Red Hat Advanced Cluster Security for Kubernetes 3.69

アップグレード

ここに簡単な説明を入力します。

Last Updated: 2022-10-31

Red Hat Advanced Cluster Security for Kubernetes 3.69 アップグレード

ここに簡単な説明を入力します。

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

法律上の通知

概要

本の主題と目的の短い概要、通常が 1 パラグラフを超えない長さ。

目次

第1章 HELM チャートを使用したアップグレード

Helm チャートを使用して Red Hat Advanced Cluster Security for Kubernetes をインストールしていて、Red Hat Advanced Cluster Security for Kubernetes の最新バージョンにアップグレードするには、次の手順を実行する必要があります。

- Helm チャートを更新します。
- central-services Helm チャートの設定ファイルを更新します。
- central-services Helm チャートをアップグレードします。
- secured-cluster-services Helm チャートの設定ファイルを更新します。
- secured-cluster-services Helm チャートをアップグレードします。

1.1. HELM チャートリポジトリーの更新

Red Hat Advanced Cluster Security for Kubernetes の新しいバージョンにアップグレードする前に、常に Helm チャートを更新する必要があります。

前提条件

- Red Hat Advanced Cluster Security for Kubernetes の Helm チャートリポジトリーをすでに追加している必要があります。

手順

- Red Hat Advanced Cluster Security for Kubernetes チャートリポジトリーを追加します。

```
$ helm repo update
```

検証

- 次のコマンドを実行して、追加されたチャートリポジトリーを確認します。

```
$ helm search repo -l rhacs/
```

1.2. 関連情報

- central-services Helm チャートの設定

1.3. CENTRAL-SERVICES HELM チャートをデプロイした後の設定オプションの変更

central-services Helm チャートをデプロイした後、任意の設定オプションに変更を加えることができます。

手順

1. **values-public.yaml** および **values-private.yaml** 設定ファイルを新しい値で更新します。

2. **helm upgrade** コマンドを実行し、**-f** オプションを使用して設定ファイルを指定します。

```
$ helm upgrade -n stackrox \
  stackrox-central-services rhacs/central-services \
  -f <path_to_values_public.yaml> \
  -f <path_to_values_private.yaml>
```

注記

--set または **--set-file** パラメーターを使用して設定値を指定することもできます。ただし、これらのオプションは保存されないため、変更を加えるたびにすべてのオプションを手動で再度指定する必要があります。

1.4. SECURE-CLUSTER-SERVICES HELM チャートをデプロイした後の設定オプションの変更

secure-cluster-services Helm チャートをデプロイした後、任意の設定オプションに変更を加えることができます。

手順

1. **values-public.yaml** および **values-private.yaml** 設定ファイルを新しい値で更新します。
2. **helm upgrade** コマンドを実行し、**-f** オプションを使用して設定ファイルを指定します。

```
$ helm upgrade -n stackrox \
  stackrox-secured-cluster-services rhacs/secured-cluster-services \
  --reuse-values \ 1
  -f <path_to_values_public.yaml> \
  -f <path_to_values_private.yaml>
```

1 **--reuse-values** パラメーターを指定する必要があります。指定しない場合、Helm upgrade コマンドは以前に設定されたすべての設定をリセットします。

注記

--set または **--set-file** パラメーターを使用して設定値を指定することもできます。ただし、これらのオプションは保存されないため、変更を加えるたびにすべてのオプションを手動で再度指定する必要があります。

第2章 ROXCTL CLI を使用して手動でアップグレード

サポートされている古いバージョンから、Red Hat Advanced Cluster Security for Kubernetes の最新バージョンにアップグレードできます。

Red Hat Advanced Cluster Security for Kubernetes を最新バージョンにアップグレードするには、以下を実行する必要があります。

- Central データベースをバックアップする
- Central をアップグレードする
- **roxctl** CLI をアップグレードする
- スキャナーをアップグレードする
- 保護されたすべてのクラスターがアップグレードされていることを確認する

2.1. CENTRAL データベースのバックアップ

Central データベースをバックアップし、そのバックアップを使用して、インフラストラクチャーの障害が発生した場合に、失敗したアップグレードまたはデータの復元からロールバックすることができます。

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** システムロールには、すべてのリソースに対する **read** 権限があります。
- **roxctl** CLI をインストールしました。
- **ROX_API_TOKEN** および **ROX_CENTRAL_ADDRESS** 環境変数を設定しました。

手順

- backup コマンドを実行します。
  - Red Hat Advanced Cluster Security for Kubernetes 3.0.55 以降の場合:

    ```
    $ roxctl -e "$ROX_CENTRAL_ADDRESS" central backup
    ```

  - Red Hat Advanced Cluster Security for Kubernetes 3.0.54 以前の場合:

    ```
    $ roxctl -e "$ROX_CENTRAL_ADDRESS" central db backup
    ```

関連情報

- roxctl CLI を使用した認証

2.2. CENTRAL クラスターのアップグレード

Central データベースをバックアップしたら、次のステップは Central クラスターをアップグレードすることです。この手順には、Central、**roxctl** CLI、および Scanner のアップグレードが含まれます。

2.2.1. Central のアップグレード

更新されたイメージをダウンロードしてデプロイすることにより、Central を最新バージョンに更新できます。

前提条件

- プライベートイメージレジストリーからイメージをデプロイする場合は、最初に新しいイメージをプライベートレジストリーにプッシュしてから、このセクションのコマンドのイメージレジストリーを置き換えます。

手順

- 次のコマンドを実行して、Central をアップグレードします。

```
$ oc -n stackrox patch deploy/central -p '{"spec":{"template":{"spec":{"containers":
[{"name":"central","env":[{"name":"ROX_NAMESPACE","valueFrom":{"fieldRef":
{"fieldPath":"metadata.namespace"}}}]}}}}}' 1
```

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

```
$ oc -n stackrox patch deployment/scanner -p '{"spec":{"template":{"spec":{"containers":
[{"name":"scanner","securityContext":{"runAsUser":65534}}]}}}}' 1
```

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

```
$ oc -n stackrox set image deploy/central central=registry.redhat.io/advanced-cluster-
security/rhacs-main-rhel8:3.69.2 1
```

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

重要

- Red Hat Advanced Cluster Security for Kubernetes 3.65.0 からアップグレードする場合は、次の追加コマンドを実行して、**stackrox-central-diagnostics** ロールを作成する必要があります。

```
$ oc -n stackrox patch role stackrox-central-diagnostics -p '{"rules":
[{"apiGroups":["*"],"resources":
["deployments","daemonsets","replicasets","configmaps","services"],"verbs":
["get","list"]}]}' 1
```

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

- Helm または Operator を使用して Red Hat Advanced Cluster Security for Kubernetes をインストールしておらず、OpenShift OAuth サーバーを使用して認証を有効にする場合は、次の追加コマンドを実行する必要があります。

```
$ oc -n stackrox set env deploy/central
ROX_ENABLE_OPENSHIFT_AUTH=true
```

```
$ oc -n stackrox patch serviceaccount/central -p '
{
"metadata": {
"annotations": {
"serviceaccounts.openshift.io/oauth-redirecturi.main":
"sso/providers/openshift/callback",
"serviceaccounts.openshift.io/oauth-redirectreference.main": "
{"kind":"OAuthRedirectReference","apiVersion":"v1","reference":
{"kind":"Route","name":"central"}}"
}
}
}'
```

検証

- 新しい Pod がデプロイされていることを確認します。

```
$ oc get deploy -n stackrox -o wide 1
```

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

```
$ oc get pod -n stackrox --watch 1
```

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

2.2.2. roxctl CLI のアップグレード

roxctl CLI を最新バージョンにアップグレードするには、既存のバージョンの **roxctl** CLI をアンインストールしてから、最新バージョンの **roxctl** CLI をインストールする必要があります。

2.2.2.1. roxctl CLI のアンインストール

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

手順

- **roxctl** バイナリーを見つけて削除します。

```
$ ROXPATH=$(which roxctl) && rm -f $ROXPATH
```
(1)

(1) 環境によっては、**roxctl** バイナリーを削除するために管理者権限が必要になる場合があります。

2.2.2.2. Linux への roxctl CLI のインストール

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

手順

1. **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.69.2/bin/Linux/roxctl
```

2. **roxctl** バイナリーを実行可能にします。

```
$ chmod +x roxctl
```

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

```
$ echo $PATH
```

検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```

2.2.2.3. macOS への roxctl CLI のインストール

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

手順

1. **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.69.2/bin/Darwin/roxctl
```

2. バイナリーからすべての拡張属性を削除します。

```
$ xattr -c roxctl
```

3. **roxctl** バイナリーを実行可能にします。

```
$ chmod +x roxctl
```

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

```
$ echo $PATH
```

検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```

2.2.2.4. Windows への roxctl CLI のインストール

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

手順

- **roxctl** CLI の最新バージョンをダウンロードします。

```
$ curl -O https://mirror.openshift.com/pub/rhacs/assets/3.69.2/bin/Windows/roxctl.exe
```

検証

- インストールした **roxctl** のバージョンを確認します。

```
$ roxctl version
```

roxctl CLI をアップグレードした後、Scanner をアップグレードできます。

2.2.3. Scanner のアップグレード

roxctl CLI を使用して、Scanner を最新バージョンに更新できます。

前提条件

- プライベートイメージレジストリーからイメージをデプロイする場合は、最初に新しいイメージをプライベートレジストリーにプッシュしてから、このセクションのコマンドのイメージレジストリーを置き換えます。
- Red Hat Advanced Cluster Security for Kubernetes をインストールしたときに Red Hat UBI ベースのイメージを使用した場合は、このセクションのコマンドのイメージ名を次の UBI ベースのイメージ名に置き換えてください。
- カスタムスキャナー設定を作成した場合は、スキャナー設定ファイルを更新する前に、これらの変更を適用する必要があります。

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" scanner generate
```

```
$ oc apply -f scanner-bundle/scanner/02-scanner-03-tls-secret.yaml 1
```

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

```
$ oc apply -f scanner-bundle/scanner/02-scanner-04-scanner-config.yaml 1
```

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

手順

1. Scanner イメージを更新します。

```
$ oc -n stackrox set image deploy/scanner scanner=registry.redhat.io/advanced-cluster-security/rhacs-scanner-rhel8:3.69.2 1
```

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

2. Scanner データベースイメージを更新します。

```
$ oc -n stackrox set image deploy/scanner-db db=registry.redhat.io/advanced-cluster-security/rhacs-scanner-db-rhel8:3.69.2 1
```

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

```
$ oc -n stackrox set image deploy/scanner-db init-db=registry.redhat.io/advanced-cluster-security/rhacs-scanner-db-rhel8:3.69.2 1
```

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

検証

- 新しい Pod が正常にデプロイされたことを確認します。

```
$ oc get pod -n stackrox --watch 1
```

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

2.2.4. Central クラスターのアップグレードの確認

Central と Scanner の両方をアップグレードした後、Central クラスターのアップグレードが完了していることを確認します。

手順

- Central ログを確認します。

```
$ oc logs -n stackrox deploy/central -c central 1
```

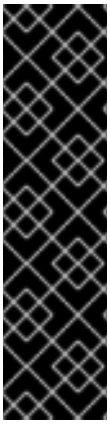
1 Kubernetes を使用する場合は、**oc** の代わりに **kubectl** と入力します。

アップグレードが成功すると、次のような出力が表示されます。

```
No database restore directory found (this is not an error).
Migrator: 2019/10/25 17:58:54: starting DB compaction
Migrator: 2019/10/25 17:58:54: Free fraction of 0.0391 (40960/1048576) is < 0.7500. Will not compact
badger 2019/10/25 17:58:54 INFO: All 1 tables opened in 2ms
badger 2019/10/25 17:58:55 INFO: Replaying file id: 0 at offset: 846357
badger 2019/10/25 17:58:55 INFO: Replay took: 50.324µs
badger 2019/10/25 17:58:55 DEBUG: Value log discard stats empty
Migrator: 2019/10/25 17:58:55: DB is up to date. Nothing to do here.
badger 2019/10/25 17:58:55 INFO: Got compaction priority: {level:0 score:1.73 dropPrefix:[]}
version: 2019/10/25 17:58:55.189866 ensure.go:49: Info: Version found in the DB was current. We're good to go!
```

2.3. 保護されたすべてのクラスターのアップグレード

Central サービスをアップグレードした後、すべての保護されたクラスターをアップグレードする必要があります。



重要

- 自動アップグレードを使用している場合は、以下を行います。
  - 自動アップグレードを使用して、保護されたすべてのクラスターを更新します。
  - このセクションの手順をスキップして、アップグレードの確認 および API トークンの取り消し セクションの手順に従ってください。
- 自動アップグレードを使用していない場合は、Central クラスターを含むすべての保護されたクラスターでこのセクションの手順を実行する必要があります。

Sensor、Collector、および Admission コントローラーを実行しているセキュリティーで保護された各クラスターの手動アップグレードを完了するには、このセクションの手順に従ってください。

2.3.1. 準備状態 (readiness) プローブを更新

Red Hat Advanced Cluster Security for Kubernetes 3.65.0 より前のバージョンからアップグレードする場合は、次の追加コマンドを実行して、readiness プローブパスを更新する必要があります。3.65 よりも新しいバージョンを実行している場合は、この手順をスキップしてください。

手順

- readiness プローブパスを更新します。

```
$ oc -n stackrox patch deploy/sensor -p '{"spec":{"template":{"spec":{"containers":
[{"name":"sensor","readinessProbe":{"httpGet":{"path":"/ready"}}}]}}}}' 1
```

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

2.3.2. OpenShift セキュリティーコンテキスト制約の更新

アップグレードする Red Hat Advanced Cluster Security for Kubernetes のバージョンに応じて、特定の OpenShift Container Platform セキュリティーコンテキスト制約 (SCC) を更新する必要があります。

警告

このセクションのコマンドは、OpenShift Container Platform で Red Hat Advanced Cluster Security for Kubernetes を使用している場合に限り実行してください。それ以外の場合は、このセクションの手順をスキップしてください。

手順

- Red Hat Advanced Cluster Security for Kubernetes 3.64.0 は SCC の名前を変更します。Red Hat Advanced Cluster Security for Kubernetes 3.64.0 より前のバージョンからアップグレードする場合は、SCC を削除して再適用する必要があります。それ以外の場合は、次の手順をスキップしてください。

  a. 次のコマンドを実行して、Central を更新します。

```
$ oc apply -f - <<EOF
kind: SecurityContextConstraints
apiVersion: security.openshift.io/v1
metadata:
  name: stackrox-central
  labels:
    app.kubernetes.io/name: stackrox
  annotations:
    kubernetes.io/description: stackrox-central is the security constraint for the central
server
    email: support@stackrox.com
    owner: stackrox
allowHostDirVolumePlugin: false
allowedCapabilities: []
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: false
allowPrivilegedContainer: false
defaultAddCapabilities: []
fsGroup:
  type: MustRunAs
  ranges:
    - max: 4000
      min: 4000
priority: 0
readOnlyRootFilesystem: true
requiredDropCapabilities: []
runAsUser:
  type: MustRunAs
```

```
  uid: 4000
seLinuxContext:
  type: MustRunAs
seccompProfiles:
  - '*'
users:
  - system:serviceaccount:stackrox:central
volumes:
  - '*'
EOF
```

```
$ oc delete scc central
```

b. 次のコマンドを実行して Scanner を更新します。

```
$ oc apply -f - <<EOF
kind: SecurityContextConstraints
apiVersion: security.openshift.io/v1
metadata:
  name: stackrox-scanner
  labels:
    app.kubernetes.io/name: stackrox
  annotations:
    email: support@stackrox.com
    owner: stackrox
    kubernetes.io/description: stackrox-scanner is the security constraint for the Scanner
container
priority: 0
runAsUser:
  type: RunAsAny
seLinuxContext:
  type: RunAsAny
seccompProfiles:
  - '*'
users:
  - system:serviceaccount:stackrox:scanner
volumes:
  - '*'
allowHostDirVolumePlugin: false
allowedCapabilities: []
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: false
allowPrivilegedContainer: false
defaultAddCapabilities: []
fsGroup:
  type: RunAsAny
readOnlyRootFilesystem: false
requiredDropCapabilities: []
EOF
```

```
$ oc delete scc scanner
```

c. 各 OpenShift Secured Cluster で以下のコマンドを実行します。

```
$ oc apply -f - <<EOF
apiVersion: security.openshift.io/v1
kind: SecurityContextConstraints
metadata:
  name: stackrox-admission-control
  labels:
    app.kubernetes.io/name: stackrox
    auto-upgrade.stackrox.io/component: "sensor"
  annotations:
    email: support@stackrox.com
    owner: stackrox
    kubernetes.io/description: stackrox-admission-control is the security constraint for the
admission controller
users:
  - system:serviceaccount:stackrox:admission-control
priority: 0
runAsUser:
  type: RunAsAny
seLinuxContext:
  type: RunAsAny
seccompProfiles:
  - '*'
supplementalGroups:
  type: RunAsAny
fsGroup:
  type: RunAsAny
groups: []
readOnlyRootFilesystem: true
allowHostDirVolumePlugin: false
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: false
allowPrivilegedContainer: false
allowedCapabilities: []
defaultAddCapabilities: []
requiredDropCapabilities: []
volumes:
  - configMap
  - downwardAPI
  - emptyDir
  - secret
---
apiVersion: security.openshift.io/v1
kind: SecurityContextConstraints
metadata:
  name: stackrox-collector
  labels:
    app.kubernetes.io/name: stackrox
    auto-upgrade.stackrox.io/component: "sensor"
  annotations:
    email: support@stackrox.com
    owner: stackrox
```

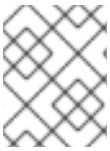
```
    kubernetes.io/description: This SCC is based on privileged, hostaccess, and
hostmount-anyuid
users:
  - system:serviceaccount:stackrox:collector
allowHostDirVolumePlugin: true
allowPrivilegedContainer: true
fsGroup:
  type: RunAsAny
groups: []
priority: 0
readOnlyRootFilesystem: true
runAsUser:
  type: RunAsAny
seLinuxContext:
  type: RunAsAny
seccompProfiles:
  - '*'
supplementalGroups:
  type: RunAsAny
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: true
allowedCapabilities: []
defaultAddCapabilities: []
requiredDropCapabilities: []
volumes:
  - configMap
  - downwardAPI
  - emptyDir
  - hostPath
  - secret
---
apiVersion: security.openshift.io/v1
kind: SecurityContextConstraints
metadata:
  name: stackrox-sensor
  labels:
    app.kubernetes.io/name: stackrox
    auto-upgrade.stackrox.io/component: "sensor"
  annotations:
    email: support@stackrox.com
    owner: stackrox
    kubernetes.io/description: stackrox-sensor is the security constraint for the sensor
users:
  - system:serviceaccount:stackrox:sensor
  - system:serviceaccount:stackrox:sensor-upgrader
priority: 0
runAsUser:
  type: RunAsAny
seLinuxContext:
  type: RunAsAny
seccompProfiles:
  - '*'
supplementalGroups:
```

```
  type: RunAsAny
fsGroup:
  type: RunAsAny
groups: []
readOnlyRootFilesystem: true
allowHostDirVolumePlugin: false
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: true
allowPrivilegedContainer: false
allowedCapabilities: []
defaultAddCapabilities: []
requiredDropCapabilities: []
volumes:
  - configMap
  - downwardAPI
  - emptyDir
  - secret
EOF
```

```
$ oc delete scc admission-control collector sensor
```

2.3.3. その他のイメージの更新

自動アップグレードを使用しない場合は、セキュリティーで保護された各クラスターのセンサー、コレクター、コンプライアンスイメージを更新する必要があります。



注記

Kubernetes を使用している場合は、この手順にリストされているコマンドに **oc** の代わりに **kubectl** を使用してください。

手順

1. Sensor イメージを更新します。

```
$ oc -n stackrox set image deploy/sensor sensor=registry.redhat.io/advanced-cluster-security/rhacs-main-rhel8:3.69.2 1
```

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

2. Compliance イメージを更新します。

```
$ oc -n stackrox set image ds/collector compliance=registry.redhat.io/advanced-cluster-security/rhacs-main-rhel8:3.69.2 1
```

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

3. Collector イメージを更新します。

```
$ oc -n stackrox set image ds/collector collector=registry.redhat.io/advanced-cluster-security/rhacs-collector-rhel8:3.69.2 1
```

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

注記

コレクタスリムイメージを使用している場合は、代わりに次のコマンドを実行します。

```
$ oc -n stackrox set image ds/collector collector=registry.redhat.io/advanced-cluster-security/rhacs-collector-slim-rhel8:{rhacs-version}
```

4. アドミッションコントロールイメージを更新します。

```
$ oc -n stackrox set image deploy/admission-control admission-control=registry.redhat.io/advanced-cluster-security/rhacs-main-rhel8:3.69.2
```

2.3.4. 保護されたクラスターのアップグレードの確認

保護されたクラスターをアップグレードしたら、更新された Pod が機能していることを確認します。

手順

- 新しい Pod がデプロイされていることを確認します。

```
$ oc get deploy,ds -n stackrox -o wide 1
```

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

```
$ oc get pod -n stackrox --watch 1
```

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

2.4. CENTRAL のロールバック

新しいバージョンへのアップグレードが失敗した場合は、以前のバージョンの Central にロールバックできます。

2.4.1. Central を通常どおりロールバック

Red Hat Advanced Cluster Security for Kubernetes のアップグレードが失敗した場合は、以前のバージョンの Central にロールバックできます。

前提条件

- Red Hat Advanced Cluster Security for Kubernetes 3.0.57.0 以降を使用している必要があります。

- ロールバックを実行する前に、永続ストレージで使用可能な空きディスク容量が必要です。Red Hat Advanced Cluster Security for Kubernetes は、ディスク領域を使用して、アップグレード中にデータベースのコピーを保持します。ディスク容量がコピーを保存するのに十分でなく、アップグレードが失敗した場合は、以前のバージョンにロールバックすることはできません。

手順

- アップグレードが失敗した場合 (Central サービスが開始する前) に、次のコマンドを実行して前のバージョンにロールバックします。

```
$ oc -n stackrox rollout undo deploy/central 1
```

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

2.5. アップグレードの確認

更新された Sensor と Collector は、保護された各クラスターからの最新データを引き続き報告します。

Sensor が Central に最後に接続した時刻は、RHACS ポータルに表示されます。

手順

1. RHACS ポータルで、**Platform Configuration → System Health** に移動します。
2. Sensor Upgrade で、Central で最新のクラスターが表示されることを確認してください。

2.6. API トークンを取り消す

セキュリティー上の理由から、Red Hat では、Central データベースのバックアップを完了するために使用した API トークンを取り消すことが推奨されます。

前提条件

- アップグレード後、RHACS ポータルページをリロードし、証明書を再承認して、RHACS ポータルを引き続き使用する必要があります。

手順

1. RHACS ポータルで、**Platform Configuration → Integrations** に移動します。
2. **Authentication Tokens** カテゴリまで下にスクロールし、**API Token** をクリックします。
3. 取り消すトークン名の前にあるチェックボックスを選択します。
4. **Revoke** をクリックします。
5. 確認ダイアログボックスで、**Confirm** をクリックします。