



# Red Hat Advanced Cluster Security for Kubernetes 3.70

統合

Red Hat Advanced Cluster Security for Kubernetes の統合



# Red Hat Advanced Cluster Security for Kubernetes 3.70 統合

---

Red Hat Advanced Cluster Security for Kubernetes の統合

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/Integrating.ent file |.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## 概要

本書では、イメージレジストリー、Slack、PagerDuty、JIRA、メール、および汎用 Webhook を使用した統合など、Red Hat Advanced Cluster Security for Kubernetes で一般的な統合を設定する方法について説明します。

## 目次

<b>第1章 イメージレジストリーとの統合</b> .....	<b>4</b>
1.1. 自動設定	5
1.2. AMAZON ECR の統合	5
1.3. イメージレジストリーを手動で設定する	5
1.3.1. OpenShift Container Platform レジストリーの手動設定	5
1.3.2. Amazon Elastic Container Registry を手動で設定する	6
1.3.2.1. Amazon ECR で assumeroles を使用する	7
1.3.2.1.1. コンテナ IAM を使用した AssumeRole の設定	7
1.3.2.1.2. コンテナ IAM を使用せずに AssumeRole を設定する	8
1.3.2.1.3. RHACS での AssumeRole の設定	9
1.3.3. Google Container Registry を手動で設定する	10
1.3.4. Google Artifact Registry を手動で設定する	11
1.3.5. Microsoft Azure Container Registry を手動で設定する	12
1.3.6. JFrog Artifactory を手動で設定する	12
1.3.7. Docker Trusted Registry を手動で設定する	13
1.3.8. Quay Container Registry を手動で設定する	13
1.3.9. IBM Cloud Container Registry を手動で設定する	14
1.3.10. Red Hat コンテナレジストリーを手動で設定する	14
<b>第2章 CI システムとの統合</b> .....	<b>16</b>
2.1. ビルドポリシーの設定	16
2.1.1. 既存のビルドフェーズポリシーを確認する	16
2.1.2. 新しいシステムポリシーの作成	16
2.2. レジストリー統合の設定	18
2.2.1. 既存のレジストリー統合を確認する	19
2.2.1.1. 関連情報	19
2.3. アクセスの設定	19
2.3.1. roxctl CLI を使用した認証	19
2.3.2. バイナリーをダウンロードして roxctl CLI をインストール	20
2.3.2.1. Linux への roxctl CLI のインストール	20
2.3.2.2. macOS への roxctl CLI のインストール	21
2.3.2.3. Windows への roxctl CLI のインストール	21
2.3.3. コンテナから roxctl CLI の実行	21
2.4. CI パイプラインとの統合	22
2.4.1. Jenkins の使用	22
2.4.2. CircleCI の使用	22
<b>第3章 PAGERDUTY との統合</b> .....	<b>25</b>
3.1. PAGERDUTY の設定	25
3.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	25
3.3. ポリシー通知の設定	26
<b>第4章 SLACK との統合</b> .....	<b>27</b>
4.1. SLACK の設定	27
4.1.1. さまざまな Slack チャンネルにアラートを送信する	27
4.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	28
4.3. ポリシー通知の設定	28
<b>第5章 一般的な WEBHOOK を使用した統合</b> .....	<b>30</b>
5.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	30
5.2. ポリシー通知の設定	31

<b>第6章 SUMO LOGIC との統合</b> .....	<b>33</b>
6.1. SUMO LOGIC の設定	33
6.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	33
6.3. ポリシー通知の設定	34
6.4. SUMO LOGIC でアラートを表示する	34
<b>第7章 GOOGLE CLOUD STORAGE との統合</b> .....	<b>35</b>
7.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	35
7.1.1. Google Cloud Storage でオンデマンドバックアップを実行する	36
7.1.1.1. 関連情報	36
<b>第8章 SYSLOG プロトコルを使用した統合</b> .....	<b>37</b>
8.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUB との SYSLOG 統合の設定	37
<b>第9章 AMAZON S3 との統合</b> .....	<b>39</b>
9.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES で AMAZON S3 統合を設けます。	39
9.2. AMAZON S3 でのオンデマンドバックアップの実行	40
9.3. 関連情報	40
<b>第10章 GOOGLE CLOUD SECURITY コマンドセンターとの統合</b> .....	<b>41</b>
10.1. GOOGLE CLOUD SCC の設定	41
10.2. GOOGLE CLOUD SCC と統合するための RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	41
10.3. ポリシー通知の設定	42
<b>第11章 SPLUNK との統合</b> .....	<b>43</b>
11.1. HTTP イベントコレクターの使用	43
11.1.1. Splunk に HTTP イベントコレクターを追加する	43
11.1.1.1. HTTP イベントコレクターの有効化	44
11.1.2. Red Hat Advanced Cluster Security for Kubernetes での Splunk 統合の設定	44
11.1.3. ポリシー通知の設定	44
11.2. STACKROX KUBERNETES SECURITY PLATFORM アドオンの使用	45
11.2.1. Splunk アドオンのインストールと設定	45
<b>第12章 イメージ脆弱性スキャナーとの統合</b> .....	<b>47</b>
12.1. ANCHORE スキャナーとの統合	47
12.2. CLAIR との統合	47
12.3. DOCKER TRUSTED REGISTRY スキャナーとの統合	48
12.4. GOOGLE CONTAINER REGISTRY との統合	49
12.5. QUAY CONTAINER REGISTRY との統合	49
12.6. TENABLE との統合	50
<b>第13章 JIRA との統合</b> .....	<b>51</b>
13.1. JIRA の設定	51
13.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定	51
13.2.1. さまざまな Jira プロジェクトで問題を作成する	52
13.2.2. Jira でのカスタムプライオリティーの設定	53
13.3. ポリシー通知の設定	53
13.4. JIRA 統合のトラブルシューティング	54
<b>第14章 メールとの統合</b> .....	<b>55</b>
14.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES プラットフォームでのメールプラグインの設定	55
14.2. ポリシー通知の設定	56



## 第1章 イメージレジストリーとの統合

Red Hat Advanced Cluster Security for Kubernetes は、さまざまなイメージレジストリーと統合されているため、イメージを理解し、イメージの使用にセキュリティーポリシーを適用できます。

イメージレジストリーと統合すると、イメージの作成日や Dockerfile の詳細 (イメージレイヤーを含む) などの重要なイメージの詳細を表示できます。

Red Hat Advanced Cluster Security for Kubernetes をレジストリーに統合すると、イメージをデプロイする前、またはクラスターにすでにデプロイされている既存のイメージにイメージをスキャンしたり、イメージコンポーネントを表示したり、セキュリティーポリシーを適用したりできます。



### 注記

イメージレジストリーと統合する場合、Red Hat Advanced Cluster Security for Kubernetes はレジストリー内のすべてのイメージをスキャンしません。Red Hat Advanced Cluster Security for Kubernetes は、次の場合にのみイメージをスキャンします。

- デプロイメントでイメージを使用する
- **roxctl** CLI を使用してイメージを確認します
- 継続的インテグレーション (CI) システムを使用して、セキュリティーポリシーを適用します

Red Hat Advanced Cluster Security for Kubernetes は、以下を含む主要なイメージレジストリーと統合できます。

- [Amazon Elastic Container Registry \(ECR\)](#)
- [Docker Hub](#)
- [Docker トラステッドレジストリー](#)
- [Google Container Registry \(GCR\)](#)
- [Google Artifact Registry](#)
- [IBM Cloud Container Registry \(ICR\)](#)
- [JFrog Artifactory](#)
- [Microsoft Azure Container Registry \(ACR\)](#)
- [Red Hat Quay](#)
- [Red Hat コンテナレジストリー](#)
- [Sonatype Nexus](#)
- [Tenable.io](#)
- [Docker Registry HTTP API](#) を使用するその他のレジストリー



## 1.1. 自動設定

Red Hat Advanced Cluster Security for Kubernetes には、Docker Hub などの標準レジストリーとのデフォルトの統合が含まれています。また、イメージのプルシークレットなど、モニターされるクラスターにあるアーティファクトに基づいて、インテグレーションを自動的に設定することもできます。通常、レジストリー統合を手動で設定する必要はありません。



### 重要

GCR レジストリーを使用している場合、Red Hat Advanced Cluster Security for Kubernetes はレジストリー統合を自動的に作成しません。

## 1.2. AMAZON ECR の統合

Amazon ECR 統合の場合、以下の条件が満たされると、Red Hat Advanced Cluster Security for Kubernetes は ECR レジストリー統合を自動的に生成します。

- クラスターのクラウドプロバイダーは AWS です。
- クラスターのノードには、インスタンス ID およびアクセス管理 (IAM) ロールの関連付けがあり、インスタンスメタデータサービスはノードで利用可能です。たとえば、Amazon Elastic Kubernetes Service (EKS) を使用してクラスターを管理する場合、このロールは EKS Node IAM ロールと呼ばれます。
- Instance IAM ロールには、デプロイする ECR レジストリーへのアクセス権限を付与する IAM ポリシーがあります。

上記の条件が満たされると、Red Hat Advanced Cluster Security for Kubernetes は ECR レジストリーからプルするデプロイメントをモニターし、それらの ECR 統合を自動的に生成します。これらのインテグレーションは、自動的に生成された後に編集できます。

## 1.3. イメージレジストリーを手動で設定する

GCR を使用している場合、イメージレジストリー統合を手動で作成する必要があります。

### 1.3.1. OpenShift Container Platform レジストリーの手動設定

Red Hat Advanced Cluster Security for Kubernetes を OpenShift Container Platform のビルトインコンテナイメージレジストリーと統合できます。

#### 前提条件

- OpenShift Container Platform レジストリーでの認証にはユーザー名とパスワードが必要。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Generic Docker Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。

- a. **Integration name:** インテグレーションの名前。
  - b. **Endpoint:** レジストリーのアドレス。
  - c. **UsernameとPassword**
5. レジストリーへの接続時に TLS 証明書を使用していない場合は、**Disable TLS certificate validation (insecure)** を選択します。
  6. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
  7. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
  8. **Save** を選択します。

### 1.3.2. Amazon Elastic Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を使用して、Amazon Elastic Container Registry (ECR) の統合を手動で作成および変更できます。Amazon ECR からデプロイする場合、Amazon ECR レジストリーのインテグレーションは通常自動的に生成されます。ただし、デプロイメント外のイメージをスキャンするために、独自にインテグレーションを作成したい場合があります。自動生成されるインテグレーションのパラメーターを変更することもできます。たとえば、自動生成された Amazon ECR 統合で使用される認証方法を変更して、AssumeRole 認証またはその他の承認モデルを使用することができます。



#### 重要

自動生成される ECR 統合に加えた変更を消去するには、統合を削除し、Red Hat Advanced Cluster Security for Kubernetes は Amazon ECR からイメージをデプロイする際に自動生成されるパラメーターで新しい統合を作成します。

#### 前提条件

- Amazon Identity and Access Management (IAM) アクセスキー ID およびシークレットアクセスキーが必要です。または、**kiam** や **kube2iam** などのノードレベルの IAM プロキシを使用することもできる。
- アクセスキーには、ECR への読み取りアクセス権が必要。詳細は、[How do I create an AWS access key?](#) を参照のこと。
- Amazon Elastic Kubernetes Service (EKS) で Red Hat Advanced Cluster Security for Kubernetes を実行していて、別の Amazon アカウントの ECR と統合する場合は、最初に ECR でリポジトリポリシーステートメントを設定する必要があります。[Setting a repository policy statement](#) の手順に従い、**Actions** で、Amazon ECR API オペレーションの次のスコープを選択する。
  - ecr:BatchCheckLayerAvailability
  - ecr:BatchGetImage
  - ecr:DescribeImages
  - ecr:GetDownloadUrlForLayer
  - ecr:ListImages

## 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで **Amazon ECR** を選択します。
3. **New integration** をクリックするか、自動生成されたインテグレーションのいずれかをクリックして開き、**Edit** をクリックします。
4. 以下のフィールドの詳細を入力または変更します。
  - a. **Update stored credentials**: アクセスキーやパスワードなどのクレデンシャルを更新せずにインテグレーションを変更する場合には、このボックスをクリアします。
  - b. **Integration name**: インテグレーションの名前。
  - c. **Registry ID**: レジストリーの ID。
  - d. **Endpoint**: レジストリーのアドレス。このフィールドは、**AssumeRole** オプションが選択されている場合には有効ではありません。
  - e. **Region**: レジストリーのリージョン (例: **us-west-1**)。
5. IAM を使用している場合は、**Use Container IAM role** を選択します。それ以外の場合は、**Use Container IAM role** ボックスの選択を解除し、アクセス **Access key ID** および **Secret access key** を入力します。
6. **AssumeRole** 認証を使用している場合は、**Use AssumeRole** を選択し、以下のフィールドの詳細を入力します。
  - a. **AssumeRole ID**: 引き受けるロールの ID。
  - b. **AssumeRole External ID** (オプション): **AssumeRole** で外部 ID を使用している場合は、ここに入力できます。
7. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
8. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
9. **Save** を選択します。

### 1.3.2.1. Amazon ECR で assumeroles を使用する

**AssumeRole** を使用すると、各ユーザーのパーミッションを手動で設定しなくても、AWS リソースへのアクセスを許可できます。代わりに、必要な権限を持つロールを定義すると、ユーザーにそのロールを引き受けるためのアクセス権が付与されます。**AssumeRole** を使用すると、よりきめ細かい権限を付与、取り消し、またはその他の方法で一般的に管理できます。

#### 1.3.2.1.1. コンテナ IAM を使用した AssumeRole の設定

Red Hat Advanced Cluster Security for Kubernetes で **AssumeRole** を使用する前に、まずそれを設定する必要があります。

## 手順

1. EKS クラスターの IAM OIDC プロバイダーを有効にします。

```
$ eksctl utils associate-iam-oidc-provider --cluster <cluster name> --approve
```

- EKS クラスターの IAM ロールを作成します。
- 新しく作成したロールをサービスアカウントに関連付けます。

```
$ kubectl -n stackrox annotate sa central eks.amazonaws.com/role-arn=arn:aws:iam::67890:role/<role-name>
```

- セントラルを再起動して、変更を適用します。

```
$ kubectl -n stackrox delete pod -l app=central
```

- 必要に応じて、ロールが別のロールを引き受けることを許可するポリシーにロールを割り当てます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::<ecr-registry>:role/<assumerole-readonly>" ❶
    }
  ]
}
```

- ❶ **<assumerole-readonly>** を引き受けたいロールに置き換えます。

- 引き受けるロールの信頼関係を更新します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::<ecr-registry>:role/<role-name>" ❶
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

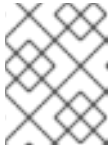
- ❶ **<role-name>** は、以前に作成した新しいロールと一致する必要があります。

### 1.3.2.1.2. コンテナ IAM を使用せずに AssumeRole を設定する

コンテナ IAM なしで AssumeRole を使用するには、アクセスと秘密鍵を使用して、[プログラムによるアクセス権を持つ AWS ユーザー](#) として認証する必要があります。

## 手順

1. AssumeRole ユーザーが ECR レジストリーと同じアカウントにあるか、別のアカウントにあるかに応じて、次のいずれかを行う必要があります。
  - ロールを引き受けるユーザーが ECR レジストリーと同じアカウントにいる場合は、必要な権限で新しいロールを作成します。



### 注記

ロールを作成するときに、必要に応じて任意の信頼できるエンティティを選択できます。ただし、作成後に変更する必要があります。

- または、ユーザーが ECR レジストリーとは異なるアカウントにいる場合は、ECR レジストリーにアクセスし、その信頼関係を定義するためのアクセス許可を提供する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::<ecr-registry>:role/<assumerole-readonly>" ❶
    }
  ]
}
```

❶ **<assumerole-readonly>** を引き受けたいロールに置き換えます。

2. Principal フィールドの下にユーザー ARN を含めることにより、ロールの信頼関係を設定します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::<ecr-registry>:user/<role-name>"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

### 1.3.2.1.3. RHACS での AssumeRole の設定

ECR で AssumeRole を設定した後、AssumeRole を使用して、Red Hat Advanced Cluster Security for Kubernetes を Amazon Elastic Container Registry (ECR) と統合できます。

## 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで **Amazon ECR** を選択します。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name**: 統合の名前。
  - b. **Registry ID**: レジストリーの ID。
  - c. **Region**: レジストリーのリージョン (例: **us-west-1**)。
5. IAM を使用している場合は、**Use container IAM role** を選択します。それ以外の場合は、**Use custom IAM role** ボックスの選択を解除し、**Access key ID** および **Secret access key** を入力します。
6. AssumeRole を使用している場合は、**Use AssumeRole** を選択し、以下のフィールドの詳細を入力します。
  - a. **AssumeRole ID**: 引き受けるロールの ID。
  - b. **AssumeRole External ID** (オプション): **AssumeRole** で外部 ID を使用している場合は、ここに入力できます。
7. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
8. **Save** を選択します。

### 1.3.3. Google Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Google Container Registry (GCR) と統合できます。

#### 前提条件

- サービスアカウントキーが必要です。
- 関連するサービスアカウントは、レジストリーにアクセスできる必要があります。ユーザーおよび他のプロジェクトに GCR へのアクセスを許可する方法は、[Configuring access control](#) を参照してください。
- [GCR Container Analysis](#) を使用している場合は、サービスアカウントに次のロールも付与する必要があります。
  - コンテナー分析ノートビューアー
  - コンテナー分析発生状況ビューアー
  - ストレージオブジェクトビューアー

## 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで、**Google Container Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Type**: **Registry** を選択します。
  - c. **Registry Endpoint**: レジストリーのアドレス。
  - d. **Project**: Google Cloud プロジェクト名。
  - e. **Service account key (JSON)** 認証に使用するサービスアカウントキー。
5. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
6. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
7. **Save** を選択します。

### 1.3.4. Google Artifact レジストリーを手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Google Artifact Registry と統合できます。

#### 前提条件

- **Artifact Registry Reader** Identity and Access Management (IAM) ロール **roles/artifactregistry.reader** でサービスアカウントキーが必要です。

## 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで、**Google Artifact Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Registry endpoint**: レジストリーのアドレス。
  - c. **Project**: Google Cloud プロジェクト名。
  - d. **Service account key (JSON)** 認証に使用するサービスアカウントキー。
5. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。

6. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
7. **Save** を選択します。

### 1.3.5. Microsoft Azure Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Microsoft Azure と統合できます。

#### 前提条件

- 認証には、ユーザー名とパスワードが必要です。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで **Microsoft Azure Container Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name:** インテグレーションの名前。
  - b. **Endpoint:** レジストリーのアドレス。
  - c. **Username**と**Password**
5. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
6. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
7. **Save** を選択します。

### 1.3.6. JFrog Artifactory を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を JFrog Artifactory と統合できます。

#### 前提条件

- JFrog Artifactory で認証するには、ユーザー名とパスワードが必要。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**JFrog Artifactory** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name:** インテグレーションの名前。
  - b. **Endpoint:** レジストリーのアドレス。



### c. UsernameとPassword

5. レジストリーへの接続時に TLS 証明書を使用していない場合は、**Disable TLS certificate validation (insecure)** を選択します。
6. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
7. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
8. **Save** を選択します。

## 1.3.7. Docker Trusted Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Docker Trusted Registry と統合できます。

### 前提条件

- Docker Trusted Registry で認証するためのユーザー名とパスワードが必要です。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Docker Trusted Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Type: Registry** を選択します。
  - c. **Endpoint**: レジストリーのアドレス。
  - d. **UsernameとPassword**
5. レジストリーへの接続時に TLS 証明書を使用していない場合は、**Disable TLS certificate validation (insecure)** を選択します。
6. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
7. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
8. **Save** を選択します。

## 1.3.8. Quay Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Quay Container Registry と統合できます。

### 前提条件

- Quay Container Registry で認証するには OAuth トークンが必要。

## 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで **Red Hat Quay.io** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Type**: **Registry** を選択します。
  - c. **Endpoint**: レジストリーのアドレス。
  - d. **OAuth トークン**
5. レジストリーへの接続時に TLS 証明書を使用していない場合は、**Disable TLS certificate validation (insecure)** を選択します。
6. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
7. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
8. **Save** を選択します。

### 1.3.9. IBM Cloud Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を IBM Cloud Container Registry と統合できます。

#### 前提条件

- IBM Cloud Container Registry で認証するための API キーが必要。

## 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで、**IBM Cloud Container Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Endpoint**: レジストリーのアドレス。
  - c. **API key**.
5. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
6. **Save** を選択します。

### 1.3.10. Red Hat コンテナレジストリーを手動で設定する

Red Hat Advanced Cluster Security for Kubernetes を Red Hat Container Registry と統合できます。

### 前提条件

- Red Hat Container Registry での認証にはユーザー名とパスワードが必要。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで **Red Hat Registry** を選択します。
3. **New integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration name**: インテグレーションの名前。
  - b. **Endpoint**: レジストリーのアドレス。
  - c. **Username**と**Password**
5. レジストリーへの接続をテストせずに統合を作成する場合は、**Create integration without testing** を選択します。
6. **Test** を選択して、選択したレジストリーとの統合が機能していることをテストします。
7. **Save** を選択します。

## 第2章 CI システムとの統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) は、さまざまな継続的インテグレーション (CI) 製品と統合され、イメージをデプロイする前にビルド時およびデプロイ時のセキュリティルールを適用できます。

Red Hat Advanced Cluster Security for Kubernetes は、イメージがビルドされてレジストリーにプッシュされた後、CI パイプラインに統合されます。最初にイメージをプッシュすることで、開発者は、他の CI テストの失敗、リンター違反、またはその他の問題とともに、ポリシー違反に対処しながら、アーティファクトのテストを続行できます。

可能であれば、バージョン管理システムを設定して、Red Hat Advanced Cluster Security for Kubernetes チェックを含むビルドステージが失敗した場合にプルリクエストまたはマージリクエストがマージされないようにブロックする必要があります。

Red Hat Advanced Cluster Security for Kubernetes のインストールにアクセスして、イメージが設定したビルドフェーズポリシーに準拠しているかどうかを確認することにより、CI 製品との統合が機能します。ポリシー違反がある場合は、ポリシーの説明、論理的根拠、修正手順などの詳細なメッセージがコンソールログに表示されます。各ポリシーには、オプションの適用設定が含まれています。ビルドフェーズの適用のためにポリシーをマークした場合、そのポリシーに失敗すると、クライアントはゼロ以外のエラーコードで終了します。

Red Hat Advanced Cluster Security for Kubernetes を CI システムと統合するには、次の手順に従います。

1. [ビルドポリシーを設定](#) します。
2. [レジストリー統合を設定](#) します。
3. Red Hat Advanced Cluster Security for Kubernetes インスタンスへの [アクセスを設定](#) します。
4. [CI パイプラインとの統合](#)。

### 2.1. ビルドポリシーの設定

ビルド中に Red Hat Cluster Security for Kubernetes ポリシーを確認するには、最初に、コンテナライフサイクルのビルドフェーズに適用されるポリシーを設定する必要があります。次に、ビルド中にイメージがプッシュされるレジストリーと統合する必要があります。

#### 2.1.1. 既存のビルドフェーズポリシーを確認する

RHACS ポータルを使用して、Red Hat Advanced Cluster Security for Kubernetes で設定した既存のビルドフェーズポリシーを確認します。

##### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. グローバル検索を使用して、**Lifecycle Stage:Build** を検索します。

#### 2.1.2. 新しいシステムポリシーの作成

デフォルトのポリシーを使用することに加えて、Red Hat Advanced Cluster Security for Kubernetes でカスタムポリシーを作成することもできます。

## 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. **+ New Policy** をクリックします。
3. ポリシーの **Name** を入力します。
4. ポリシーの **Severity** レベルを選択します: クリティカル、高、中、または低。
5. **Build**、**Deploy**、または **Runtime** から、ポリシーが適用される **Lifecycle Stages** を選択します。複数のステージを選択できます。



## 注記

CIシステムと統合するための新しいポリシーを作成する場合は、ライフサイクルステージとして **Build** を選択します。

- ビルド時ポリシーは、CVE や Dockerfile 手順などのイメージフィールドに適用されます。
  - デプロイ時のポリシーには、すべてのビルド時のポリシー基準を含めることができます。また、特権モードでの実行や Docker デーモンソケットのマウントなど、クラスター設定からのデータを取得することもできます。
  - ランタイムポリシーには、すべてのビルド時とデプロイ時のポリシー基準、およびランタイム中のプロセス実行に関するデータを含めることができます。
6. **Description**、**Rationale**、および **Remediation** フィールドにポリシーに関する情報を入力します。CIがビルドを検証すると、これらのフィールドのデータが表示されます。したがって、ポリシーを説明するすべての情報を含めてください。
  7. **Categories** ドロップダウンメニューからカテゴリーを選択します。
  8. このポリシーの違反が発生したときにアラート通知を受信する通知ドロップダウンメニューから **Notifications** 機能を選択します。



## 注記

アラート通知を受信するには、Red Hat Advanced Cluster Security for Kubernetes を Webhook、Jira、PagerDuty などの通知プロバイダーと統合する必要があります。通知機能は、通知プロバイダーを Red Hat Advanced Cluster Security for Kubernetes と統合した場合にのみ表示されます。

9. 特定のクラスター、namespace、またはラベルに対してのみこのポリシーを有効にするには、**Restrict to Scope** を使用します。複数のスコープを追加したり、namespaces とラベルの RE2 構文で正規表現を使用したりすることもできます。
10. **Exclude by Scope** を使用して、デプロイメント、クラスター、namespaces、およびラベルを除外します。このフィールドは、指定したエンティティーにポリシーが適用されないことを示します。複数のスコープを追加したり、namespaces とラベルの RE2 構文で正規表現を使用したりすることもできます。ただし、デプロイメントの選択に正規表現を使用することはできません。
11. **除外されたイメージ (ビルドライフサイクルのみ)** の場合、ポリシーの違反をトリガーしたくないすべてのイメージをリストから選択します。



## 注記

**除外されたイメージ (ビルドライフサイクルのみ)** 設定は、継続的インテグレーションシステム (ビルドライフサイクルステージ) でイメージをチェックする場合にのみ適用されます。このポリシーを使用して、実行中のデプロイメント (デプロイライフサイクルステージ) またはランタイムアクティビティ (ランタイムライフサイクルステージ) をチェックする場合、効果はありません。

12. **Policy Criteria** セクションで、ポリシーをトリガーする属性を設定します。
13. パネルヘッダーで **Next** を選択します。
14. 新しいポリシーパネルには、ポリシーを有効にした場合にトリガーされる違反のプレビューが表示されます。
15. パネルヘッダーで **Next** を選択します。
16. ポリシーの適用動作を選択します。適用設定は、**Lifecycle Stages** オプションで選択したステージでのみ使用できます。ポリシーを適用して違反を報告するには、**ON** を選択します。違反のみを報告するには、**OFF** を選択します。



## 注記

適用の振る舞いは、ライフサイクルの各ステージで異なります。

- **Build** ステージでは、イメージがポリシーの条件に一致すると、Red Hat Advanced Cluster Security for Kubernetes は CI ビルドに失敗します。
- **Deploy** ステージでは、Red Hat Advanced Cluster Security for Kubernetes は、ポリシーの条件に一致するデプロイの作成をブロックします。アドミSSIONコントローラーが適用されているクラスターでは、Kubernetes または OpenShift Container Platform サーバーがすべての非準拠のデプロイメントをブロックします。他のクラスターでは、Red Hat Advanced Cluster Security for Kubernetes が非準拠のデプロイメントを編集して、Pod がスケジュールされないようにします。
- **Runtime** ステージでは、Red Hat Advanced Cluster Security for Kubernetes はポリシーの条件に一致するすべての Pod を停止します。



## 警告

ポリシーの適用は、実行中のアプリケーションまたは開発プロセスに影響を与える可能性があります。適用オプションを有効にする前に、すべての利害関係者に通知し、自動適用アクションに対応する方法を計画してください。

## 2.2. レジストリー統合の設定

イメージをスキャンするには、ビルドパイプラインで使用しているイメージレジストリーへのアクセスを Red Hat Advanced Cluster Security for Kubernetes に提供する必要があります。

## 2.2.1. 既存のレジストリー統合を確認する

RHACS ポータルを使用して、レジストリーとすでに統合されているかどうかを確認できます。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integration** セクションで、強調表示された **Registry** タイルを探します。タイルには、そのタイルにすでに設定されているアイテムの数も表示されます。

レジストリータイルが強調表示されていない場合は、最初にイメージレジストリーと統合する必要があります。

### 2.2.1.1. 関連情報

- [イメージレジストリーとの統合](#)

## 2.3. アクセスの設定

Red Hat Advanced Cluster Security for Kubernetes は、コマンドラインインターフェイス (CLI) **roxctl** を提供して、Red Hat Advanced Cluster Security for Kubernetes ポリシーをビルドパイプラインに簡単に統合できるようにします。**roxctl** CLI は、問題に関する詳細情報とその修正方法を出力して、開発者がコンテナライフサイクルの初期ステージで高水準を維持できるようにします。

Red Hat Advanced Cluster Security for Kubernetes API サーバーに対して安全に認証するには、API トークンを作成する必要があります。

### 2.3.1. roxctl CLI を使用した認証

認証には、認証トークンまたは管理者パスワードを使用できます。Red Hat は、各トークンに特定のアクセス制御権限が割り当てられているため、実稼働環境で認証トークンを使用することが推奨されます。

次の手順を使用して、認証トークンを生成します。

### 手順

1. RHACS ポータルに移動します。
2. **Platform Configuration** → **Integrations** に移動します。
3. **Authentication Tokens** カテゴリーまで下にスクロールし、**API Token** をクリックします。
4. **Generate Token** をクリックします。
5. トークンの名前を入力し、必要なレベルのアクセスを提供するロールを選択します (たとえば、**Continuous Integration** または **Sensor Creator**)。
6. **Generate** をクリックします。



### 重要

生成されたトークンをコピーして安全に保存します。再度表示することはできません。



## 注記

認証トークンを生成したら、それを **ROX\_API\_TOKEN** 変数としてエクスポートします。

```
$ export ROX_API_TOKEN=<api_token>
```

トークンをファイルに保存して、**-token-file** オプションとともに使用することもできます。以下に例を示します。

```
$ roxctl central debug dump --token-file <token_file>
```

- **-password (-p)** オプションと **--token-file** オプションの両方を同時に使用することはできません。
- すでに **ROX\_API\_TOKEN** 変数を設定し、**--token-file** オプションを指定している場合、**roxctl** CLI は指定されたトークンファイルを認証に使用します。
- すでに **ROX\_API\_TOKEN** 変数を設定し、**--password** オプションを指定している場合、**roxctl** CLI は指定されたパスワードを認証に使用します。

## 2.3.2. バイナリーをダウンロードして roxctl CLI をインストール

**roxctl** CLI をインストールして、コマンドラインインターフェイスから Red Hat Advanced Cluster Security for Kubernetes と対話できます。**roxctl** は、Linux、Windows、または macOS にインストールできます。

### 2.3.2.1. Linux への roxctl CLI のインストール

次の手順を使用して、Linux に **roxctl** CLI バイナリーをインストールできます。

#### 手順

1. **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.70.2/bin/Linux/roxctl
```

2. **roxctl** バイナリーを実行可能にします。

```
$ chmod +x roxctl
```

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

```
$ echo $PATH
```

#### 検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```



### 2.3.2.2. macOS への roxctl CLI のインストール

次の手順を使用して、**roxctl** CLI バイナリーを macOS にインストールできます。

#### 手順

1. **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.70.2/bin/Darwin/roxctl
```

2. バイナリーからすべての拡張属性を削除します。

```
$ xattr -c roxctl
```

3. **roxctl** バイナリーを実行可能にします。

```
$ chmod +x roxctl
```

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

```
$ echo $PATH
```

#### 検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```

### 2.3.2.3. Windows への roxctl CLI のインストール

次の手順を使用して、**roxctl** CLI バイナリーを Windows にインストールできます。

#### 手順

- **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.70.2/bin/Windows/roxctl.exe
```

#### 検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```

### 2.3.3. コンテナから roxctl CLI の実行

**roxctl** クライアントは、Red Hat Advanced Cluster Security for Kubernetes **roxctl** イメージのデフォルトのエントリーポイントです。コンテナイメージで **roxctl** クライアントを実行するには、以下を行います。

## 手順

1. **registry.redhat.io** レジストリーにログインします。

```
$ docker login registry.redhat.io
```

2. **roxctl** CLI の最新のコンテナイメージをプルします。

```
$ docker pull registry.redhat.io/advanced-cluster-security/rhacs-roxctl-rhel8:3.70.2
```

CLI をインストールした後、次のコマンドを使用して CLI を実行できます。

```
$ docker run -e ROX_API_TOKEN=$ROX_API_TOKEN \  
-it registry.redhat.io/advanced-cluster-security/rhacs-roxctl-rhel8:3.70.2 \  
-e $ROX_CENTRAL_ADDRESS <command>
```

## 検証

- インストールした **roxctl** のバージョンを確認します。

```
$ docker run -it registry.redhat.io/advanced-cluster-security/rhacs-roxctl-rhel8:3.70.2 version
```

## 2.4. CI パイプラインとの統合

これらの手順を完了したら、次のステップは CI パイプラインと統合することです。

各 CI システムでは、わずかに異なる設定が必要になる場合があります。

### 2.4.1. Jenkins の使用

[StackRox Container Image Scanner](#) Jenkins プラグインを使用して、Jenkins と統合します。このプラグインは、Jenkins フリースタイルプロジェクトとパイプラインの両方で使用できます。

### 2.4.2. CircleCI の使用

Red Hat Advanced Cluster Security for Kubernetes を CircleCI と統合できます。

## 前提条件

- **Image** リソースの **read** と **write** 権限を持つトークンがある。
- Docker Hub アカウントのユーザー名とパスワードがある。

## 手順

1. CircleCI にログインして、既存のプロジェクトを開くか、新しいプロジェクトを作成します。
2. **Project Settings** をクリックします。
3. **Environment variables** をクリックします。
4. 変数の **Add variable** をクリックして、次の 3 つの環境変数を作成します。

- **Name: STACKROX\_CENTRAL\_HOST** - Central の DNS 名または IP アドレス。
  - **Name: ROX\_API\_TOKEN** - Red Hat Advanced Cluster Security for Kubernetes にアクセスするための API トークン。
  - **Name: DOCKERHUB\_PASSWORD** - Docker Hub アカウントのパスワード。
  - **Name: DOCKERHUB\_USER** - Docker Hub アカウントのユーザー名。
5. CircleCI 設定ファイルがまだない場合は、選択したプロジェクトのローカルコードリポジトリのルートディレクトリーに **.circleci** というディレクトリーを作成します。
  6. **.circleci** ディレクトリーの次の行に **config.yml** 設定ファイルを作成します。

```

version: 2
jobs:
  check-policy-compliance:
    docker:
      - image: 'circleci/node:latest'
        auth:
          username: $DOCKERHUB_USER
          password: $DOCKERHUB_PASSWORD
    steps:
      - checkout
      - run:
          name: Install roxctl
          command: |
            curl -H "Authorization: Bearer $ROX_API_TOKEN"
            https://$STACKROX_CENTRAL_HOST:443/api/cli/download/roxctl-linux -o roxctl && chmod
            +x ./roxctl
      - run:
          name: Scan images for policy deviations and vulnerabilities
          command: |
            ./roxctl image check --endpoint "$STACKROX_CENTRAL_HOST:443" --image "
            <your_registry/repo/image_name>" ❶
      - run:
          name: Scan deployment files for policy deviations
          command: |
            ./roxctl image check --endpoint "$STACKROX_CENTRAL_HOST:443" --image "
            <your_deployment_file>" ❷
            # Important note: This step assumes the YAML file you'd like to test is located in the
            project.
    workflows:
      version: 2
      build_and_test:
        jobs:
          - check-policy-compliance

```

❶ **<your\_registry/repo/image\_name>** をレジストリーとイメージパスに置き換えます。

❷ **<your\_deployment\_file>** をデプロイメントファイルへのパスに置き換えます。



### 注記

リポジトリに CircleCI の **config.yml** ファイルがすでにある場合は、既存の設定ファイルに指定された詳細を含む新しいジョブセクションを追加します。

7. 設定ファイルをリポジトリにコミットした後、CircleCI ダッシュボードの **Jobs** キューに移動して、ビルドポリシーの適用を確認します。

## 第3章 PAGERDUTY との統合

PagerDuty を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から PagerDuty にアラートを転送できます。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を PagerDuty と統合するための高レベルのワークフローを表しています。

1. PagerDuty に新しい API サービスを追加し、統合キーを取得します。
2. 統合キーを使用して、Red Hat Advanced Cluster Security for Kubernetes で通知を設定します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 3.1. PAGERDUTY の設定

新しいサービスを作成し、統合キーを取得して、PagerDuty との統合を開始します。

#### 手順

1. **Configuration** → **Services** に移動します。
2. **Add Services** を選択します。
3. **General Settings** で、**Name** と **Description** を指定します。
4. **Integration Setting** の **Integration Type** ドロップダウンメニューで **Events v2 API** を選択した状態で、**Use our API Directly** をクリックします。
5. **Incident Settings** で、**Escalation Policy** を選択し、通知設定とインシデントタイムアウトを設定します。
6. **Incident Behavior** と **Alert Grouping** のデフォルト設定を受け入れるか、必要に応じて設定します。
7. **Add Service** をクリックします。
8. **Service Details** ページで、**Integration Key** をメモします。

### 3.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

統合キーを使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまで下にスクロールして、**PagerDuty** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。

5. **PagerDuty integration key** フィールドに統合キーを入力します。
6. **Test (checkmark アイコン)** をクリックして、PagerDuty との統合が機能していることを確認します。
7. **Create (save アイコン)** をクリックして、設定を作成します。

### 3.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、PagerDuty notifier を選択します。



#### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



#### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 第4章 SLACK との統合

Slack を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から Slack にアラートを転送できます。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を Slack と統合するための高レベルのワークフローを表しています。

1. 新しい Slack アプリを作成し、受信 Webhook を有効にして、WebhookURL を取得します。
2. Webhook URL を使用して、Slack を Red Hat Advanced Cluster Security for Kubernetes と統合します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 4.1. SLACK の設定

新しい Slack アプリケーションを作成することから始めて、Webhook の URL を取得します。

#### 前提条件

1. Webhook を作成するには、管理者アカウントまたは権限を持つユーザーアカウントが必要。

#### 手順

1. 新しい Slack アプリケーションを作成します。



#### 注記

既存の Slack アプリケーションを使用する場合は、<https://api.slack.com/apps> にアクセスしてアプリケーションを選択してください。

- a. <https://api.slack.com/apps/new> に移動します。
  - b. **App Name** を入力し、**Development Slack Workspace** を選択してアプリケーションをインストールします。
  - c. **Create App** をクリックします。
2. 設定ページの **Basic Information** セクションで、**Incoming Webhooks** を選択します (**Add features and functionality** の下)。
  3. **Activate Incoming Webhooks** トグルをオンにします。
  4. **Add New Webhook to Workspace** を選択します。
  5. アプリケーションが投稿する **channel** を選択し、**Authorize** を選択します。ページが更新され、アプリケーションの設定ページに戻ります。
  6. **Webhook URLs for Your Workspace** セクションにある Webhook URL をコピーします。

詳細は、Slack のドキュメントトピック [Getting started with Incoming Webhooks](#) を参照してください。

#### 4.1.1. さまざまな Slack チャンネルにアラートを送信する

さまざまな Slack チャンネルに通知を送信して、適切なチームに直接送信されるように、Red Hat Advanced Cluster Security for Kubernetes を設定できます。

## 手順

1. 受信 Webhook を設定した後、デプロイメント YAML ファイルに次のようなアノテーションを追加します。

```
example.com/slack-webhook:  
https://hooks.slack.com/services/T00000000/B00000000/XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

2. Red Hat Advanced Cluster Security for Kubernetes を設定するときは、**Label/Annotation Key For Slack Webhook** フィールドでアノテーションキー **example.com/slack-webhook** を使用します。

設定が完了した後、デプロイメントに YAML ファイルで設定したアノテーションが含まれている場合、Red Hat Advanced Cluster Security for Kubernetes はそのアノテーションに指定した Webhook URL にアラートを送信します。それ以外の場合は、デフォルトの Webhook URL にアラートを送信します。

## 4.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

Webhook URL を使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

### 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Notifier Integrations** セクションまで下にスクロールして、**Slack** を選択します。
3. **New Integration (add アイコン)** をクリックします。
4. **Integration Name** の名前を入力します。
5. 生成された Webhook URL を **Default Slack Webhook** フィールドに入力します。
6. **Test (checkmark アイコン)** を選択して、Slack との統合が機能していることをテストします。
7. **Create (save アイコン)** を選択して、設定を作成します。

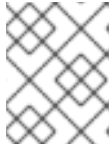
## 4.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

### 手順

1. RHACS ポータルで、**Platform Configuration → Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Slack notifier** を選択します。





### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 第5章 一般的な WEBHOOK を使用した統合

Red Hat Advanced Cluster Security for Kubernetes を使用すると、アラート通知を JSON メッセージとして任意の Webhook レシーバーに送信できます。違反が発生すると、Red Hat Advanced Cluster Security for Kubernetes は設定された URL に対して HTTP POST リクエストを行います。POST リクエストの本文には、アラートに関する JSON 形式の情報が含まれています。

次の例に示すように、Webhook POST リクエストの JSON データには、**v1.Alert** オブジェクトと設定したカスタムフィールドが含まれます。

```
{
  "alert": {
    "id": "<id>",
    "time": "<timestamp>",
    "policy": {
      "name": "<name>",
      ...
    },
    ...
  },
  ...
},
"<custom_field_1>": "<custom_value_1>"
}
```

複数の Webhook を作成できます。たとえば、すべての監査ログを受信するための1つの Webhook と、アラート通知のための別の Webhook を作成できます。

Red Hat Advanced Cluster Security for Kubernetes から任意の Webhook レシーバーにアラートを転送するには:

1. アラートを受信するための Webhook URL を設定します。
2. Webhook URL を使用して、Red Hat Advanced Cluster Security for Kubernetes で通知を設定します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 5.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

Webhook URL を使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまでスクロールダウンし、**Generic Webhook** を選択します。
3. **New integration** をクリックします。
4. **Integration name** の名前を入力します。
5. **Endpoint** フィールドに Webhook の URL を入力します。

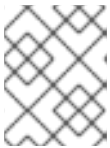
- Webhook レシーバーが信頼できない証明書を使用している場合は、**CA certificate** フィールドに CA 証明書を入力します。それ以外の場合は、空白のままにします。



### 注記

Webhook レシーバーが使用するサーバー証明書は、エンドポイント DNS 名に対して有効である必要があります。**Skip TLS verification** をクリックして、この検証を無視できます。Red Hat は、TLS 検証をオフにすることを推奨していません。TLS 検証がないと、意図しない受信者によってデータが傍受される可能性があります。

- オプション: **Enable audit logging** をクリックして、Red Hat Advanced Cluster Security for Kubernetes で行われたすべての変更に関するアラートを受信します。



### 注記

Red Hat は、アラートと監査ログに別々の Webhook を使用して、これらのメッセージを異なる方法で処理することを提案しています。

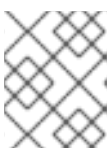
- Webhook レシーバーで認証するには、次のいずれかの詳細を入力します。
  - 基本 HTTP 認証のユーザー名とパスワード
  - カスタム ヘッダー、例: **Authorization: Bearer <access\_token>**
- Extra fields** を使用して、Red Hat Advanced Cluster Security for Kubernetes が送信する JSON オブジェクトに追加のキーと値のペアを含めます。たとえば、Webhook レシーバーが複数のソースからのオブジェクトを受け入れる場合、追加フィールドとして **"source": "rhacs"** を追加し、この値でフィルター処理して、Red Hat Advanced Cluster Security for Kubernetes からのすべてのアラートを識別できます。
- Test** を選択してテストメッセージを送信し、一般的な Webhook との統合が機能していることを確認します。
- Save** を選択して設定を作成します。

## 5.2. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

### 手順

- RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
- アラートの送信先となるポリシーを1つ以上選択します。
- Bulk actions** で **Enable notification** を選択します。
- Enable notification** ウィンドウで、**webhook notifier** を選択します。



### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

## 5. Enable をクリックします。



### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 第6章 SUMO LOGIC との統合

Sumo Logic を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から Sumo Logic にアラートを転送できます。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を Sumo Logic と統合するための高レベルのワークフローを表しています。

1. Sumo Logic に新しいカスタムアプリケーションを追加し、HTTP ソースを設定して、HTTP URL を取得します。
2. HTTP URL を使用して、Sumo Logic を Red Hat Advanced Cluster Security for Kubernetes と統合します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 6.1. SUMO LOGIC の設定

Setup Wizard を使用して、Streaming Data を設定し、HTTP URL を取得します。

#### 手順

1. Sumo Logic ホームページにログインし、**Setup Wizard** を選択します。
2. カーソルを **Set Up Streaming Data** に移動し、**Get Started** を選択します。
3. データ型の選択ページで、**Your Custom App** を選択します。
4. コレクションの設定ページで、**HTTP Source** を選択します。
5. **Source Category** の名前 (たとえば、**rhacs**) を入力し、**Continue** をクリックします。
6. 生成された URL を **Copy** します。

### 6.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

HTTP URL を使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまで下にスクロールして、**Sumo Logic** を選択します。
3. **New Integration (add アイコン)** をクリックします。
4. **Integration Name** の名前を入力します。
5. 生成された HTTP URL を **HTTP Collector Source Address** フィールドに入力します。
6. **Test (checkmark アイコン)** をクリックして、Sumo Logic との統合が機能していることをテストします。

7. **Create** (save アイコン) をクリックして、設定を作成します。

### 6.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、Sumo Logic notifier を選択します。



#### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



#### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

### 6.4. SUMO LOGIC でアラートを表示する

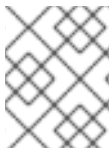
Sumo Logic で Kubernetes の Red Hat からのアラートを表示できます。

1. Sumo Logic ホームページにログインし、**Log Search** をクリックします。
2. 検索ボックスに **\_sourceCategory=rhacs** と入力します。Sumo Logic の設定時に入力したものと同一 **Source Category** 名を使用してください。
3. 時間を選択し、**Start** をクリックします。

## 第7章 GOOGLE CLOUD STORAGE との統合

Google Cloud Storage (GCS) と統合して、データのバックアップを有効にすることができます。これらのバックアップは、インフラストラクチャーの災害やデータの破損が発生した場合のデータの復元に使用できます。GCS と統合した後、毎日または毎週のバックアップをスケジュールし、手動のオンデマンドバックアップを実行できます。

バックアップには、Red Hat Advanced Cluster Security for Kubernetes データベース全体が含まれます。これには、すべての設定、リソース、イベント、および証明書が含まれます。バックアップが安全に保存されていることを確認してください。



### 注記

バージョン 3.0.53 以前の Red Hat Advanced Cluster Security for Kubernetes を使用している場合は、バックアップに証明書が含まれていません。

## 7.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

Google Cloud Storage (GCS) でデータバックアップを設定するには、Red Hat Advanced Cluster Security for Kubernetes に統合を作成します。

### 前提条件

- 既存の **bucket**。新しいバケットを作成するには、Google Cloud Storage の公式ドキュメントトピック [Creating storage buckets](#) を参照。
- 使用するストレージバケットに **Storage Object Admin** IAM ロールを持つ **service account**。Google Cloud Storage の公式ドキュメントトピック [Using Cloud IAM permissions](#) を参照。
- 使用しているサービスアカウントの **サービスアカウントキーファイル** (JSON)。GoogleCloud の公式ドキュメントトピック [Creating a service account](#) と [Creating service account keys](#) を参照。



### 注記

現在、Red Hat Advanced Cluster Security for Kubernetes は、[Workload Identity](#) を使用した GCS への認証をサポートしていない。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **External backups** セクションまで下にスクロールして、**Google Cloud Storage** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. **Backups To Retain** ボックスに、保持するバックアップの数を入力します。
6. **Schedule** で、バックアップの頻度 (毎日または毎週) とバックアッププロセスを実行する時間を選択します。

7. バックアップを保存する **Bucket** 名を入力します。
8. **Service Account JSON** フィールドに、サービスアカウントのキーファイルのコンテンツを入力します。
9. **Test** (**checkmark** アイコン) を選択して、GCS との統合が機能していることを確認します。
10. **Create** (**save** アイコン) を選択して、設定を作成します。

設定が完了すると、Red Hat Advanced Cluster Security for Kubernetes は、指定されたスケジュールに従ってすべてのデータを自動的にバックアップします。

### 7.1.1. Google Cloud Storage でオンデマンドバックアップを実行する

RHACS ポータルを使用して、Google Cloud Storage 上の Red Hat Advanced Cluster Security for Kubernetes の手動バックアップをトリガーします。

#### 前提条件

- Red Hat Advanced Cluster Security for Kubernetes を Google Cloud Storage にすでに統合している必要があります。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **External backups** セクションで、**Google Cloud Storage** をクリックします。
3. バックアップを実行する GCS バケットの統合名を選択します。
4. **Trigger Backup** をクリックします。



#### 注記

現在、**Trigger Backup** オプションを選択しても、通知はありません。ただし、Red Hat Advanced Cluster Security for Kubernetes は、バックグラウンドでバックアップタスクを開始します。

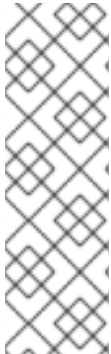
#### 7.1.1.1. 関連情報

- [Red Hat Advanced Cluster Security for Kubernetes のバックアップ](#)
- [バックアップからの復元](#)



## 第8章 SYSLOG プロトコルを使用した統合

**Syslog** は、データ保持とセキュリティ調査のために、アプリケーションが SIEM や syslog コレクターなどのセントラルの場所にメッセージを送信するために使用するイベントログプロトコルです。Red Hat Advanced Cluster Security for Kubernetes を使用すると、syslog プロトコルを使用してアラートと監査イベントを送信できます。



### 注記

- syslog プロトコルを使用してイベントを転送するには、Red Hat Advanced Cluster Security for Kubernetes バージョン 3.0.52 以降が必要です。
- syslog 統合を使用する場合、Red Hat Advanced Cluster Security for Kubernetes は、設定した違反アラートとすべての監査イベントの両方を転送します。
- 現在、Red Hat Advanced Cluster Security for Kubernetes は **CEF** (共通イベント形式) のみをサポートしています。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を syslog イベントレシーバーと統合するための高レベルのワークフローを表しています。

1. アラートを受信するように syslog イベントレシーバーを設定します。
2. レシーバーのアドレスとポート番号を使用して、Red Hat Advanced Cluster Security for Kubernetes で通知を設定します。

設定後、Red Hat Advanced Cluster Security for Kubernetes は、設定された syslog レシーバーにすべての違反と監査イベントを自動的に送信します。

### 8.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUB との SYSLOG 統合の設定

Red Hat Advanced Cluster Security for Kubernetes で新しい syslog 統合を作成します。

#### 手順

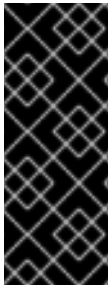
1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまでスクロールダウンし、**Syslog** を選択します。
3. **New Integration** (add アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. **Logging Facility** の値を **local0** から **local7** まで選択します。
6. **Receiver Host** アドレスと **Receiver Port** 番号を入力します。
7. TLS を使用している場合は、**Use TLS** トグルをオンにします。
8. 信頼されていない証明書を使用している syslog レシーバーの場合は、**Disable TLS Certificate Validation (Insecure)** トグルをオンにします。それ以外の場合は、このトグルをオフのままにします。

9. **Test** (**checkmark** アイコン) を選択してテストメッセージを送信し、汎用 Webhook との統合が機能していることを確認します。
10. **Create** (**save** アイコン) を選択して、設定を作成します。

## 第9章 AMAZON S3 との統合

Red Hat Advanced Cluster Security for Kubernetes を [Amazon S3](#) と統合して、データのバックアップを有効にすることができます。これらのバックアップは、インフラストラクチャーの災害やデータの破損が発生した場合のデータの復元に使用できます。Amazon S3 と統合した後、毎日または毎週のバックアップをスケジュールし、手動のオンデマンドバックアップを実行できます。

バックアップには、Red Hat Advanced Cluster Security for Kubernetes データベース全体が含まれます。これには、すべての設定、リソース、イベント、および証明書が含まれます。バックアップが安全に保存されていることを確認してください。



### 重要

- バージョン 3.0.53 以前の Red Hat Advanced Cluster Security for Kubernetes を使用している場合は、バックアップに証明書が含まれていません。
- Amazon S3 がエアギャップ環境の一部である場合は、Red Hat Advanced Cluster Security for Kubernetes で AWS ルート CA を [trusted certificate authority](#) として追加する必要があります。

### 9.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES で AMAZON S3 統合を設けます。

Amazon S3 バックアップを設定するには、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

#### 前提条件

- 既存の S3 バケット。必要な権限を持つ新しいバケットを作成するには、Amazon のドキュメントトピック [Creating a bucket](#) を参照。
- S3 バケット、**Access key ID**、および **Secret access key** の **Read**、**write**、および **delete** の権限。
- KIAM、kube2iam、または別のプロキシを使用している場合は、**read**、**write**、および **delete** の権限を持つ **IAM role**。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **External backups** セクションまで下にスクロールして、**Amazon S3** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. **Backups To Retain** ボックスに、保持するバックアップの数を入力します。
6. **Schedule** で、バックアップの頻度を日次または週次として選択し、バックアッププロセスを実行する時間を選択します。
7. バックアップを保存する **Bucket** 名を入力します。

8. バックアップを特定のフォルダー構造に保存する場合は、必要に応じて **Object Prefix** を入力します。詳細については、Amazon のドキュメントトピック [Working with object metadata](#) を参照してください。
9. 非公開の S3 インスタンスを使用している場合は、バケットの **Endpoint** を入力します。それ以外の場合は、空白のままにします。
10. バケットの **Region** を入力します。
11. **Use Container IAM Role** トグルをオンにするか、**Access Key ID** と **Secret Access Key** を入力します。
12. **Test (checkmark アイコン)** を選択して、AmazonS3 との統合が機能していることを確認します。
13. **Create (save アイコン)** を選択して、設定を作成します。

設定が完了すると、Red Hat Advanced Cluster Security for Kubernetes は、指定されたスケジュールに従ってすべてのデータを自動的にバックアップします。

## 9.2. AMAZON S3 でのオンデマンドバックアップの実行

RHACS ポータルを使用して、Amazon S3 上の Red Hat Advanced Cluster Security for Kubernetes の手動バックアップをトリガーします。

### 前提条件

- Red Hat Advanced Cluster Security for Kubernetes を Amazon S3 にすでに統合している必要があります。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **External backups** セクションで、**AmazonS3** をクリックします。
3. バックアップを実行する S3 バケットの統合名を選択します。
4. **Trigger Backup** をクリックします。



### 注記

現在、**Trigger Backup** オプションを選択しても、通知はありません。ただし、Red Hat Advanced Cluster Security for Kubernetes は、バックグラウンドでバックアップタスクを開始します。

## 9.3. 関連情報

- [Red Hat Advanced Cluster Security for Kubernetes のバックアップ](#)
- [バックアップからの復元](#)

## 第10章 GOOGLE CLOUD SECURITY コマンドセンターとの統合

[Google Cloud Security Command Center](#) (Cloud SCC) を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から Cloud SCC にアラートを転送できます。このガイドでは、Red Hat Advanced Cluster Security for Kubernetes を Cloud SCC と統合する方法について説明します。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を Cloud SCC と統合するための高レベルのワークフローを表しています。

1. 新しいセキュリティーソースを Google Cloud に登録します。
2. Red Hat Advanced Cluster Security for Kubernetes にソース ID とサービスアカウントキーを提供します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 10.1. GOOGLECLOUD SCC の設定

まず、信頼できるクラウド SCC ソースとして Red Hat Advanced Cluster Security for Kubernetes を追加します。

#### 手順

1. [Adding vulnerability and threat sources to Cloud Security Command Center](#) ガイドに従い、Red Hat Advanced Cluster Security for Kubernetes を信頼できるクラウド SCC ソースとして追加します。Red Hat Advanced Cluster Security for Kubernetes の統合のために Google Cloud が作成する **Source ID** をメモしておきます。登録後にソース ID が表示されない場合は、[Cloud SCC Security Sources page](#) で確認できます。
2. 前の手順で作成したサービスアカウント、または使用した既存のアカウントのキーを作成します。詳細については [creating and managing service account keys](#) Google Cloud のガイドを参照してください。

### 10.2. GOOGLE CLOUD SCC と統合するための RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

**Source ID** と **service account key** を使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい Google Cloud SCC 統合を作成します。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまで下にスクロールして、**Google Cloud SCC** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. **Cloud SCC Source ID** と **Service Account Key (JSON)** を入力します。
6. **Create** (**save** アイコン) を選択して、設定を作成します。

## 10.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Google Cloud SCC notifier** を選択します。



### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 第11章 SPLUNK との統合

Splunk を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から Splunk にアラートを転送し、Splunk 内から脆弱性とコンプライアンスに関連するデータを表示できます。

ユースケースに応じて、次の方法を使用して、Red Hat Advanced Cluster Security for Kubernetes を Splunk と統合できます。

- Splunk で、[using an HTTP event collector](#) による
  - イベントコレクターオプションを使用して、アラートと監査ログデータを転送します
- [using the StackRox Kubernetes Security Platform add-on](#) による
  - アドオンを使用して、脆弱性の検出とコンプライアンスのデータを Splunk に取り込みます



### 注記

StackRox Kubernetes Security Platform アドオンは、Red Hat Advanced Cluster Security for Kubernetes バージョン 3.0.51.0 以降を使用している場合にのみ使用できます。

これらの統合オプションの一方または両方を使用して、Red Hat Advanced Cluster Security for Kubernetes を Splunk と統合できます。

### 11.1. HTTP イベントコレクターの使用

HTTP イベントコレクターを使用して、Red Hat Advanced Cluster Security for Kubernetes から Splunk にアラートを転送できます。

HTTP イベントコレクターを使用して Red Hat Cluster Security for Kubernetes を Splunk と統合するには、次の手順に従います。

1. Splunk に新しい HTTP イベントコレクターを追加し、トークン値を取得します。
2. トークン値を使用して、Red Hat Advanced Cluster Security for Kubernetes で通知を設定します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

#### 11.1.1. Splunk に HTTP イベントコレクターを追加する

Splunk インスタンスの新しい HTTP イベントコレクターを追加し、トークンを取得します。

##### 手順

1. Splunk ダッシュボードで、**Settings** → **Add Data** に移動します。
2. **Monitor** をクリックします。
3. **Add Data** ページで、**HTTP Event Collector** をクリックします。
4. イベントコレクターの **Name** を入力し、**Next** > をクリックします。
5. デフォルトの **Input Settings** を受け入れて、**Review** > をクリックします。

6. イベントコレクターのプロパティを確認し、**Submit** をクリックします。
7. イベントコレクターの **Token Value** をコピーします。このトークン値は、Red Hat Advanced Cluster Security for Kubernetes で Splunk との統合を設定するために必要です。

#### 11.1.1.1. HTTP イベントコレクターの有効化

イベントを受信する前に、HTTP イベントコレクタートークンを有効にする必要があります。

##### 手順

1. Splunk ダッシュボードで、**Settings** → **Data inputs** に移動します。
2. **HTTP Event Collector** をクリックします。
3. **Global Settings** をクリックします。
4. 開いたダイアログで、**Enabled** をクリックし、**Save** をクリックします。

#### 11.1.2. Red Hat Advanced Cluster Security for Kubernetes での Splunk 統合の設定

トークン値を使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい Splunk 統合を作成します。

##### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまでスクロールダウンし、**Splunk** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. SplunkURL を **HTTP Event Collector URL** フィールドに入力します。HTTPS の場合は **443**、HTTP の場合は **80** でない場合は、ポート番号を指定する必要があります。また、URL の最後に URL パス **/services/collector/event** を追加する必要があります。たとえば、**https://<splunk-server-path>:8088/services/collector/event** です。
6. **HTTP Event Collector Token** フィールドにトークンを入力します。



##### 注記

Red Hat Advanced Cluster Security for Kubernetes バージョン 3.0.57 以降を使用している場合は **Source Type for Alert** イベントと **Source Type for Audit** イベントのソースタイプを指定できます。

7. **Test** (**checkmark** アイコン) を選択してテストメッセージを送信し、Splunk との統合が機能していることを確認します。
8. **Create** (**save** アイコン) を選択して、設定を作成します。

#### 11.1.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。



## 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Splunk notifier** を選択します。



### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 11.2. STACKROX KUBERNETES SECURITY PLATFORM アドオンの使用

StackRox Kubernetes Security Platform アドオンを使用して、脆弱性の検出とコンプライアンスに関連するデータを Red Hat Advanced Cluster Security for Kubernetes から Splunk に転送できます。

まず、Red Hat Advanced Cluster Security for Kubernetes のすべてのリソースに対する読み取り権限を持つ API トークンを生成し、そのトークンを使用してアドオンをインストールおよび設定します。

### 11.2.1. Splunk アドオンのインストールと設定

StackRox Kubernetes Security Platform アドオンを Splunk インスタンスからインストールできます。

#### 前提条件

- Red Hat Advanced Cluster Security for Kubernetes のすべてのリソースに対する **read** 権限を持つ API トークンがある。**Analyst** システムロールを割り当てて、このレベルのアクセスを許可できます。**Analyst** ロールには、すべてのリソースに対する read 権限がある。

## 手順

1. StackRox Kubernetes Security Platform アドオンを [Splunkbase](#) からダウンロードします。
2. Splunk インスタンスの Splunk ホームページに移動します。
3. **Apps** → **Manage Apps** に移動します。
4. **Install app from file** を選択します。
5. **Upload app** ポップアップボックスで、**Choose File** を選択し、StackRox Kubernetes Security Platform アドオンファイルを選択します。
6. **アップロード** をクリックします。
7. **Restart Splunk** をクリックし、再起動することを確認します。
8. Splunk が再起動したら、**Apps** メニューから **StackRox** を選択します。
9. **Create New Input** をクリックします。
10. **StackRox Compliance** を選択してコンプライアンスデータをプルするか、**StackRox Vulnerability Management** を選択して脆弱性データを Splunk にプルします。
11. 入力の **Name** を入力します。
12. Red Hat Advanced Cluster Security for Kubernetes からデータをプルする **Interval** を選択します。たとえば、14400 秒ごと。
13. データの送信先となる Splunk **Index** を選択します。
14. **Central Endpoint** には、セントラルインスタンスの IP アドレスまたは名前を入力します。
15. アドオン用に生成した **API token** を入力します。
16. **Add** をクリックします。

## 第12章 イメージ脆弱性スキャナーとの統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) は、さまざまな脆弱性スキャナーと統合して、コンテナイメージをインポートし、脆弱性を監視できるようにします。Red Hat Advanced Cluster Security for Kubernetes を設定して、次のような多くのオープンソースおよび商用コンテナのイメージ脆弱性スキャナーからイメージの脆弱性データを取得できます。

- [アンカー](#)
- [Clair](#)
- [Docker トラステッドレジストリー](#)
- [Google コンテナ分析](#)
- [Red Hat Quay](#)
- [Tenable.io](#)

DevOps ワークフローでこれらの製品のいずれかを使用している場合は、RHACS ポータルを使用して、Red Hat Advanced Cluster Security for Kubernetes との接続を設定できます。統合されると、イメージの脆弱性は RHACS ポータルに表示され、簡単にトリアージできます。

### 12.1. ANCHORE スキャナーとの統合

Red Hat Advanced Cluster Security for Kubernetes を Anchore スキャナーと統合できます。

#### 前提条件

- 認証用のアンカースキャナーのユーザー名とパスワードがある。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Anchore Scanner** を選択します。 **Configure Anchore scanner image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name**: 統合の名前。
  - b. **Types**: **Scanner** を選択します。
  - c. **Endpoint**: スキャナーのアドレス。
  - d. **Username**と**Password**
5. **Test** (**checkmark** アイコン) を選択して、統合が機能していることをテストします。
6. **Create** (**save** アイコン) を選択して、設定を作成します。

### 12.2. CLAIR との統合

イメージの脆弱性を静的に分析するために、Red Hat Advanced Cluster Security for Kubernetes を Clair と統合できます。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Clair** を選択します。 **Configure Anchore scanner image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name**: 統合の名前。
  - b. **Types**: **Scanner** を選択します。
  - c. **Endpoint**: スキャナーのアドレス。
5. **Test** (**checkmark** アイコン) を選択して、統合が機能していることをテストします。
6. **Create** (**save** アイコン) を選択して、設定を作成します。

### 12.3. DOCKER TRUSTED REGISTRY スキャナーとの統合

イメージをスキャンするために、Red Hat Advanced Cluster Security for Kubernetes を Docker Trusted Registry と統合できます。

#### 前提条件

- Docker Trusted Registry で認証するためのユーザー名とパスワードが必要です。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Docker Trusted Registry** を選択します。 **Configure image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name**: 統合の名前。
  - b. **Types**: **Scanner** を選択します。
  - c. **Endpoint**: レジストリーのアドレス。
  - d. **Username**と**Password**
5. **Test** (**checkmark** アイコン) を選択して、選択したレジストリーとの統合が機能していることをテストします。
6. **Create** (**save** アイコン) を選択して、設定を作成します。

## 12.4. GOOGLE CONTAINER REGISTRY との統合

コンテナ分析と脆弱性スキャンのために、Red Hat Advanced Cluster Security for Kubernetes を Google Container Registry (GCR) と統合できます。

### 前提条件

- Google Container Registry のサービスアカウントキーが必要。
- 関連付けられたサービスアカウントは、レジストリーにアクセスできる。ユーザーおよび他のプロジェクトに GCR へのアクセスを許可する方法は、[Configuring access control](#) を参照してください。
- [GCR Container Analysis](#) を使用している場合は、サービスアカウントに次のロールを付与しています。
  - コンテナ分析ノートビューアー
  - コンテナ分析発生状況ビューアー
  - ストレージオブジェクトビューアー

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Image Integrations** セクションで、**Google Container Registry** を選択します。**Configure image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name**: 統合の名前。
  - b. **Types**: **Scanner** を選択します。
  - c. **Registry Endpoint**: レジストリーのアドレス。
  - d. **Project**: Google Cloud プロジェクト名。
  - e. **Service Account Key (JSON)**: 認証用のサービスアカウントキー。
5. **Test** (**checkmark** アイコン) を選択して、選択したレジストリーとの統合が機能していることをテストします。
6. **Create** (**save** アイコン) を選択して、設定を作成します。

## 12.5. QUAY CONTAINER REGISTRY との統合

イメージをスキャンするために、Red Hat Advanced Cluster Security for Kubernetes を Quay Container Registry と統合できます。

### 前提条件

- Quay Container Registry で認証するには OAuth トークンが必要。

## 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで、**Quay Container Registry** を選択します。**Configure image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下に必要な詳細を入力します。
  - a. **Integration Name:** 統合の名前。
  - b. **Types: Scanner** を選択します。
  - c. **Endpoint:** レジストリーのアドレス。
  - d. **OAuth Token:** Quay Container Registry の OAuth トークン。
5. **Test (checkmark アイコン)** を選択して、選択したレジストリーとの統合が機能していることをテストします。
6. **Create (save アイコン)** を選択して、設定を作成します。

## 12.6. TENABLE との統合

イメージをスキャンするために、Red Hat Advanced Cluster Security for Kubernetes を Tenable と統合できます。

### 前提条件

- 認証用の Tenable アクセスキーと秘密キーが必要。

### 手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Image Integrations** セクションで、**Tenable** を選択します。**Configure Anchore scanner image integration** ボックスが開きます。
3. **New Integration** をクリックします。
4. 以下のフィールドに詳細を記入します。
  - a. **Integration Name:** 統合の名前。
  - b. **Source Inputs: Scanner** を選択します。
  - c. **Access Key:** Tenable のアクセスキー。
  - d. **Secret Key:** Tenable のシークレットキー。
5. **Test (checkmark アイコン)** を選択して、統合が機能していることをテストします。
6. **Create (save アイコン)** を選択して、設定を作成します。

## 第13章 JIRA との統合

Jira を使用している場合は、Red Hat Advanced Cluster Security for Kubernetes から Jira にアラートを転送できます。

次の手順は、Red Hat Advanced Cluster Security for Kubernetes を Jira と統合するための高レベルのワークフローを表しています。

1. Jira でユーザーを設定します。
2. Jira の URL、ユーザー名、パスワードを使用して、Jira を Red Hat Advanced Cluster Security for Kubernetes と統合します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

### 13.1. JIRA の設定

新しいユーザーを作成することから始め、適切なロールと権限を割り当てます。

#### 前提条件

- 統合するプロジェクトで課題を作成および編集するための権限を持つ Jira アカウントが必要。

#### 手順

- 問題を作成するプロジェクトにアクセスできるユーザーを Jira に作成します。
  - 新しいユーザーを作成するには、Jira ドキュメントトピックの [Create, edit, or remove a user](#) を参照してください。
  - ユーザーにプロジェクトのロールとアプリケーションへのアクセスを許可するには、Jira ドキュメントトピックの [Assign users to groups, project roles, and applications](#) を参照してください。



#### 注記

Jira Software Cloud を使用している場合は、ユーザーを作成した後、ユーザーのトークンを作成する必要があります。

1. <https://id.atlassian.com/manage/api-tokens> にアクセスして、新しいトークンを生成します。
2. Kubernetes の Red Hat Cluster Security を設定するときは、トークンをパスワードとして使用してください。

### 13.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES の設定

Jira サーバーの URL とユーザーのクレデンシャルを使用して、Red Hat Advanced Cluster Security for Kubernetes に新しい統合を作成します。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。

2. **Notifier Integrations** セクションまでスクロールダウンし、**Jira Software** を選択します。
3. **New Integration** をクリックします。
4. **Integration Name** の名前を入力します。
5. **Username** ボックスと **Password or API Token** ボックスにユーザーのクレデンシャルを入力します。
6. **Issue Type** に、有効な **Jira Issue Type** (Task、Sub-task、Bug など) を入力します。
7. **Jira URL** ボックスに Jira サーバーの URL を入力します。
8. 問題を作成するプロジェクトのキーを **Default Project** ボックスに入力します。
9. **Annotation Key For Project** ボックスを使用して、さまざまな Jira プロジェクトで課題を作成します。
10. Jira プロジェクトでカスタムプライオリティーを使用する場合は、**Priority Mapping** トグルを使用してカスタムプライオリティーを設定します。
11. JIRA プロジェクトで必須のカスタムフィールドを使用する場合は、**Default Fields JSON (Necessary If Required Fields)** ボックスに JSON 値として入力します。以下に例を示します。

```
{
  "customfield_10004": 3,
  "customfield_20005": "Alerts",
}
```

12. **Test** (checkmark アイコン) を選択して、Jira との統合が機能していることをテストします。
13. **Create** (save アイコン) を選択して、設定を作成します。

### 13.2.1. さまざまな Jira プロジェクトで問題を作成する

Red Hat Advanced Cluster Security for Kubernetes を設定して、さまざまな Jira プロジェクトで問題を作成し、適切なチームに直接移動できるようにすることができます。

#### 前提条件

- アラートを送信する各プロジェクトにアクセスできるアカウントが必要。

#### 手順

1. デプロイ YAML ファイルに次のようなアノテーションを追加します。

```
jira/project-key: <jira_project_key>
```

2. Red Hat Advanced Cluster Security for Kubernetes Cluster Security を設定するときには、**Annotation Key For Project** フィールドでアノテーションキー **jira/project-key** を使用します。

設定が完了した後、デプロイメントの YAML ファイルにアノテーションが含まれている場合、Red Hat Advanced Cluster Security for Kubernetes はそのアノテーションに指定されたプロジェクトにアラートを送信します。それ以外の場合は、アラートがデフォルトのプロジェクトに送信されます。



### 13.2.2. Jira でのカスタムプライオリティーの設定

Jira プロジェクトでカスタムプライオリティーを使用している場合は、Red Hat Advanced Cluster Security for Kubernetes でそれらを設定できます。

#### 手順

1. Red Hat Advanced Cluster Security for Kubernetes で Jira 統合を設定しているときに、**Priority Mapping** トグルをオンにします。Red Hat Advanced Cluster Security for Kubernetes は JIRA プロジェクトスキーマを取得し、**CRITICAL\_SEVERITY**、**HIGH\_SEVERITY**、**MEDIUM\_SEVERITY**、および **LOW\_SEVERITY** フィールドの値を自動入力します。
2. JIRA プロジェクト設定に基づいて、プライオリティーの値を確認または更新します。
3. **Test** (**checkmark** アイコン) を選択して、Jira との統合が機能していることをテストします。
4. **Create** (**save** アイコン) を選択して、設定を作成します。



#### 注記

エラーが発生した場合は、[Troubleshooting Jira integration](#) セクションの手順に従います。

### 13.3. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

#### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Jira notifier** を選択します。



#### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



## 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。

## 13.4. JIRA 統合のトラブルシューティング

Jira プロジェクトでカスタムプライオリティーまたは必須のカスタムフィールドを使用している場合、Red Hat Advanced Cluster Security for Kubernetes を Jira Software と統合しようとするとうエラーが発生する可能性があります。このエラーは、重大度と優先度フィールドの値が一致していないことが原因である可能性があります。

JIRA プロジェクトのカスタムプライオリティーの値がわからない場合は、**roxctl** CLI を使用して JIRA 統合のデバッグログを有効にします。

### 手順

1. JIRA プロジェクトからカスタムプライオリティーの値を取得するには、次のコマンドを実行して、JIRA 統合のデバッグログをオンにします。

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central debug log --level Debug --modules
notifiers/jira
```

2. 指示に従って、Jira 統合用の Red Hat Advanced Cluster Security for Kubernetes を設定します。統合をテストすると、統合テストが失敗した場合でも、生成されたログには JIRA プロジェクトスキーマとカスタムプライオリティーが含まれます。
3. デバッグ情報を圧縮された **.zip** ファイルとして保存するには、次のコマンドを実行します。

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central debug dump
```

4. **.zip** ファイルを解凍して、JIRA プロジェクトで使用されているカスタムプライオリティーの値を取得します。
5. デバッグログをオフにするには、次のコマンドを実行します。

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central debug log --level Info
```

6. Jira 統合のために Red Hat Advanced Cluster Security for Kubernetes を再度設定し、プライオリティーの値を使用してカスタムプライオリティーを設定します。

## 第14章 メールとの統合

標準の電子メールプロバイダーにアラートを送信するように、Red Hat Advanced Cluster Security for Kubernetes (RHACS) を設定します。

RHACS から標準の電子メールプロバイダーにアラートを転送することにより、通知方法として電子メールを使用できます。RHACS プラットフォームから電子メールアドレスに警告を転送するには、**Default Recipients** または **Deployment Annotations** パラメーターのいずれかを使用します。

デフォルトの受信者を使用して、標準の集中型チームに電子メールを送信できます。デプロイメントを管理する特定のチームに電子メールを送信するには、**Deployment Annotations** を使用します。namespace でオーディエンスを管理するには、**Namespace Annotations** を使用します。

アノテーションキーを使用して、デプロイメントまたは namespace に関連付けられているポリシー違反について通知するオーディエンスを定義できます。デプロイメントにアノテーションがある場合、アノテーションの値がデフォルト値をオーバーライドします。namespace にアノテーションがある場合、namespace の値がデフォルト値を上書きします。

- デプロイメントにアノテーションキーと定義されたオーディエンスがある場合、キーによって定義されたオーディエンスに電子メールが送信されます。
- デプロイメントにアノテーションキーがない場合、namespace でアノテーションキーがチェックされ、定義されたオーディエンスに電子メールが送信されます。
- アノテーションキーが存在しない場合、統合で定義されているデフォルトの受信者に電子メールが送信されます。

### 14.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES プラットフォームでのメールプラグインの設定

電子メール通知機能を使用して、Red Hat Advanced Cluster Security for Kubernetes プラットフォームに新しい統合を作成できます。

アノテーションを使用して電子メールの受信者を動的に判別するには:

1. デプロイメント YAML ファイルに次の例のようなアノテーションを追加します。ここで、電子メールは、電子メール統合で指定する **Annotation** キーです。

```
annotations:
  email: <email_address>
```

2. RHACS を設定するときは、**Annotation key for recipient** フィールドでアノテーションキーの **email** を使用します。



#### 注記

アノテーションは、デプロイメントまたは namespace に付けることができます。

アノテーションを使用してデプロイメントまたは namespace を設定した場合、RHACS プラットフォームはアノテーションで指定された電子メールにアラートを送信します。それ以外の場合は、デフォルトの受信者にアラートを送信します。

#### 手順

1. **Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションで、**Email** を選択します。
3. **New Integration** アイコンを選択します。
4. **Integration Name** フィールドに、電子メール統合の名前を入力します。
5. **Email Server** に、電子メールサーバーのアドレスを入力します。電子メールサーバーアドレスは、完全修飾ドメイン名とポート番号です。例は **smtp.example.com:465** です。
6. 認証に使用するサービスアカウントのユーザー名とパスワードを入力します。
7. オプション: デフォルト値と異なる場合は、送信者の名前を指定します。
8. **Default Recipient** ボックスで通知を受信する電子メールアドレスを指定するか、**Annotation key for recipient** にアノテーションキーを入力します。
9. オプション: TLS なしで電子メールを送信するには、**Disable TLS certificate validation (insecure)** トグルをオンにします。StartTLS を使用していない限り、TLS を無効にしないでください。



### 注記

電子メール通知には TLS を使用します。TLS がないと、すべての電子メールは暗号化されずに送信されます。

10. オプション: StartTLS を使用するには、**Use STARTTLS (Requires TLS To Be Disabled)** ドロップダウンメニューで **Login** または **Plain** を選択します。



### 重要

StartTLS を使用すると、セッションの暗号化が確立される前に、クレデンシャルがプレーンテキストで電子メールサーバーに渡されます。

- **login** パラメーターを指定した StartTLS は、**base64** でエンコードされた文字列で認証クレデンシャルを送信します。
- **plain** パラメーターを指定した StartTLS は、認証クレデンシャルをプレーンテキストでメールリレーに送信します。

## 14.2. ポリシー通知の設定

システムポリシーのアラート通知を有効にします。

### 手順

1. RHACS ポータルで、**Platform Configuration** → **Policies** に移動します。
2. アラートの送信先となるポリシーを1つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、電子メール notifier を選択します。



### 注記

他の統合を設定していない場合、システムは通知機能が設定されていないメッセージが表示します。

5. **Enable** をクリックします。



### 注記

- Red Hat Advanced Cluster Security for Kubernetes は、オプトインベースで通知を送信します。通知を受信するには、最初に通知機能をポリシーに割り当てる必要があります。
- 通知は、特定のアラートに対して1回だけ送信されます。ポリシーに通知機能を割り当てた場合、違反によって新しいアラートが生成されない限り、通知は受信されません。
- Red Hat Advanced Cluster Security for Kubernetes は、次のシナリオに対して新しいアラートを作成します。
  - ポリシー違反は、デプロイメントで初めて発生します。
  - ランタイムフェーズのポリシー違反は、そのデプロイメントのポリシーに対する以前のランタイムアラートを解決した後のデプロイメントで発生します。