE クラウドプロバイダープロファイルの例

```
{
  "apiVersion": "tuned.openshift.io/v1",
  "kind": "Tuned",
  "metadata": {
    "name": "provider-gce",
    "namespace": "openshift-cluster-node-tuning-operator"
  },
  "spec": {
    "profile": [
      {
        "data": "[main]\nsummary=GCE Cloud provider-specific profile\n# Your tuning for GCE Cloud provider goes here.\n",
        "name": "provider-gce"
      }
    ]
  }
}
```



注記

プロファイルの継承により、`provider-<cloud-provider>` プロファイルで指定された設定は、`openshift` プロファイルとその子プロファイルによって上書きされます。

4.2. ROSA WITH HCP のノードチューニング設定の作成

Red Hat OpenShift Service on AWS (ROSA) CLI (`rosa`) を使用してチューニング設定を作成できます。

前提条件

- ROSA CLI の最新バージョンをダウンロードしている。
- 最新バージョンのクラスターがある。
- ノードチューニング用に設定された仕様ファイルがある。

手順

1. 次のコマンドを実行して、チューニング設定を作成します。

```
$ rosa create tuning-config -c <cluster_id> --name <name_of_tuning> --spec-path
<path_to_spec_file>
```

spec.json ファイルへのパスを指定する必要があります。指定しない場合、コマンドはエラーを返します。

出力例

```
$ I: Tuning config 'sample-tuning' has been created on cluster 'cluster-example'.
$ I: To view all tuning configs, run 'rosa list tuning-configs -c cluster-example'
```

検証

- 次のコマンドを使用して、アカウントによって適用されている既存のチューニング設定を確認できます。

```
$ rosa list tuning-configs -c <cluster_name> [-o json]
```

設定リストに必要な出力のタイプを指定できます。

- 出力タイプを指定しないと、チューニング設定の ID と名前が表示されます。

出力タイプを指定しない出力例

```
ID                NAME
20468b8e-edc7-11ed-b0e4-0a580a800298 sample-tuning
```

- **json** などの出力タイプを指定すると、チューニング設定を JSON テキストとして受け取ります。



注記

次の JSON 出力には、読みやすくするために改行が含まれています。この JSON 出力は、JSON 文字列内の改行を削除しない限り無効です。

JSON 出力を指定したサンプル出力

```
[
  {
    "kind": "TuningConfig",
    "id": "20468b8e-edc7-11ed-b0e4-0a580a800298",
    "href":
"/api/clusters_mgmt/v1/clusters/23jbsevqb22l0m58ps39ua4trff9179e/tuning_configs/20468b8e-edc7-11ed-b0e4-0a580a800298",
    "name": "sample-tuning",
    "spec": {
      "profile": [
        {
          "data": "[main]\nsummary=Custom OpenShift profile\ninclude=openshift-node\n\n[sysctl]nvm.dirty_ratio=\"55\"\n",
```

```

        "name": "tuned-1-profile"
      }
    ],
    "recommend": [
      {
        "priority": 20,
        "profile": "tuned-1-profile"
      }
    ]
  }
}
]

```

4.3. ROSA WITH HCP のノードチューニング設定の変更

Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) を使用して、ノードチューニング設定を表示し、更新できます。

前提条件

- ROSA CLI の最新バージョンをダウンロードしている。
- 最新バージョンのクラスターがある。
- クラスターにノード調整設定が追加されている。

手順

1. チューニング設定を表示するには、**rosa description** コマンドを使用します。

```

$ rosa describe tuning-config -c <cluster_id> ❶
  --name <name_of_tuning> ❷
  [-o json] ❸

```

この仕様ファイルの次の項目は次のとおりです。

- ❶ ノードチューニング設定を適用する、所有するクラスターのクラスター ID を指定します。
- ❷ チューニング設定の名前を指定します。
- ❸ 任意で、出力タイプを指定できます。出力を指定しない場合は、チューニング設定の ID と名前のみが表示されます。

出力タイプを指定しない出力例

```

Name: sample-tuning
ID: 20468b8e-edc7-11ed-b0e4-0a580a800298
Spec: {
  "profile": [
    {
      "data": "[main]\nsummary=Custom OpenShift profile\ninclude=openshift-
node\n\n[sysctl]\nvm.dirty_ratio=55\n",

```

```

        "name": "tuned-1-profile"
      }
    ],
    "recommend": [
      {
        "priority": 20,
        "profile": "tuned-1-profile"
      }
    ]
  }
}

```

JSON 出力を指定したサンプル出力

```

{
  "kind": "TuningConfig",
  "id": "20468b8e-edc7-11ed-b0e4-0a580a800298",
  "href":
"/api/clusters_mgmt/v1/clusters/23jbsevqb2210m58ps39ua4trff9179e/tuning_configs/20468b8e-
edc7-11ed-b0e4-0a580a800298",
  "name": "sample-tuning",
  "spec": {
    "profile": [
      {
        "data": "[main]\nsummary=Custom OpenShift profile\ninclude=openshift-
node\n\n[sysctl]\nvm.dirty_ratio=55\n",
        "name": "tuned-1-profile"
      }
    ],
    "recommend": [
      {
        "priority": 20,
        "profile": "tuned-1-profile"
      }
    ]
  }
}

```

2. チューニング設定を確認した後、**rosa edit** コマンドを使用して既存の設定を編集します。

```

$ rosa edit tuning-config -c <cluster_id> --name <name_of_tuning> --spec-path
<path_to_spec_file>

```

このコマンドでは、**spec.json** ファイルを使用して設定を編集します。

検証

- **rosa description** コマンドを再度実行して、**spec.json** ファイルに加えた変更がチューニング設定で更新されていることを確認します。

```

$ rosa describe tuning-config -c <cluster_id> --name <name_of_tuning>

```

出力例

```

Name: sample-tuning

```

```

ID: 20468b8e-edc7-11ed-b0e4-0a580a800298
Spec: {
  "profile": [
    {
      "data": "[main]\nsummary=Custom OpenShift profile\ninclude=openshift-
node\n\n[sysctl]\nvm.dirty_ratio=55\n",
      "name": "tuned-2-profile"
    }
  ],
  "recommend": [
    {
      "priority": 10,
      "profile": "tuned-2-profile"
    }
  ]
}

```

4.4. ROSA WITH HCP のノードチューニング設定の削除

Red Hat OpenShift Service on AWS (ROSA) CLI (**rosa**) を使用して、チューニング設定を削除できます。



注記

マシンプールで参照されているチューニング設定は削除できません。チューニング設定を削除する前に、すべてのマシンプールからチューニング設定を削除する必要があります。

前提条件

- ROSA CLI の最新バージョンをダウンロードしている。
- 最新バージョンのクラスターがある。
- クラスターに、削除したいノードチューニング設定がある。

手順

- チューニング設定を削除するには、次のコマンドを実行します。

```
$ rosa delete tuning-config -c <cluster_id> <name_of_tuning>
```

クラスター上のチューニング設定が削除されました。

出力例

```
? Are you sure you want to delete tuning config sample-tuning on cluster sample-cluster? Yes
l: Successfully deleted tuning config 'sample-tuning' from cluster 'sample-cluster'
```