Red Hat OpenShift Service on AWS 4

セキュリティーおよびコンプライアンス

AWS クラスターの Security Context Constraints の設定

Last Updated: 2024-04-19

Red Hat OpenShift Service on AWS 4 セキュリティーおよびコンプライアンス

AWS クラスターの Security Context Constraints の設定

法律上の通知

概要

このガイドでは、セキュリティーコンテキスト制約を設定する手順を説明します。

目次

第1章 監査ログ **3**
1.1. API の監査ログについて 3
1.2. 監査ログの収集 4

第2章 IP ベースの AWS ロールを引き受けるための追加制約を追加する **6**
2.1. アイデンティティベースの IAM ポリシーを作成する 6
2.2. アイデンティティベースの IAM ポリシーのアタッチ 7
2.3. 関連情報 8

第1章 監査ログ

Red Hat OpenShift Service on AWS 監査は、システムに影響を与えた一連のアクティビティーを個別のユーザー、管理者、またはその他システムのコンポーネント別に記述したセキュリティー関連の時系列のレコードを提供します。

1.1. API の監査ログについて

監査は API サーバーレベルで実行され、サーバーに送られるすべての要求をログに記録します。それぞれの監査ログには、以下の情報が含まれます。

表1.1 監査ログフィールド

フィールド	説明
level	イベントが生成された監査レベル。
auditID	要求ごとに生成される一意の監査 ID。
stage	このイベントインスタンスの生成時の要求処理のステージ。
requestURI	クライアントによってサーバーに送信される要求 URI。
verb	要求に関連付けられる Kubernetes の動詞。リソース以外の要求の場合、これは小文字の HTTP メソッドになります。
user	認証されたユーザーの情報。
impersonatedUser	オプション。偽装ユーザーの情報 (要求で別のユーザーを偽装する場合)。
sourceIPs	オプション。要求の送信元および中間プロキシーからのソース IP。
userAgent	オプション。クライアントが報告するユーザーエージェントの文字列。ユーザーエージェントはクライアントによって提供されており、信頼できないことに注意してください。
objectRef	オプション。この要求のターゲットとなっているオブジェクト参照。これは、**List** タイプの要求やリソース以外の要求には適用されません。
responseStatus	オプション。**ResponseObject** が **Status** タイプでなくても設定される応答ステータス。正常な応答の場合、これにはコードのみが含まれます。ステータス以外のタイプのエラー応答の場合、これにはエラーメッセージが自動的に設定されます。

フィールド	説明
requestObject	オプション。JSON 形式の要求からの API オブジェクト。**RequestObject** は、バージョンの変換、デフォルト設定、受付またはマージの前に要求の場合のように記録されます (JSON として再エンコードされる可能性がある)。これは外部のバージョン付けされたオブジェクトタイプであり、それ自体では有効なオブジェクトではない可能性があります。これはリソース以外の要求の場合には省略され、要求レベル以上でのみログに記録されます。
responseObject	オプション。JSON 形式の応答で返される API オブジェクト。**ResponseObject** は外部タイプへの変換後に記録され、JSON としてシリアライズされます。これはリソース以外の要求の場合には省略され、応答レベルでのみログに記録されます。
requestReceivedTimestamp	要求が API サーバーに到達した時間。
stageTimestamp	要求が現在の監査ステージに達した時間。
annotations	オプション。監査イベントと共に保存される構造化されていないキーと値のマップ。これは、認証、認可、受付プラグインなど、要求提供チェーンで呼び出されるプラグインによって設定される可能性があります。これらのアノテーションは監査イベント用のもので、送信されたオブジェクトの **metadata.annotations** に対応しないことに注意してください。キーは、名前の競合が発生しないように通知コンポーネントを一意に識別する必要があります (例: **podsecuritypolicy.admission.k8s.io/policy**)。値は短くする必要があります。アノテーションはメタデータレベルに含まれます。

Kubernetes API サーバーの出力例:


```
{"kind":"Event","apiVersion":"audit.k8s.io/v1","level":"Metadata","auditID":"ad209ce1-fec7-4130-8192-c4cc63f1d8cd","stage":"ResponseComplete","requestURI":"/api/v1/namespaces/openshift-kube-controller-manager/configmaps/cert-recovery-controller-lock?timeout=35s","verb":"update","user":{"username":"system:serviceaccount:openshift-kube-controller-manager:localhost-recovery-client","uid":"dd4997e3-d565-4e37-80f8-7fc122ccd785","groups":["system:serviceaccounts","system:serviceaccounts:openshift-kube-controller-manager","system:authenticated"]},"sourceIPs":["::1"],"userAgent":"cluster-kube-controller-manager-operator/v0.0.0 (linux/amd64) kubernetes/$Format","objectRef":{"resource":"configmaps","namespace":"openshift-kube-controller-manager","name":"cert-recovery-controller-lock","uid":"5c57190b-6993-425d-8101-8337e48c7548","apiVersion":"v1","resourceVersion":"574307"},"responseStatus":{"metadata":{},"code":200},"requestReceivedTimestamp":"2020-04-02T08:27:20.200962Z","stageTimestamp":"2020-04-02T08:27:20.206710Z","annotations":{"authorization.k8s.io/decision":"allow","authorization.k8s.io/reason":"RBAC: allowed by ClusterRoleBinding \"system:openshift:operator:kube-controller-manager-recovery\" of ClusterRole \"cluster-admin\" to ServiceAccount \"localhost-recovery-client/openshift-kube-controller-manager\""}}
```


1.2. 監査ログの収集

must-gather ツールを使用して、クラスターをデバッグするための監査ログを収集できます。このログは、確認したり、Red Hat サポートに送信したりできます。

手順

1. **-- /usr/bin/gather_audit_logs** を指定して **oc adm must-gather** コマンドを実行します。

```
$ oc adm must-gather -- /usr/bin/gather_audit_logs
```

2. 作業ディレクトリーに作成された **must-gather** ディレクトリーから圧縮ファイルを作成します。たとえば、Linux オペレーティングシステムを使用するコンピューターで以下のコマンドを実行します。

```
$ tar cvaf must-gather.tar.gz must-gather.local.472290403699006248 1
```

1 **must-gather-local.472290403699006248** は、実際のディレクトリー名に置き換えます。

3. Red Hat カスタマーポータルの **カスタマーサポート** ページ で、圧縮ファイルをサポートケースに添付します。

第2章 IP ベースの AWS ロールを引き受けるための追加制約を追加する

AWS アカウントに追加のセキュリティーレイヤーを実装して、許可リストに登録されていない IP アドレスからロールを引き受けないようにできます。

2.1. アイデンティティベースの IAM ポリシーを作成する

Red Hat が提供する IP 以外の IP アドレスから要求が発信された場合に、すべての AWS アクションへのアクセスを拒否する、アイデンティティベースの Identity and Access Management (IAM) ポリシーを作成できます。

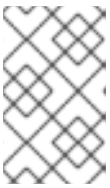
前提条件

- IAM ポリシーの作成および変更に必要な権限を持ち、AWS Management Console にアクセスできる。

手順

1. AWS アカウントの認証情報を使用して AWS マネジメントコンソールにサインインします。
2. IAM サービスに移動します。
3. IAM コンソールで、左側のナビゲーションメニューから **Policies** を選択します。
4. **Create policy** をクリックします。
5. **JSON** タブを選択して、JSON 形式を使用してポリシーを定義します。
6. JSON ポリシードキュメントに入力する必要がある IP アドレスを取得するには、次のコマンドを実行します。

```
$ ocm get /api/clusters_mgmt/v1/trusted_ip_addresses
```



注記

これらの IP アドレスは永続的なものではなく、変更される可能性があります。API 出力を継続的に確認し、JSON ポリシードキュメントで必要な更新を行う必要があります。

7. 次の **policy_document.json** ファイルをコピーし、エディターに貼り付けます。

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Deny",
         "Action": "*",
         "Resource": "*",
         "Condition": {
            "NotIpAddress": {
               "aws:SourceIp": []
            },
```

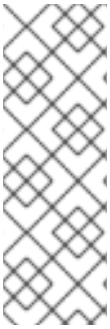
```
            "Bool": {
              "aws:ViaAWSService": "false"
            }
          }
        }
      ]
    }
```

8. 手順 6 で取得したすべての IP アドレスをコピーし、**policy_document.json** ファイルの **"aws:SourceIp": []** 配列に貼り付けます。

9. **Review and Create** をクリックします。

10. ポリシーの名前と説明を入力し、詳細が正確であることを確認します。

11. **Create policy** をクリックしてポリシーを保存します。



注記

最初の呼び出しに基づいて後続の呼び出しが成功するためには、条件キー **aws:ViaAWSService** を false に設定する必要があります。たとえば、**aws ec2 description-instances** への初回呼び出し時に条件キー **aws:ViaAWSService** が false に設定されていない場合、、ec2 インスタンスにアタッチされている EBS ボリュームに関する情報を取得するために AWS API サーバー内で実行される後続の呼び出しはすべて失敗します。後続の呼び出しは、AllowList に含まれない AWS IP アドレスから発信されるため、失敗します。

2.2. アイデンティティベースの IAM ポリシーのアタッチ

アイデンティティベースの IAM ポリシーを作成したら、それを AWS アカウント内の関連する IAM ユーザー、グループ、またはロールにアタッチして、これらのエンティティーが IP ベースのロールを引き受けないようにします。

手順

1. AWS マネジメントコンソールの IAM コンソールに移動します。

2. ポリシーのアタッチ先となるデフォルトの IAM **ManagedOpenShift-Support-Role** ロールを選択します。

注記

デフォルトの IAM **ManagedOpenShift-Support-Role** ロールは変更できます。ロールの詳細は、Red Hat サポートのアクセス を参照してください。

3. **Permissions** タブで、**Add Permissions** ドロップダウンリストから **Add Permissions** または **Create inline policy** を選択します。

4. 以下を実行して、作成したポリシーを検索します。

   a. ポリシー名を入力します。

   b. 適切なカテゴリーでフィルタリングします。

5. ポリシーを選択し、**Attach policy** をクリックします。

重要

IP ベースのロールの引き受けを効果的に防止するには、許可リストに登録された IP を最新の状態に保つ必要があります。これを行わないと、Red Hat サイトリライアビリティーエンジニアリング (SRE) がアカウントにアクセスできなくなり、SLA に影響を与える可能性があります。さらにご質問がある場合、またはサポートが必要な場合は、サポートチームにお問い合わせください。

2.3. 関連情報

- ソース IP に基づきアクセスを拒否する方法の詳細は、AWS ドキュメントの AWS: Denies access to AWS based on the source IP を参照してください。