Red Hat Advanced Cluster Security for Kubernetes 4.5

統合

Red Hat Advanced Cluster Security for Kubernetes の統合

Last Updated: 2024-09-16

Red Hat Advanced Cluster Security for Kubernetes 4.5 統合

Red Hat Advanced Cluster Security for Kubernetes の統合

法律上の通知

概要

このドキュメントでは、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を説明します。

目次

第1章 イメージレジストリーとの統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) 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。

RHACS をレジストリーと統合した後、デプロイメント前またはデプロイメント後にイメージをスキャンし、イメージコンポーネントを表示し、セキュリティーポリシーをイメージに適用できます。

注記

イメージレジストリーと統合すると、RHACS 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。RHACS 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

RHACS は、次のような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
- Google Container Registry (GCR)
- Google Artifact Registry
- IBM Cloud Container Registry (ICR)
- JFrog Artifactory
- Microsoft Azure Container Registry (ACR)
- Red Hat Quay
- Red Hat コンテナーレジストリー
- Sonatype Nexus
- 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

- Google Container Registry (GCR) を使用する場合、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。
- Red Hat Advanced Cluster Security Cloud Service を使用する場合、自動設定は利用できないため、レジストリー統合を手動で作成する必要があります。

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、Instance Identity and Access Management (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ー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。

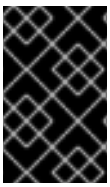
手順

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 プロキシー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 でリポジトリー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。この値は、Amazon ECR にプライベート仮想プライベートクラウド (VPC) エンドポイントを使用する場合にのみ必要です。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 assumerole 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して、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るようにすることができます。**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. コンテナー 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
```

2. EKS クラスターの IAM ロールを作成し ます。

3. 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>
```

4. Central を再起動して、変更を適用します。

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

5. 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>" (1)
      }
   ]
}
```

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

6. 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>" (1)
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

(1) **<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. コンテナー IAM を使用せずに AssumeRole を設定する

コンテナー 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ーを選択できます。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>" 1
           }
        ]
     }
     ```

     1 **<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 の手動設定

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

前提条件

- 認証用の Workload Identity またはサービスアカウントキーがある。
- 関連付けられたサービスアカウントで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. **Use workload identity**: Workload Identity を使用して認証する場合は、チェックボックスをオンにします。
    f. **Service account key (JSON)** 認証用の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 レジストリーの手動設定

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

前提条件

- 認証用の Workload Identity またはサービスアカウントキーがある。
- 関連付けられたサービスアカウントに、**Artifact Registry Reader** アイデンティティーおよびアクセス管理 (IAM) ロールの **role/artifactregistry.reader** 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、**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. **Use workload identity**: Workload Identity を使用して認証する場合は、チェックボックスをオンにします。
    e. **Service account key (JSON)** 認証用の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 の手動設定

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 の手動設定

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. Quay Container Registry を手動で設定する

Red Hat Advanced Cluster Security for Kubernetes (RHACS) を Quay Container Registry と統合できます。次の方法を使用して、Quay と統合できます。

- Quay パブリックリポジトリー (レジストリー) との統合: この方法では認証は必要ありません。
- ロボットアカウントを使用した Quay プライベートレジストリーとの統合: この方法では、Quay で使用するロボットアカウントを作成する必要があります (推奨)。詳細は、Quay documentation を参照してください。
- Quay と統合して RHACS Scanner ではなく Quay スキャナーを使用する: この方法では API を使用し、認証に OAuth トークンが必要です。"Additional Resources" の "Integrating with Quay Container Registry to scan images" を参照してください。

前提条件

- Quay プライベートレジストリーでの認証には、ロボットアカウントまたは OAuth トークン (非推奨) に関連付けられた資格情報が必要。

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. **Integration name** を入力します。
5. **Endpoint** またはレジストリーのアドレスを入力します。

a. Quay パブリックリポジトリーと統合する場合は、**Type** で **Registry** を選択し、次の手順に進みます。

b. Quay プライベートレジストリーと統合する場合は、**Type** で **Registry** を選択し、次のフィールドに情報を入力します。

- **Robot username**: Quay ロボットアカウントを使用してレジストリーにアクセスしている場合は、ユーザー名を **<namespace>+<accountname>** の形式で入力します。

- **Robot password**: Quay ロボットアカウントを使用してレジストリーにアクセスしている場合は、ロボットアカウントのユーザー名のパスワードを入力します。
- **OAuth token**: OAuth トークン (非推奨) を使用してレジストリーにアクセスしている場合は、このフィールドに入力します。

6. オプション: 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。
7. オプション: テストを行わずにインテグレーションを作成するには、**Create integration without testing** を選択します。
8. **Save** を選択します。

注記

Quay インテグレーションを編集しているが資格情報を更新したくない場合は、**Update stored credentials** が選択されていないことを確認します。

1.4. 関連情報

- イメージをスキャンするための Quay Container Registry との統合

1.4.1. IBM Cloud Container Registry の手動設定

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。

1.4.2. Red Hat Container Registry の手動設定

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) 製品と統合されます。イメージをデプロイする前に、RHACS を使用して、ビルド時およびデプロイ時のセキュリティールールをイメージに適用できます。

イメージが構築され、レジストリーにプッシュされた後、RHACS は 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、RHACS 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します。

CI 製品との統合は、RHACS インストールに接続して、イメージが設定したビルド時ポリシーに準拠しているかどうかを確認することで機能します。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。ビルド時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. ビルドポリシーの設定
2. レジストリー統合の設定
3. RHACS 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 RHACS ポリシーを確認できます。

手順

1. コンテナーライフサイクルのビルド時に適用されるポリシーを設定します。
2. 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します。

関連情報

イメージレジストリーとの統合

2.1.1. 既存のビルド時ポリシーの確認

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時ポリシーを確認します。

手順

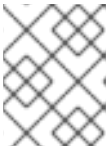
1. RHACS ポータルで、**Platform Configuration → Policy Management** に移動します。
2. グローバル検索を使用して、**Lifecycle Stage:Build** を検索します。

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 → Policy Management** 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。

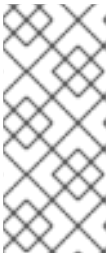
注記

アラート通知を受信するには、RHACS 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。通知プロバイダーを RHACS に統合している場合にのみ、通知が表示されます。

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ー (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、RHACS は CI ビルドを失敗します。
- **Deploy** 段階では、RHACS アドミッションコントローラーが設定され実行されている場合、RHACS はポリシーの条件に一致するデプロイメントの作成と更新をブロックします。
  - アドミッション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、RHACS は準拠していない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。
  - 既存のデプロイメントの場合、ポリシーの変更は、Kubernetes イベントが発生したときに、基準が次に検出されたときにのみ適用されます。適用の詳細は、「デプロイステージのセキュリティーポリシーの適用」を参照してください。
- **Runtime** ステージでは、RHACS 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。

2.1.2.1. デプロイステージのセキュリティーポリシーの実施

Red Hat Advanced Cluster Security for Kubernetes は、デプロイ時のポリシーに対して、アドミッションコントローラーによるハードな適用と RHACS センサーによるソフトな適用という 2 つの形式のセキュリティーポリシー適用をサポートしています。アドミッションコントローラーは、ポリシーに違反するデプロイメントの作成または更新をブロックします。アドミッションコントローラーが無効または使用できない場合、Sensor はポリシーに違反するデプロイメントのレプリカを **0** にスケールダウンして強制を実行できます。

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。

2.1.2.1.1. ハードエンフォースメント

ハードエンフォースメントは、RHACS アドミッションコントローラーによって実行されます。アドミッション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。アドミッションコントローラーは、**CREATE** および **UPDATE** 操作をブロックします。デプロイ時の強制が有効に設定されたポリシーを満たす Pod の作成または更新リクエストはすべて失敗します。

注記

Kubernetes アドミッション Webhook は、**CREATE**、**UPDATE**、**DELETE**、または **CONNECT** 操作のみをサポートします。RHACS アドミッションコントローラーは、**CREATE** および **UPDATE** 操作のみをサポートします。**kubectl patch**、**kubectl set**、**kubectl scale** などの操作は、UPDATE 操作ではなく、PATCH 操作です。Kubernetes では PATCH 操作がサポートされていないため、RHACS は PATCH 操作の強制を実行できません。

ブロックを強制するには、RHACS でクラスターに対して次の設定を有効にする必要があります。

- オブジェクト作成時に強制: **Dynamic Configuration** セクションのこのトグルは、アドミッションコントロールサービスの動作を制御します。これを機能させるには、**Static Configuration** セクションの **Configure Admission Controller Webhook to listen on Object Creates**トグルをオンにする必要があります。

- オブジェクトの更新時に強制: **Dynamic Configuration** セクションのこのトグルは、アドミッションコントロールサービスの動作を制御します。これを機能させるには、**Static Configuration** セクションの **Configure Admission Controller Webhook to listen on Object Updates** トグルをオンにする必要があります。

Static Configuration 設定の項目を変更した場合に、その変更を有効にするにはセキュアクラスターを再デプロイする必要があります。

2.1.2.1.2. ソフトエンフォースメント

ソフトエンフォースメントは RHACS センサーによって実行されます。このエンフォースメントにより、操作が開始しなくなります。ソフトエンフォースメントでは、Sensor はレプリカを 0 にスケーリングし、Pod がスケジュールされるのを防ぎます。このエンフォースメントでは、クラスター内で準備ができていないデプロイメントが使用可能になります。

ソフトエンフォースメントが設定されていて、Sensor がダウンしていると、RHACS はエンフォースメントを実行できません。

2.1.2.1.3. namespace の除外

デフォルトでは、RHACS は、**stackrox**、**kube-system**、**istio-system** namespace などの特定の管理 namespace をエンフォースメントブロックから除外します。その理由は、RHACS が正しく機能するためには、これらの namespace 内の一部の項目をデプロイする必要があるためです。

2.1.2.1.4. 既存のデプロイメントへのエンフォースメント

既存のデプロイメントの場合、ポリシーの変更は、Kubernetes イベントが発生したときに、基準が次に検出されたときにのみ適用されます。ポリシーに変更を加えた場合は、**Policy Management** を選択し、**Reassess All** をクリックしてポリシーを再評価する必要があります。このアクションは、新しい受信 Kubernetes イベントがあるかどうかに関係なく、すべての既存のデプロイメントにデプロイポリシーを適用します。ポリシーに違反があった場合は、RHACS がエンフォースメントを実行します。

関連情報

- アドミッションコントローラーの適用の使用

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

RHACS は、RHACS 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、**roxctl** コマンドラインインターフェイス (CLI) を提供します。**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、開発者がコンテナー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. API トークンのエクスポートと保存

手順

1. 認証トークンを生成したら、次のコマンドを入力して、**ROX_API_TOKEN** 変数としてエクスポートします。

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

2. (オプション): 次のコマンドを入力して、トークンをファイルに保存し、**--token-file** オプションとともに使用することもできます。

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

次のガイドラインに注意してください。

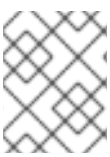
- **-password** (**-p**) オプションと **--token-file** オプションの両方を同時に使用することはできません。
- すでに **ROX_API_TOKEN** 変数を設定しており、**--token-file** オプションを指定していると、**roxctl** CLI は指定されたトークンファイルを認証に使用します。
- すでに **ROX_API_TOKEN** 変数を設定しており、**--password** オプションを指定していると、**roxctl** CLI は指定されたパスワードを認証に使用します。

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。



注記

Linux 用の **roxctl** CLI は、**amd64**、**arm64**、**ppc64le**、**s390x** アーキテクチャーで使用できます。

手順

1. ターゲットのオペレーティングシステムの **roxctl** アーキテクチャーを確認します。

```
$ arch="$(uname -m | sed "s/x86_64//")"; arch="${arch:+-$arch}"
```

2. **roxctl** CLI をダウンロードします。

```
$ curl -L -f -o roxctl
"https://mirror.openshift.com/pub/rhacs/assets/4.5.1/bin/Linux/roxctl${arch}"
```

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
```

2.3.2.2. 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。

注記

macOS 用の **roxctl** CLI は、**amd64** および **arm64** アーキテクチャーで使用できます。

手順

1. ターゲットのオペレーティングシステムの **roxctl** アーキテクチャーを確認します。

```
$ arch="$(uname -m | sed "s/x86_64//")"; arch="${arch:+-$arch}"
```

2. **roxctl** CLI をダウンロードします。

```
$ curl -L -f -o roxctl
"https://mirror.openshift.com/pub/rhacs/assets/4.5.1/bin/Darwin/roxctl${arch}"
```

3. バイナリーからすべての拡張属性を削除します。

```
$ xattr -c roxctl
```

4. **roxctl** バイナリーを実行可能にします。

```
$ chmod +x roxctl
```

5. **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。

注記

Windows 用の **roxctl** CLI は、**amd64** アーキテクチャーで使用できます。

手順

- **roxctl** CLI をダウンロードします。

```
$ curl -f -O https://mirror.openshift.com/pub/rhacs/assets/4.5.1/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. コンテナーから roxctl CLI の実行

roxctl クライアントは、RHACS **roxctl** イメージのデフォルトエントリーポイントです。コンテナーイメージで **roxctl** クライアントを実行するには、以下を行います。

前提条件

- はじめに、RHACS ポータルから認証トークンを生成している。

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
```

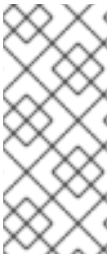
2. **roxctl** CLI の最新のコンテナーイメージをプルします。

```
$ docker pull registry.redhat.io/advanced-cluster-security/rhacs-roxctl-rhel8:4.5.1
```

-

CLI をインストールしたら、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-rhel8:4.5.1 \
  -e $ROX_CENTRAL_ADDRESS <command>
```



注記

Red Hat Advanced Cluster Security Cloud Service (RHACS Cloud Service) で、Central アドレスを必要とする **roxctl** コマンドを使用する場合は、Red Hat Hybrid Cloud Console の **Instance Details** セクションに表示される **Central** インスタンスアドレスを使用します。たとえば、**acs-data-ABCD12345.acs.rhcloud.com** の代わりに **acs-ABCD12345.acs.rhcloud.com** を使用します。

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-rhel8:4.5.1 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ー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 "$STACKROX_CENTRAL_HOST:443" --image "
<your_registry/repo/image_name>" 1
      - run:
          name: Scan deployment files for policy deviations
          command: |
            ./roxctl image check --endpoint "$STACKROX_CENTRAL_HOST:443" --image "
<your_deployment_file>" 2
            # 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
```

1. **<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。
2. **<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

リポジトリーに 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ーにコミットした後、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。

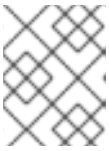
5. **PagerDuty integration key** フィールドに統合キーを入力します。
6. **Test** をクリックして、PagerDuty との統合が機能していることを確認します。
7. **Create** をクリックして設定を作成します。

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

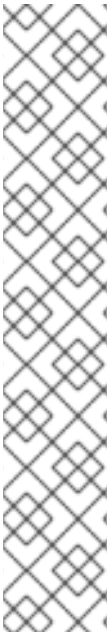
1. RHACS ポータルで、**Platform Configuration → Policy Management** に移動します。
2. アラートの送信先となるポリシーを 1 つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**PagerDuty** 通知機能を選択します。



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、Webhook URL 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

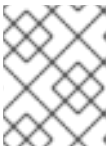
新しい Slack アプリケーションを作成することから始めて、Webhook の URL を取得します。

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 にアクセスしてください。
   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** 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 ドキュメントのトピック 受信 Webhook の開始 を参照してください。

4.1.1. さまざまな Slack チャネルにアラートを送信する

さまざまな Slack チャ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/XXXXXXXXXXXXXXXXXXXXXXXX
```

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

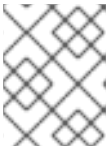
1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションまで下にスクロールして、**Slack** を選択します。
3. **New Integration** (**add** アイコン) をクリックします。
4. **Integration Name** の名前を入力します。
5. 生成された Webhook URL を **Default Slack Webhook** フィールドに入力します。
6. **Test** を選択して、Slack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

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

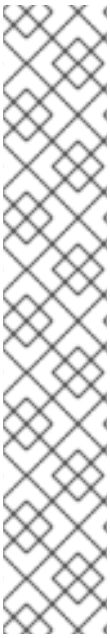
1. RHACS ポータルで、**Platform Configuration** → **Policy Management** に移動します。
2. アラートの送信先となるポリシーを 1 つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Slack** 通知機能を選択します。



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:

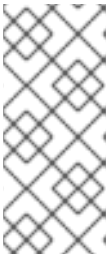
1. アラートを受信するための Webhook URL を設定します。
2. Webhook URL を使用して、Red Hat Advanced Cluster Security for Kubernetes で通知を設定します。
3. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。

5.1. WEBHOOK を使用した統合の設定

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。
6. 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。

7. 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。

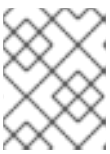
8. Webhook レシーバーで認証するには、次のいずれかの詳細を入力します。
   - 基本 HTTP 認証の ユーザー名 と パスワード
   - カスタム ヘッダー、例: **Authorization: Bearer <access_token>**
9. **Extra fields** を使用して、Red Hat Advanced Cluster Security for Kubernetes が送信する JSON オブジェクトに追加のキーと値のペアを含めます。たとえば、Webhook レシーバーが複数のソースからのオブジェクトを受け入れる場合は、追加フィールドとして **"source": "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。
10. **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。
11. **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

1. RHACS ポータルで、**Platform Configuration → Policy Management** 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、**webhook** 通知機能を選択します。



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 QRADAR との統合

RHACS で汎用 Webhook 統合を設定することにより、QRadar にイベントを送信するように Red Hat Advanced Cluster Security for Kubernetes を設定できます。

以下のステップは、RHACS を QRadar と統合するための大まかなワークフローを表しています。

1. RHACS では:
   a. 汎用 Webhook を設定します。

注記

RHACS で統合を設定するときは、**Endpoint** フィールドで、次の例をガイドとして使用します: **<URL to QRadar Box>:<Port of Integration>**。

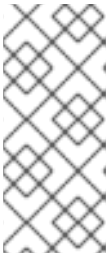
   b. 通知を送信するポリシーを特定し、それらのポリシーの通知設定を更新します。
2. QRadar がログソースを自動的に検出しない場合は、QRadar コンソールに RHACS ログソースを追加します。QRadar および RHACS の設定の詳細は、Red Hat Advanced Cluster Security for Kubernetes IBM リソースを参照してください。

6.1. WEBHOOK を使用した統合の設定

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

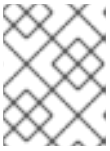
1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Notifier Integrations** セクションまでスクロールダウンし、**Generic Webhook** を選択します。
3. **New integration** をクリックします。
4. **Integration name** の名前を入力します。
5. **Endpoint** フィールドに Webhook の URL を入力します。
6. Webhook レシーバーが信頼できない証明書を使用している場合は、**CA certificate** 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。

7. 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。

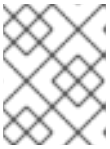
8. Webhook レシーバーで認証するには、次のいずれかの詳細を入力します。
   - 基本 HTTP 認証の ユーザー名 と パスワード
   - カスタム ヘッダー、例: **Authorization: Bearer <access_token>**
9. **Extra fields** を使用して、Red Hat Advanced Cluster Security for Kubernetes が送信する JSON オブジェクトに追加のキーと値のペアを含めます。たとえば、Webhook レシーバーが複数のソースからのオブジェクトを受け入れる場合は、追加フィールドとして **"source": "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。
10. **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。
11. **Save** を選択して設定を作成します。

6.2. 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** → **Policy Management** 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、**webhook** 通知機能を選択します。



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7章 SERVICENOW との統合

RHACS で汎用 Webhook 統合を設定することにより、イベントを ServiceNow に送信するように Red Hat Advanced Cluster Security for Kubernetes を設定できます。

次の手順は、RHACS を ServiceNow 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. ServiceNow で、RHACS で使用する REST API エンドポイントを設定します。ServiceNow 設定の手順を含む詳細は、Red Hat Advanced Cluster Security for Kubernetes と ServiceNow を統合する方法 を参照してください。
2. RHACS では:
   a. 汎用 Webhook を設定します。
   b. 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。

7.1. WEBHOOK を使用した統合の設定

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。
6. 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。

7. 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。

8. 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>**

9. **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は、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。
10. **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。
11. **Save** を選択して設定を作成します。

7.2. 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 → Policy Management** 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、**webhook** 通知機能を選択します。

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8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。

8.1. 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。

8.2. 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。

8.3. 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 → Policy Management** 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** 通知機能を選択します。

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。

8.4. 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じ **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9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障害やデータ破損が発生した場合のデータ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。バックアップがセキュア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。

9.1. 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**。詳細は、Using Cloud IAM permissions を参照してください。
- Workload Identity またはサービスアカウントの サービスアカウントキー **(JSON)** のいずれか。詳細は、Creating a service account および Creating service account keys を参照してください。

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. Workload Identity を使用する場合は、**Use workload identity** のチェックボックスをオンにします。使用しない場合は、サービスアカウントキーファイルの内容を **Service account key (JSON)** フィールドに入力します。

9. **Test** 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** を選択して設定を生成します。

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。

9.1.1. 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。

9.1.1.1. 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10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ー調査のために、アプリケーションが SIEM や syslog コレクターなどの Central 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。

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

Red Hat Advanced Cluster Security for Kubernetes (RHACS) 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. **Add new extra field** をクリックして、追加フィールドを追加します。たとえば、syslog レシーバーが複数のソースからオブジェクトを受け入れる場合は、**Key** フィールドと **Value** フィールドに **source** と **rhacs** を入力します。
   syslog レシーバーのカスタム値を使用してフィルタリングし、RHACS からのすべてのアラートを識別できます。

10. **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。

11. **Create** (**save** アイコン) を選択して、設定を作成します。

第11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。バックアップがセキュア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、AWS ルート CA を Red Hat Advanced Cluster Security for Kubernetes の 信頼できる認証局 として追加する必要があります。

11.1. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES で AMAZON S3 統合の設定

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ー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。詳細は、Amazon ドキュメントのトピック オブジェクトメタデータの操作 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** を選択して、Amazon S3 との統合が機能していることを確認します。
13. **Create** を選択して設定を生成します。

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。

11.2. AMAZON S3 でのオンデマンドバックアップの実行

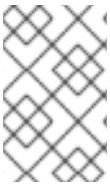
RHACS ポータルを使用して、Amazon S3 上の Red Hat Advanced Cluster Security for Kubernetes の手動バックアップをトリガーします。

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、**Amazon S3** 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。

11.3. 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12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 と統合する方法を説明します。

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。

12.1. GOOGLE CLOUD 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。詳細は、creating and managing service account keys Google Cloud のガイドを参照してください。

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

ソース ID と Google サービスアカウントを使用して、Red Hat Advanced Cluster Security for Kubernetes で新しい Google Cloud SCC 統合を作成できます。

前提条件

- 組織レベルの **Security Center Findings Editor** IAM ロールを持つ サービスアカウント。詳細は、Access control with IAM を参照してください。
- Workload Identity またはサービスアカウントの サービスアカウントキー **(JSON)** のいずれか。詳細は、Creating a service account および Creating service account keys を参照してください。

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**を入力します。
6. Workload Identity を使用する場合は、**Use workload identity** のチェックボックスをオンにします。使用しない場合は、サービスアカウントキーファイルの内容を **Service account key (JSON)** フィールドに入力します。
7. **Create** を選択して設定を生成します。

12.3. 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。

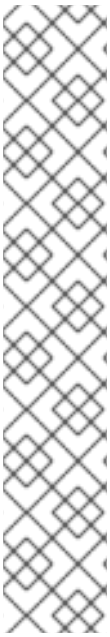
手順

1. RHACS ポータルで、**Platform Configuration → Policy Management** 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** 通知機能を選択します。

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。

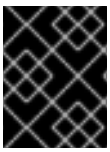


注記

- 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13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 内から違反、脆弱性検出、コンプライアンス関連データを表示できます。



重要

現在、Splunk インテグレーションは IBM Power (**ppc64le**) および IBM Z (**s390x**) ではサポートされていません。

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 HTTP イベントコレクターを使用する 場合:
  - 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。
- Red Hat Advanced Cluster Security for Kubernetes アドオンを使用する 場合:
  - アドオンを使用して、違反、脆弱性検出、およびコンプライアンスのデータを Splunk に取り込みます。

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。

13.1. 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。

13.1.1. 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ーを確認し、**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。

13.1.1.1. 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。

13.1.2. 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. Splunk URL 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。

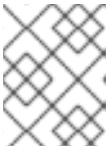
7. **Test** を選択してテストメッセージを送信し、Splunk との統合が機能していることを確認します。
8. **Create** を選択して設定を生成します。

13.1.3. 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

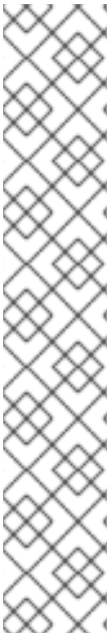
1. RHACS ポータルで、**Platform Configuration → Policy Management** に移動します。
2. アラートの送信先となるポリシーを 1 つ以上選択します。
3. **Bulk actions** で **Enable notification** を選択します。
4. **Enable notification** ウィンドウで、**Splunk** 通知機能を選択します。



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。

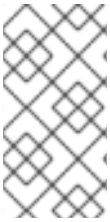
13.2. RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES アドオンの使用

Red Hat Advanced Cluster Security for Kubernetes アドオンを使用すると、脆弱性検出およびコンプライアンス関連のデータを 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 **read** 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。

13.2.1. Splunk アドオンのインストールと設定

Splunk インスタンスから Red Hat Advanced Cluster Security for Kubernetes アドオンをインストールできます。



注記

StackRox Kubernetes Security Platform アドオンとの下位互換性を維持するために、設定された入力の **source_type** パラメーターと **input_type** パラメーターは引き続き **stackrox_compliance**、**stackrox_violations**、および **stackrox_vulnerability_management** と呼ばれます。

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. Red Hat Advanced Cluster Security for Kubernetes 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、Red Hat Advanced Cluster Security for Kubernetes アドオンファイルを選択します。
6. **Upload** 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 **Red Hat Advanced Cluster Security for Kubernetes** を選択します。
9. **Configuration** に移動し、**Add-on Settings** をクリックします。
    a. **Central Endpoint** には、Central インスタンスの IP アドレスまたは名前を入力します。たとえば、**central.custom:443** です。
    b. アドオン用に生成した **API token** を入力します。
    c. **Save** をクリックします。
10. **Inputs** に移動します。
11. **Create New Input** をクリックし、次のいずれかを選択します。
    - コンプライアンスデータを取得する **ACS Compliance**
    - 違反データを取得する **ACS Violations**
    - 脆弱性データを取得する **ACS Vulnerability Management**
12. 入力の **Name** を入力します。
13. 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。

14. データの送信先となる Splunk **Index** を選択します。
15. **Central Endpoint** には、Central インスタンスの IP アドレスまたは名前を入力します。
16. アドオン用に生成した **API token** を入力します。
17. **Add** をクリックします。

検証

- Red Hat Advanced Cluster Security for Kubernetes アドオンのインストールを確認するには、受信したデータをクエリーします。
    a. Splunk インスタンスで、**Search** に移動し、クエリーに **index=* sourcetype="stackrox-*"** と入力します。
    b. **Enter** キーを押します。

設定したソースが検索結果に表示されることを確認します。

13.2.2. StackRox Kubernetes Security Platform アドオンの更新

StackRox Kubernetes Security Platform アドオンを使用している場合は、新しい Red Hat Advanced Cluster Security for Kubernetes アドオンにアップグレードする必要があります。

更新通知は、Splunk ホームページの左側のアプリのリストの下にあります。または、**Apps → Manage apps** ページに移動して更新通知を確認することもできます。

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. 更新通知で **Update** をクリックします。
2. 利用規約に同意するためのチェックボックスを選択し、**Accept and Continue** をクリックして更新をインストールします。
3. インストール後、**Apps** メニューから **Red Hat Advanced Cluster Security for Kubernetes**を選択します。
4. **Configuration** に移動し、**Add-on Settings** をクリックします。
    a. アドオン用に生成した **API token** を入力します。
    b. **Save** をクリックします。

13.2.3. Splunk アドオンのトラブルシューティング

Red Hat Advanced Cluster Security for Kubernetes アドオンからのイベントの受信を停止した場合は、Splunk アドオンのデバッグログでエラーを確認してください。

Splunk は、設定された入力ごとにデバッグログファイルを **/opt/splunk/var/log/splunk** ディレクトリーに作成します。**stackrox_<input>_<uid>.log** という名前のファイル (たとえば、**stackrox_compliance_29a3e14798aa2363d.log**) を見つけて、問題を探します。

第14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、脆弱性スキャナーと統合されているため、コンテナーイメージをインポートして、脆弱性を監視できます。

サポート対象のコンテナーイメージレジストリー

Red Hat は、次のコンテナーイメージレジストリーをサポートしています。

- Amazon Elastic Container Registry (ECR)
- 汎用 Docker レジストリー (任意の汎用 Docker または Open Container Initiative 準拠のイメージレジストリー、たとえば、DockerHub、**gcr.io**、**mcr.microsoft.com**)
- Google Container Registry
- Google Artifact Registry
- IBM Cloud Container Registry
- JFrog Artifactory
- Microsoft Azure Container Registry (ACR)
- Red Hat Quay
- Red Hat レジストリー (**registry.redhat.io**、**registry.access.redhat.com**)
- Sonatype Nexus

この強化されたサポートにより、優先レジストリーでコンテナーイメージを管理する際の柔軟性と選択肢がさらに広がります。

サポート対象のスキャナー

次の商用コンテナーイメージ脆弱性スキャナからイメージ脆弱性データを取得するように RHACS を設定できます。

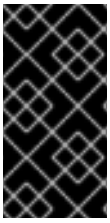
RHACS に含まれるスキャナー

- Scanner V4: RHACS バージョン 4.4 以降、Clair スキャナーにも利用されている ClairCore 上に構築された新しいスキャナーが導入されました。Scanner V4 は、言語および OS 固有のイメージコンポーネントのスキャンをサポートしています。このスキャナーを使用するために統合を作成する必要はありませんが、インストール中またはインストール後にスキャナーを有効にする必要があります。バージョン 4.4 でこのスキャナーを有効にする場合は、StackRox Scanner も有効にする必要があります。インストールドキュメントへのリンクなど、Scanner V4 の詳細は、RHACS Scanner V4 について を参照してください。
- StackRox Scanner: このスキャナーは、RHACS のデフォルトのスキャナーです。これは、Clair v2 オープンソーススキャナーのフォークから生まれました。Scanner V4 を有効にした場合は、RHCOS ノードと、Red Hat OpenShift、Kubernetes、Istio などのプラットフォームの脆弱性のスキャンを提供するために、このスキャナーも有効にする必要があります。この機能は、今後のリリースの Scanner V4 でサポートされる予定です。

代替スキャナー

- Clair: バージョン 4.4 以降の RHACS では、Scanner V4 を有効にすると、Clair V4 スキャナーにも利用されている ClairCore の機能を提供できます。一方、統合を設定すると、Clair V4 をスキャナーとして設定できます。

- Google Container Analysis
- Red Hat Quay



重要

StackRox Scanner は、Scanner V4 (オプション) と併せて RHACS で使用することが推奨されるイメージ脆弱性スキャナーです。StackRox Scanner および Scanner V4 を使用したコンテナーイメージのスキャンの詳細は、イメージのスキャン を参照してください。

DevOps ワークフローでこれらの代替スキャナーのいずれかを使用する場合は、RHACS ポータルを使用して脆弱性スキャナーとの統合を設定できます。統合後、RHACS ポータルにイメージの脆弱性が表示されるため、簡単にトリアージできます。

複数のスキャナーが設定されている場合、RHACS は StackRox/RHACS 以外のスキャナーと Clair スキャナーの使用を試みます。これらのスキャナーが失敗した場合、RHACS は設定された Clair スキャナーの使用を試みます。それが失敗した場合、RHACS は Scanner V4 の使用を試みます (設定されている場合)。Scanner V4 が設定されていない場合、RHACS は StackRox Scanner の使用を試みます。

14.1. CLAIR との統合

バージョン 4.4 以降、Clair スキャン機能は RHACS の新しいスキャナーである Scanner V4 で利用できるようになりました。個別の統合は必要ありません。このセクションの手順は、Clair V4 スキャナーを使用している場合にのみ必要です。

次のガイドラインに注意してください。

- Red Hat は、RHACS 3.74 以降、Clair V4 統合を優先して、以前の CoreOS Clair 統合を非推奨にしました。Clair V4 スキャナーを使用するには、別の統合が必要でした。バージョン 4.4 以降、Scanner V4 を使用している場合、この統合は必要なくなりました。
- 次の RHACS 4.0 のバージョンで、Clair V4 統合用の JWT ベースの認証 オプションをサポートする予定はありません。

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 v4** 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**: スキャナーのアドレス。
5. オプション: 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. (オプション) **Test** をクリックして、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** をクリックします。

14.2. GOOGLE CONTAINER REGISTRY との統合

コンテナー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。**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)**認証に使用する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。

14.3. イメージをスキャンするための QUAY CONTAINER REGISTRY との統合

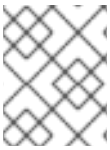
イメージをスキャンするために、Red Hat Advanced Cluster Security for Kubernetes を Quay Container Registry と統合できます。

前提条件

- イメージをスキャンするには、Quay Container Registry で認証するための OAuth トークンが必要です。

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 **Red Hat Quay.io** を選択します。
3. **New integration** をクリックします。
4. **Integration name** を入力します。
5. **Type** で、**Scanner** を選択します。(レジストリーとも統合する場合は、**Scanner + Registry** を選択します。)以下のフィールドに情報を入力します。
   - **Endpoint**: レジストリーのアドレスを入力します。
   - **OAuth token**: RHACS が API を使用して認証するために使用する OAuth トークンを入力します。
   - オプション: **Robot username**: **Scanner + Registry** を設定しており、Quay ロボットアカウントを使用してレジストリーにアクセスしている場合は、**<namespace>+<accountname>** の形式でユーザー名を入力します。
   - オプション: **Robot password**: **Scanner + Registry** を設定していて、Quay ロボットアカウントを使用してレジストリーにアクセスしている場合は、ロボットアカウントのユーザー名のパスワードを入力します。
6. オプション: 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。
7. オプション: テストを行わずにインテグレーションを作成するには、**Create integration without testing** を選択します。
8. **Save** を選択します。



注記

Quay インテグレーションを編集しているが資格情報を更新したくない場合は、**Update stored credentials** が選択されていないことを確認します。

第15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。

15.1. 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。

15.2. 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** フィールドおよび **Password or API token** フィールド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** フィールドに 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** フィールドに入力します。
9. オプション： **Annotation key for project**フィールドを使用して、次の手順を実行して、さまざまな Jira プロジェクトに課題を作成します。アノテーションを使用して、問題を動的に作成できます。
    a. namespace またはデプロイメント 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。**jira/project-key** は Jira インテグレーションで指定したアノテーションキーです。デプロイメントまたは namespace の注釈を作成できます。

    ```
    annotations:
    # ...
      jira/project-key: <jira_project_key>
    # ...
    ```

    b. **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。
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** フィールドに 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** 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** を選択して設定を生成します。

15.2.1. 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 プロジェクトで課題を作成し、正しいチームに直接移動できるようにすることができます。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、RHACS はそのアノテーションに指定されたプロジェクトに問題を作成します。それ以外の場合、RHACS はデフォルトプロジェクトに問題を作成します。

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. namespace またはデプロイメント 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。

```
annotations:
# ...
  jira/project-key: <jira_project_key>
# ...
```

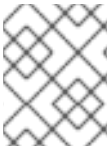
2. Red Hat Advanced Cluster Security for Kubernetes を設定するときは、**Annotation key for project** フィールドでアノテーションキー **jira/project-key** を使用します。

15.2.2. 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** 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** を選択して設定を生成します。



注記

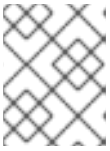
エラーが発生した場合は、Jira 統合のトラブルシューティング セクションの指示に従ってください。

15.3. 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** → **Policy Management** 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** 通知機能を選択します。



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。

15.4. 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16章 メールとの統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) を使用すると、既存のメールプロバイダーを設定して、ポリシー違反に関する通知を送信できます。Red Hat Advanced Cluster Security Cloud Service (RHACS Cloud Service) を使用している場合は、既存のメールプロバイダーまたは組み込みのメール通知機能を使用してメール通知を送信できます。

Default recipient フィールドを使用して、RHACS および RHACS Cloud Service からのアラートをメールアドレスに転送できます。それ以外の場合は、アノテーションを使用して対象者を定義し、特定のデプロイメントまたは namespace に関連するポリシー違反について通知することができます。

16.1. RHACS でのメールとの統合

RHACS からアラートを転送することで、メールを通知方法として使用できます。

16.1.1. メールプラグインの設定

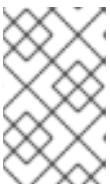
RHACS 通知機能は、インテグレーションで指定された受信者にメールを送信したり、アノテーションを使用して受信者を決定したりできます。

重要

RHACS Cloud Service を使用している場合、デフォルトでポート **25** がブロックされます。ポート **587** または **465** を使用してメール通知を送信するようにメールサーバーを設定してください。

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** を選択します。
4. **Integration Name** フィールドに、メール統合の名前を入力します。
5. **Email server** フィールドに、メールサーバーのアドレスを入力します。メールサーバーアドレスには、完全修飾ドメイン名 (FQDN) とポート番号が含まれます (例: **smtp.example.com:465**)。
6. オプション: 非認証 SMTP を使用している場合は、**Enable unauthenticated SMTP** を選択します。これは安全ではなく、推奨されませんが、インテグレーションによっては必要になる場合があります。たとえば、認証を必要としない通知に内部サーバーを使用する場合、このオプションを有効にする必要がある場合があります。



注記

認証を使用する既存のメール統合を変更して、非認証 SMTP を有効にすることはできません。既存の統合を削除し、**Enable unauthenticated SMTP** を選択して新しい統合を作成する必要があります。

7. 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。

8. オプション: メール通知の **FROM** ヘッダーに表示する名前を **From** フィールドに入力します。たとえば、**Security Alerts** などです。

9. メール通知の **SENDER** ヘッダーに表示するメールアドレスを **Sender** フィールドに指定します。

10. **Default recipient** フィールドに、通知を受信するメールアドレスを指定します。

11. オプション: **Annotation key for recipient**にアノテーションキーを入力します。アノテーションを使用すると、メールの受信者を動的に決定できます。これを実行するには、以下を行います。

    a. namespace またはデプロイメント 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。**email** は、メール統合で指定する **Annotation key** です。デプロイメントまたは namespace の注釈を作成できます。

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

    b. **Annotation key for recipient**フィールドにアノテーションキーの **email** を使用します。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 はアノテーションで指定されたメールアドレス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。

注記

RHACS がメール通知の受信者を決定する仕組みは、次のルールによって制御されます。

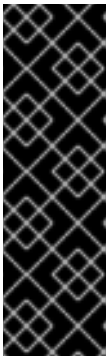
- デプロイメントにアノテーションキーがある場合、アノテーションの値がデフォルト値をオーバーライドします。
- namespace にアノテーションキーがある場合、namespace の値がデフォルト値をオーバーライドします。
- デプロイメントにアノテーションキーと定義済みの対象者がある場合、RHACS はキーで指定された対象者にメールを送信します。
- デプロイメントにアノテーションキーがない場合、RHACS は namespace でアノテーションキーを確認し、指定された対象者にメールを送信します。
- アノテーションキーが存在しない場合、RHACS はデフォルトの受信者にメールを送信します。

12. オプション: **Disable TLS certificate validation (insecure)** を選択して、TLS を使用せずにメールを送信します。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

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

13. 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。

関連情報

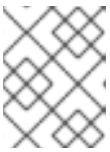
- 配信先およびスケジューリングの設定

16.1.2. 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** → **Policy Management** 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、**Email** 通知機能を選択します。



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。

16.2. RHACS CLOUD SERVICE でのメールとの統合

既存のメールプロバイダーまたは RHACS Cloud Service に組み込まれているメール通知機能を使用して、ポリシー違反に関するメールアラートを送信できます。

- 独自のメールプロバイダーを使用するには、メールプラグインの設定 セクションで説明されているようにメールプロバイダーを設定する必要があります。
- 組み込みのメール通知機能を使用するには、RHACS Cloud Service メールプラグインを設定する必要があります。

16.2.1. RHACS Cloud Service メールプラグインの設定

RHACS Cloud Service 通知機能は受信者にメールを送信します。統合で受信者を指定することも、RHACS Cloud Service でアノテーションキーを使用して受信者を検索することもできます。

重要

- 送信できるメールは、24 時間ごとに 250 通までです。この制限を超えた場合、RHACS Cloud Service は 24 時間が経過してからメールを送信します。
- レート制限のため、Red Hat では、重要なアラートまたは脆弱性レポートにのみメール通知を使用することを推奨しています。

手順

1. **Platform Configuration** → **Integrations** に移動します。
2. **Notifier Integrations** セクションで、**RHACS Cloud Service Email**を選択します。
3. **New Integration** を選択します。
4. **Integration Name** フィールドに、メール統合の名前を入力します。
5. **Default recipient** フィールドに、メール通知を送信するメールアドレスを指定します。
6. オプション: **Annotation key for recipient**にアノテーションキーを入力します。アノテーションを使用すると、メールの受信者を動的に決定できます。これを実行するには、以下を行います。
   a. namespace またはデプロイメント 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。**email** は、メール統合で指定する **Annotation key** です。デプロイメントまたは namespace の注釈を作成できます。

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

   b. **Annotation key for recipient** フィールドにアノテーションキーの **email** を使用します。

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 Cloud Service はアノテーションで指定されたメールアドレス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。

注記

RHACS Cloud Service がメール通知の受信者を決定する仕組みは、次のルールによって制御されます。

- デプロイメントにアノテーションキーがある場合、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 にアノテーションキーがある場合、namespace の値がデフォルト値をオーバーライドします。
- デプロイメントにアノテーションキーと定義済みの対象者がある場合、RHACS Cloud Service はキーで指定された対象者にメールを送信します。
- デプロイメントにアノテーションキーがない場合、RHACS Cloud Service は namespace でアノテーションキーを確認し、指定された対象者にメールを送信します。
- アノテーションキーが存在しない場合、RHACS Cloud Service はデフォルトの受信者にメールを送信します。

関連情報

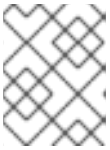
- 配信先およびスケジューリングの設定

16.2.2. 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** → **Policy Management** 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、**RHACS Cloud Service Email**通知機能を選択します。



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17章 クラウド管理プラットフォームとの統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) をさまざまなクラウド管理プラットフォームと統合して、保護対象の候補となるクラスターを検出できます。クラスター検出の目的は、RHACS によってすでに保護されているクラスターアセットやまだ保護されていないクラスターアセットの概要を詳しく把握することです。

クラウド管理プラットフォームによって検出されたクラスターには、**Platform Configuration → Clusters → Discovered clusters** ページからアクセスできます。

RHACS は、検出されたクラスターをすでに保護されているクラスターと照合します。検出されたクラスターは、照合の結果に応じて、次のいずれかのステータスになります。

- **Secured**: クラスターは RHACS によって保護されています。
- **Unsecured**: クラスターは RHACS によって保護されていません。
- **Undetermined**: 保護対象のクラスターから収集されたメタデータが不十分なため、一意の一致が見つかりません。クラスターは保護されているか、保護されていないかのどちらかです。

クラスターの照合を正常に行うには、次の条件が満たされていることを確認してください。

- 保護対象のクラスター上で実行されている Sensor が最新バージョンに更新されている。
- AWS 上で実行されている保護対象のクラスターに対して、インスタンスメタデータのタグへのアクセス が許可されている。Sensor は、クラスターのステータスを判断するために AWS EC2 インスタンスタグにアクセスする必要があります。

RHACS は次のクラウド管理プラットフォームと統合できます。

- Paladin Cloud
- OpenShift Cluster Manager

17.1. PALADIN CLOUD 統合の設定

Paladin Cloud からクラスターアセットを検出するには、Red Hat Advanced Cluster Security for Kubernetes で新しい統合を作成します。

前提条件

- Paladin Cloud のアカウント
- Paladin Cloud API トークン

手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Cloud source integrations** セクションまで下にスクロールし、**Paladin Cloud** 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. **Paladin Cloud endpoint** で Paladin Cloud の API エンドポイントを入力します。デフォルトは **https://api.paladincloud.io** です。
6. **Paladin Cloud token** で Paladin Cloud の API トークンを入力します。
7. **Test** を選択して、認証が機能していることを確認します。
8. **Create** を選択して設定を生成します。

設定が完了すると、Red Hat Advanced Cluster Security for Kubernetes が、接続された Paladin Cloud アカウントからクラスターアセットを検出します。

17.2. RED HAT OPENSHIFT CLUSTER MANAGER 統合の設定

Red Hat OpenShift Cluster Manager からクラスターアセットを検出するには、Red Hat Advanced Cluster Security for Kubernetes で新しい統合を作成します。

前提条件

- Red Hat アカウント
- Red Hat OpenShift Cluster Manager API トークン

手順

1. RHACS ポータルで、**Platform Configuration** → **Integrations** に移動します。
2. **Cloud source integrations** セクションまで下にスクロールし、**Red Hat OpenShift Cluster Manager** 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** に Red Hat OpenShift Cluster Manager の API エンドポイントを入力します。デフォルトは **https://api.openshift.com** です。
6. **API token** に Red Hat OpenShift Cluster Manager の API トークンを入力します。
7. **Test** を選択して、認証が機能していることを確認します。
8. **Create** を選択して設定を生成します。

設定が完了すると、Red Hat Advanced Cluster Security for Kubernetes が、接続された Red Hat アカウントからクラスターアセットを検出します。

第18章 有効期間の短いトークンを使用した RHACS の統合

Red Hat Advanced Cluster Security for Kubernetes (RHACS) を使用すると、有効期間の短いトークンを使用して、選択したクラウドプロバイダー API に対して認証できます。RHACS は、次のクラウドプロバイダー統合をサポートしています。

- Secure Token Service (STS) を使用する Amazon Web Services (AWS)
- Workload Identity 連携を使用する Google Cloud Platform (GCP)

RHACS は、次のプラットフォームに RHACS をインストールする場合にのみ、有効期間の短いトークンによる統合をサポートします。

- AWS 上の Elastic Kubernetes Service (EKS)
- GCP 上の Google Kubernetes Engine (GKE)
- OpenShift Container Platform

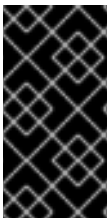
有効期間の短い認証を有効にするには、Kubernetes または OpenShift Container Platform クラスターとクラウドプロバイダーの間に信頼関係を確立する必要があります。EKS および GKE クラスターの場合は、クラウドプロバイダーのメタデータサービスを使用します。OpenShift Container Platform クラスターの場合は、OpenShift Container Platform サービスアカウント署名者キーを含む、公開されている OpenID Connect (OIDC) プロバイダーバケットが必要です。

注記

有効期間の短いトークンによる統合を使用するすべての Central クラスターで、クラウドプロバイダーとの信頼関係を確立する必要があります。ただし、有効期間の短いトークンイメージによる統合とスキャン委譲を組み合わせて使用する場合は、Sensor クラスターでも信頼関係を確立する必要があります。

18.1. AWS SECURE TOKEN SERVICE の設定

RHACS の統合では、Secure Token Service を使用して Amazon Web Services に対して認証できます。統合の **Use container IAM role** オプションを有効にするには、RHACS で **AssumeRole** を設定する必要があります。



重要

RHACS の Pod に関連付けられた AWS ロールには、統合に必要な IAM 権限が必要です。たとえば、Elastic Container Registry と統合するためのコンテナーロールを設定するには、レジストリーへの完全な読み取りアクセスを有効にしてください。AWS IAM ロールの詳細は、IAM roles を参照してください。

18.1.1. Elastic Kubernetes Service (EKS) の設定

EKS で Red Hat Advanced Cluster Security for Kubernetes (RHACS) を実行する場合、Amazon Secure Token Service を通じて有効期間の短いトークンを設定できます。

手順

1. 次のコマンドを実行して、EKS クラスターの IAM OpenID Connect (OIDC) プロバイダーを有効にします。

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

2. EKS クラスターの IAM ロールを作成し ます。
3. ロールの権限ポリシーを編集し、統合に必要な権限を付与します。以下に例を示します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:DescribeImages",
                "ecr:DescribeRepositories",
                "ecr:GetAuthorizationToken",
                "ecr:GetDownloadUrlForLayer",
                "ecr:ListImages"
            ],
            "Resource": "arn:aws:iam::<ecr_registry>:role/<role_name>"
        }
    ]
}
```

4. 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>" (1)
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

(1) **<role_name>** は、前の手順で作成した新しいロールと一致している必要があります。

5. 次のコマンドを入力して、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。

```
$ oc -n stackrox annotate sa central eks.amazonaws.com/role-
arn=arn:aws:iam::67890:role/<role_name> (1)
```

(1) Kubernetes を使用する場合は、**oc** の代わりに **kubectl** を入力します。

6. 次のコマンドを入力して Central Pod を再起動し、変更を適用します。

```
$ oc -n stackrox delete pod -l "app in (central,sensor)" 1
```

1 Kubernetes を使用する場合は、**oc** の代わりに **kubectl** を入力します。

18.1.2. OpenShift Container Platform の設定

OpenShift Container Platform で Red Hat Advanced Cluster Security for Kubernetes (RHACS) を実行する場合、Amazon Secure Token Service を通じて有効期間の短いトークンを設定できます。

前提条件

- OpenShift Container Platform サービスアカウント署名者キーを含むパブリックな OpenID Connect (OIDC) 設定バケットがある。OpenShift Container Platform クラスターの OIDC 設定を取得するには、Cloud Credential Operator in manual mode for short-term credentials の手順を使用することを推奨します。
- AWS IAM へのアクセス権と、ロールを作成および変更する権限がある。

手順

1. Creating OpenID Connect (OIDC) identity providers の手順に従って、OpenShift Container Platform クラスターの Web アイデンティティーを作成します。**openshift** を **Audience** の値として使用します。
2. OpenShift Container Platform クラスターの Web アイデンティティー用の IAM ロールを作成します。
3. ロールの権限ポリシーを編集し、統合に必要な権限を付与します。以下に例を示します。

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "VisualEditor0",
         "Effect": "Allow",
         "Action": [
            "ecr:BatchCheckLayerAvailability",
            "ecr:BatchGetImage",
            "ecr:DescribeImages",
            "ecr:DescribeRepositories",
            "ecr:GetAuthorizationToken",
            "ecr:GetDownloadUrlForLayer",
            "ecr:ListImages"
         ],
         "Resource": "arn:aws:iam::<ecr_registry>:role/<role_name>"
      }
   ]
}
```

4. 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": {
          "Federated": "<oidc_provider_arn>"
        },
        "Action": "sts:AssumeRoleWithWebIdentity",
        "Condition": {
          "StringEquals": {
            "<oidc_provider_name>:aud": "openshift"
          }
        }
      }
    ]
}
```

5. Central または Sensor デプロイメントで次の RHACS 環境変数を設定します。

```
AWS_ROLE_ARN=<role_arn>
AWS_WEB_IDENTITY_TOKEN_FILE=/var/run/secrets/openshift/serviceaccount/token
```

18.2. GOOGLE の WORKLOAD IDENTITY 連携の設定

RHACS の統合では、Workload Identity を使用して Google Cloud Platform に対して認証できます。作成時に **Use workload identity** オプションを選択して、Google Cloud 統合で Workload Identity 認証を有効にします。

重要

Workload Identity を通じて RHACS の Pod に関連付けられた Google サービスアカウントには、統合に必要な IAM 権限が必要です。たとえば、Google Artifact Registry と統合するための Workload Identity を設定するには、サービスアカウントを **roles/artifactregistry.reader** ロールに接続してください。Google IAM ロールの詳細は、Configure roles and permissions を参照してください。

18.2.1. Google Kubernetes Engine (GKE) の設定

GKE で Red Hat Advanced Cluster Security for Kubernetes (RHACS) を実行する場合、Google Workload Identity を通じて有効期間の短いトークンを設定できます。

前提条件

- クラスターと統合リソースを含む Google Cloud プロジェクトにアクセスできる。

手順

1. Google Cloud Platform ドキュメントの手順に従って、GKE 用 Workload Identity 連携を使用 します。
2. 次のコマンドを実行して、RHACS サービスアカウントにアノテーションを付けます。

```
$ oc annotate serviceaccount \ 1
    central \ 2
```

```
    --namespace stackrox \
    iam.gke.io/gcp-service-account=
<GSA_NAME>@<GSA_PROJECT>.iam.gserviceaccount.com
```

1. Kubernetes を使用する場合は、**oc** の代わりに **kubectl** を入力します。
2. スキャン委譲を設定する場合は、**central** の代わりに **sensor** を使用します。

18.2.2. OpenShift Container Platform の設定

OpenShift Container Platform で Red Hat Advanced Cluster Security for Kubernetes (RHACS) を実行する場合、Google Workload Identity を通じて有効期間の短いトークンを設定できます。

前提条件

- OpenShift Container Platform サービスアカウント署名者キーを含むパブリックな OIDC 設定バケットがある。OpenShift Container Platform クラスターの OIDC 設定を取得するには、Cloud Credential Operator in manual mode for short-term credentials の手順を使用することを推奨します。
- **role/iam.workloadIdentityPoolAdmin** ロールを使用して Google Cloud プロジェクトにアクセスします。

手順

1. Manage workload identity pools の手順に従って、Workload Identity プールを作成します。以下に例を示します。

```
$ gcloud iam workload-identity-pools create rhacs-pool \
    --location="global" \
    --display-name="RHACS workload pool"
```

2. Manage workload identity pool providers の手順に従って、Workload Identity プロバイダーを作成します。以下に例を示します。

```
$ gcloud iam workload-identity-pools providers create-oidc rhacs-provider \
    --location="global" \
    --workload-identity-pool="rhacs-pool" \
    --display-name="RHACS provider" \
    --attribute-mapping="google.subject=assertion.sub" \
    --issuer-uri="https://<oidc_configuration_url>" \
    --allowed-audiences=openshift
```

3. Google サービスアカウントを Workload Identity プールに接続します。以下に例を示します。

```
$ gcloud iam service-accounts add-iam-policy-binding
<GSA_NAME>@<GSA_PROJECT>.iam.gserviceaccount.com \
    --role roles/iam.workloadIdentityUser \
    --
member="principal://iam.googleapis.com/projects/<GSA_PROJECT_NUMBER>/locations/globa
l/workloadIdentityPools/rhacs-provider/subject/system:serviceaccount:stackrox:central" (1)
```

1 スキャン委譲を設定する場合は、サブジェクトを **system:serviceaccount:stackrox:sensor** に設定します。

4. Security Token Service (STS) 設定を含むサービスアカウント JSON を作成します。以下に例を示します。

```
{
  "type": "external_account",
  "audience":
"//iam.googleapis.com/projects/<GSA_PROJECT_ID>/locations/global/workloadIdentityPools/rhacs-pool/providers/rhacs-provider",
  "subject_token_type": "urn:ietf:params:oauth:token-type:jwt",
  "token_url": "https://sts.googleapis.com/v1/token",
  "service_account_impersonation_url": "https://iamcredentials.googleapis.com/v1/projects/-/serviceAccounts/<GSA_NAME>@<GSA_PROJECT>.iam.gserviceaccount.com:generateAccesToken",
  "credential_source": {
    "file": "/var/run/secrets/openshift/serviceaccount/token",
    "format": {
      "type": "text"
    }
  }
}
```

5. サービスアカウントの JSON を RHACS namespace へのシークレットとして使用します。

```
apiVersion: v1
kind: Secret
metadata:
  name: gcp-cloud-credentials
  namespace: stackrox
data:
  credentials: <base64_encoded_json>
```