Red Hat CloudForms 5.0

Integration with OpenShift Container Platform

Adding Red Hat OpenShift Container Platform (with Metrics Enabled) as a Container Provider

Last Updated: 2020-08-05

Red Hat CloudForms 5.0 Integration with OpenShift Container Platform

Adding Red Hat OpenShift Container Platform (with Metrics Enabled) as a Container Provider

Red Hat CloudForms Documentation Team
cloudforms-docs@redhat.com

Legal Notice

Abstract

This document provides a quick guide for integrating Red Hat OpenShift Container Platform container services (with metrics enabled) with Red Hat CloudForms. It is intended as an abridged reference for users already familiar with Red Hat CloudForms, Red Hat OpenShift Container Platform, and Red Hat Enterprise Linux.

Table of Contents

CHAPTER 1. OVERVIEW

This guide walks you through adding a Red Hat OpenShift Container Platform cluster to a Red Hat CloudForms container provider catalog. This deployment focuses on enabling the OpenShift Container Platform cluster metrics plug-in, so that CloudForms can collect information from an OpenShift Container Platform cluster upon integration.

Each procedure in this guide is covered in greater detail in the Red Hat CloudForms and OpenShift Container Platform product documentation. However, links to the corresponding sections are provided for more detail.

The following sections will describe the required configuration for both products prior to integration.

CHAPTER 2. PREREQUISITES

This guide assumes that you have:

- Already deployed Red Hat CloudForms on your chosen platform
- Already deployed OpenShift Container Platform

When enabling metrics on OpenShift Container Platform, you can store your metrics data on *persistent* or *non-persistent* storage. To use persistent storage, you need to provision a persistent volume specifically for this purpose before configuring the metrics components. See Persistent Volumes in the OpenShift Container Platform *Architecture* documentation for more information.

CHAPTER 3. ENABLING OPENSHIFT CONTAINER PLATFORM METRICS

In order for CloudForms to collect OpenShift Container Platform node, pod, and container, you must first enable *cluster metrics* for your OpenShift cluster. This involves running the OpenShift Metrics services inside your cluster. If cluster metrics are already enabled on OpenShift, skip this section and proceed to Chapter 4, *Retrieving the OpenShift Container Platform Management Token* .

> NOTE
>
> This section is an abridged version of a more detailed chapter, namely Enabling Cluster Metrics from the OpenShift Container Platform *Installation and Configuration* documentation. Refer to that chapter for more information.

3.1. CONFIGURING THE REQUIRED SERVICE ACCOUNTS

Cluster metrics requires the following service accounts:

- **metrics-deployer**
- **heapster**

> IMPORTANT
>
> If you deployed OpenShift using **openshift-ansible-3.0.20**, then the service account and roles required for enabling metrics will already be installed. You can skip this section and go to Section 3.2, “Configuring Metrics Components”.

To create these accounts:

1. Log in as an administrator to any node within the OpenShift Container Platform cluster.
2. Open a terminal.
3. Switch to the **openshift-infra** project:

```
$ oc project openshift-infra
```

4. Create a service account for the Metrics Deployer named **metrics-deployer**:

```
$ oc create -f - <<API
  apiVersion: v1
  kind: ServiceAccount
  metadata:
    name: metrics-deployer
  secrets:
  - name: metrics-deployer
  API
```

5. As described in Section 3.2, “Configuring Metrics Components”, the Metrics Deployer uses the **metrics-deployer** service account. Configure the **metrics-deployer** account to have **edit** permissions in the **openshift-infra** project:

```
$ oadm policy add-role-to-user \
   edit system:serviceaccount:openshift-infra:metrics-deployer
```

6. The **heapster** account will be automatically created in Section 3.2.1, “Deploying the Metrics Components”. However, pre-emptively grant it **cluster-reader** permission to the **openshift-infra** project:

```
$ oadm policy add-cluster-role-to-user \
   cluster-reader system:serviceaccount:openshift-infra:heapster
```

3.2. CONFIGURING METRICS COMPONENTS

The Metrics Deployer installs and configures the components required for OpenShift Container Platform metrics. By default, the Metrics Deployer uses *self-signed certificates* to secure communication between components. This document assumes that you will use this default. For information on alternative secure communication configurations, see Using Secrets from the OpenShift Container Platform *Installation and Configuration* documentation.

NOTE

This section is an abridged version of a more detailed chapter, namely Metrics Data Storage in the OpenShift Container Platform *Installation and Configuration* documentation. Refer to that chapter for more information on how to deploy the metrics plug-in using persistent and non-persistent storage.

3.2.1. Deploying the Metrics Components

OpenShift Container Platform uses *Hawkular Metrics* as its metrics engine. The Metrics Deployer will install the Hawkular Metrics service; however, you need to provide the external hostname so that CloudForms can reach the Hawkular Metrics service. The base configuration of the Metrics Deployer is defined in the **/usr/share/openshift/examples/infrastructure-templates/enterprise/metrics-deployer.yaml** file.

Before deploying OpenShift metrics, choose a storage option, then log in as an administrator to any node within the OpenShift Container Platform cluster. From there, open a terminal and run the corresponding command:

Deploying with persistent storage

With *persistent storage*, OpenShift metrics will be stored on a persistent volume. This offers metrics data protection by allowing it to survive a pod recreation or restart. OpenShift metrics requires a specifically configured persistent volume; see Persistent Volumes in the OpenShift Container Platform *Architecture* documentation.

```
$ oc new-app \
   -f /usr/share/openshift/examples/infrastructure-templates/enterprise/metrics-deployer.yaml \
   -p HAWKULAR_METRICS_HOSTNAME=HAWKULARHOST
```

Deploying with non-persistent storage

With *non-persistent storage*, any stored metrics will be deleted when the pod is deleted. While it is much easier to run cluster metrics with non-persistent data, it does come with the risk of permanent data loss. So, while you no longer need to provision and configure a volume to store metric data, this does not offer the same protection as persistent storage.

-

```
$ oc new-app \
   -f /usr/share/openshift/examples/infrastructure-templates/enterprise/metrics-deployer.yaml \
   -p HAWKULAR_METRICS_HOSTNAME=HAWKULARHOST \
   -p USE_PERSISTENT_STORAGE=false
```

For either command, replace *HAWKULARHOST* with the external hostname that CloudForms will use to reach the Hawkular Metrics service. *HAWKULARHOST* must be a fully-qualified domain name.

Either storage method deploys the required metrics components and creates the necessary service accounts. In particular, the metrics components will be configured to also use the specified *HAWKULARHOST* as its public endpoint.

3.2.2. Applying the Hawkular Metrics Settings to OpenShift Container Platform

After deploying the metrics components, configure OpenShift Container Platform to use them:

1. Open the OpenShift Master Configuration file at **/etc/origin/master/master-config.yaml**. Add the **metricsPublicURL** parameter to the **assetConfig** section, specifying the *HAWKULARHOST* you specified in Section 3.2.1, “Deploying the Metrics Components”:

   ```
   assetConfig:
      ...
      metricsPublicURL: "https://HAWKULARHOST/hawkular/metrics"
   ```

2. Restart your OpenShift Container Platform master host:

   ```
   $ sudo systemctl restart atomic-openshift-master
   ```

CHAPTER 4. RETRIEVING THE OPENSHIFT CONTAINER PLATFORM MANAGEMENT TOKEN

After enabling cluster metrics on your OpenShift Container Platform deployment, retrieve the *management token* while you are still logged in to the OpenShift Container Platform host. This will be required later in Chapter 6, *Adding OpenShift Container Platform as a Container Provider* .

For OpenShift Container Platform 3.3 or later

Provide the token needed to add an OpenShift Container Platform 3.3 (or later) provider.

Run the following to obtain the token needed to add an OpenShift Container Platform 3.3 (or later) provider:

```
# oc sa get-token -n management-infra management-admin
eyJhbGciOiJSUzI1NiI...
```

For OpenShift Enterprise 3.2

Provide the token needed to add an OpenShift Enterprise 3.2 provider.

Run the following to obtain the token needed to add an OpenShift Enterprise 3.2 provider:

```
# oc sa get-token -n management-infra management-admin
eyJhbGciOiJSUzI1NiI...
```

For OpenShift Enterprise 3.1

Provide the token needed to add an OpenShift Enterprise 3.1 provider.

Run the following to obtain the token needed to add an OpenShift Enterprise 3.1 provider:

1. Obtain the **management** service account token name:

   ```
   # oc describe sa -n management-infra management-admin
   ...
   Tokens:  management-admin-token-0f3fh
            management-admin-token-q7a87
   ```

2. Select and describe one of the tokens to retrieve the full token output, replacing **management-admin-token-0f3fh** with the name of your token:

   ```
   # oc describe secret -n management-infra management-admin-token-0f3fh
   ...
   Data
   ====
   token:  eyJhbGciOiJSUzI1NiI...
   ```

CHAPTER 5. CONFIGURING RED HAT CLOUDFORMS

Configuring CloudForms involves two steps:

1. Section 5.1, "Configuring CloudForms Capacity and Utilization", and

2. Section 5.2, "Enabling SmartState Analysis"

These steps are required to allow CloudForms to collect metrics from OpenShift Container Platform (Chapter 3, *Enabling OpenShift Container Platform Metrics*) and use them to perform a SmartState analysis. You can choose different servers to perform either function; the following sections assume that you will.

5.1. CONFIGURING CLOUDFORMS CAPACITY AND UTILIZATION

For metrics collection to work properly, you also need to configure Red Hat CloudForms to allow for all three **Capacity & Utilization** server roles, which are available from the settings menu under **Configuration → Server → Server Control**. For more information on capacity and utilization collection, see Assigning the Capacity and Utilization Server Roles in the *Deployment Planning Guide*.

To enable these server roles:

1. Click **Configuration**, then select the server to configure from **Settings → Zone** in the accordion menu on the left.

2. Navigate to the **Server Roles** list in the **Server → Server Control** section. From there, set the required capacity and utilization roles to **ON**, namely:

   a. **Capacity & Utilization Coordinator**

   b. **Capacity & Utilization Data Collector**

   c. **Capacity & Utilization Data Processor**

3. Click **Save**.

Data collection is enabled immediately. However, the first collection begins 5 minutes after the server is started, and every 10 minutes after that. Therefore, the longest the collection takes after enabling the Capacity & Utilization Collector role is 10 minutes. The first collection from a particular provider may take a few minutes since Red Hat CloudForms is gathering data points going one month back in time.

For more information, see Capacity and Utilization Collection in the *Deployment Planning Guide*.

5.2. ENABLING SMARTSTATE ANALYSIS

After enabling the required server roles, enable SmartState analysis. See Smart State Analysis Support from the Support Matrix and Running a SmartState Analysis in the *Managing Providers* guide for more information.

Enabling SmartState analysis is similar to Section 5.1, "Configuring CloudForms Capacity and Utilization", in that the procedure also involves enabling server roles on a specific server. To do so:

1. Click i*Configuration*.

2. Select the server to configure from **Settings → Zone** in the left pane of the appliance.

3. Navigate to the **Server Roles** list in the **Server → Server Control** section. From there, set the appropriate SmartState roles to **ON**. Namely:

    a. **SmartProxy**

    b. **SmartState Analysis**

4. Click **Save**.

CHAPTER 6. ADDING OPENSHIFT CONTAINER PLATFORM AS A CONTAINER PROVIDER

At this point, you should now be ready to add your OpenShift cluster to Red Hat CloudForms as a container provider. To do so, prepare the token you retrieved in Chapter 4, *Retrieving the OpenShift Container Platform Management Token* and follow the procedure below:

1. Navigate to **Compute** → **Containers** → **Providers**.
2. Click ⚙ (**Configuration**), then click ⊕ (**Add a New Containers Provider**).
3. Enter a **Name** for the provider.
4. From the **Type** list, select **OpenShift Container Platform**.
5. Enter the appropriate **Zone** for the provider. If you do not specify a zone, it is set to **default**.
6. From the **Alerts** list, select **Prometheus** to enable external alerts. Selecting **Prometheus** adds an **Alerts** tab to the lower pane to configure the Prometheus service. Alerts are disabled by default.
7. From the **Metrics** list, select **Hawkular** or **Prometheus** to collect capacity and utilization data, or leave as **Disabled**. Selecting **Prometheus** or **Hawkular** adds a **Metrics** tab to the lower pane for further configuration. Metrics are disabled by default.
8. In the **Default** tab, configure the following for the OpenShift provider:

    a. Select a **Security Protocol** method to specify how to authenticate the provider:

    - **SSL**: Authenticate the provider securely using a trusted Certificate Authority. Select this option if the provider has a valid SSL certificate and it is signed by a trusted Certificate Authority. No further configuration is required for this option.
    - **SSL trusting custom CA**: Authenticate the provider with a self-signed certificate. For this option, copy your provider’s CA certificate to the **Trusted CA Certificates** box in PEM format.

        NOTE

        You can obtain your OpenShift Container Platform provider’s CA certificate for all endpoints (default, metrics, alerts) from **/etc/origin/master/ca.crt**. Paste the output (a block of text starting with **-----BEGIN CERTIFICATE-----**) into the **Trusted CA Certificates** field.

    - **SSL without validation**: Authenticate the provider insecurely (not recommended).

    b. Enter the **Hostname** (or IPv4 or IPv6 address) of the provider.

    IMPORTANT

    The **Hostname** must use a unique fully qualified domain name.

    c. Enter the **API Port** of the provider. The default port is **8443**.

d. Enter a token for your provider in the **Token** box.

> NOTE
>
> To obtain a token for your provider, run the **oc get secret** command on your provider; see Obtaining an OpenShift Container Platform Management Token.
>
> For example:
>
> ```
> # oc get secret --namespace management-infra management-admin-token-8ixxs --template='{{index .data "ca.crt"}}' | base64 --decode
> ```

e. Click **Validate** to confirm that Red Hat CloudForms can connect to the OpenShift Container Platform provider.

9. For the **Prometheus** alerts service, add the Prometheus alerts endpoint in the **Alerts** tab:

    a. Select a **Security Protocol** method to specify how to authenticate the service:

    - **SSL**: Authenticate the provider securely using a trusted Certificate Authority. Select this option if the provider has a valid SSL certificate and it is signed by a trusted Certificate Authority. No further configuration is required for this option.

    - **SSL trusting custom CA**: Authenticate the provider with a self-signed certificate. For this option, copy your provider's CA certificate to the **Trusted CA Certificates** box in PEM format.

    - **SSL without validation**: Authenticate the provider insecurely using SSL. (Not recommended)

    b. Enter the **Hostname** (or IPv4 or IPv6 address) or alert **Route**.

    c. Enter the **API Port** if your Prometheus provider uses a non-standard port for access. The default port is **443**.

    d. Click **Validate** to confirm that CloudForms can connect to the alerts service.

10. If you selected a metrics service, configure the service details in the **Metrics** tab:

    a. Select a **Security Protocol** method to specify how to authenticate the service:

    - **SSL**: Authenticate the provider securely using a trusted Certificate Authority. Select this option if the provider has a valid SSL certificate and it is signed by a trusted Certificate Authority. No further configuration is required for this option.

    - **SSL trusting custom CA**: Authenticate the provider with a self-signed certificate. For this option, copy your provider's CA certificate to the **Trusted CA Certificates** box in PEM format.

> NOTE
>
> In OpenShift, the default deployment of the router generates certificates during installation, which can be used with the **SSL trusting custom CA** option. Connecting a Hawkular endpoint with this option requires the CA certificate that the cluster uses for service certificates, which is stored in **/etc/origin/master/service-signer.crt** on the first master in a cluster.

- **SSL without validation**: Authenticate the provider insecurely using SSL. (Not recommended)

b. Enter the **Hostname** (or IPv4 or IPv6 address) of the provider, or use the **Detect** button to find the hostname.

c. Enter the **API Port** if your Hawkular or Prometheus provider uses a non-standard port for access. The default port is **443**.

d. Click **Validate** to confirm that Red Hat CloudForms can connect to the metrics endpoint.

11. Click the **Advanced** tab to add image inspector settings for scanning container images on your provider using OpenSCAP.

> NOTE
>
> - These settings control downloading the image inspector container image from the registry and obtaining the Common Vulnerabilities and Exposures (CVE) information (for effective scanning) via a proxy.
> - CVE URL that CloudForms requires to be open for OpenSCAP scanning: https://www.redhat.com/security/data/metrics/ds/. This information is based on the source code of OpenSCAP.

a. Enter the proxy information for the provider in either **HTTP**, **HTTPS**, or **NO Proxy** depending on your environment.

b. Enter the **Image-Inspector Repository** information. For example, **openshift3/image-inspector**.

c. Enter the **Image-Inspector Registry** information. For example, **registry.access.redhat.com**.

d. Enter the **Image-Inspector Tag** value. A tag is a mark used to differentiate images in a repository, typically by the application version stored in the image.

e. Enter **https://www.redhat.com/security/data/metrics/ds/** in **CVE location**.

12. Click **Add**.

NOTE

You can also set global default image-inspector settings for all OpenShift providers in the advanced settings menu by editing the values under **ems_kubernetes**, instead of setting this for each provider.

For example:

```
:image_inspector_registry: registry.access.redhat.com
:image_inspector_repository: openshift3/image-inspector
```

CHAPTER 7. CONTAINER IMAGE SCANNING

7.1. CONFIGURING IMAGE SCANNING

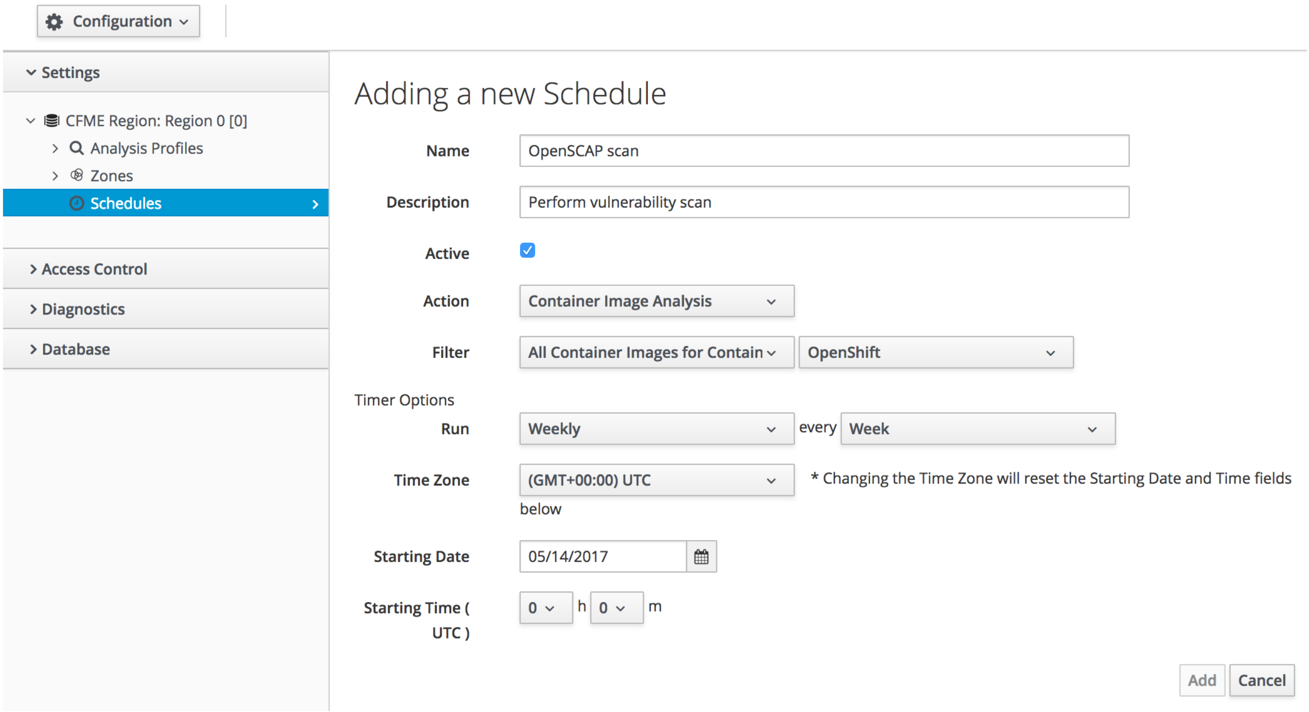
Red Hat CloudForms manages vulnerability scanning of container images. When an OpenShift provider is added, OpenShift images from the internal registry are discovered. To enable image scanning, perform the following configuration steps:

1. Navigate to **Compute** → **Containers** → **Providers**.
2. Select the checkboxes of the OpenShift providers on which to enable scanning.
3. From the **Policy** pull-down menu, click **Manage Policies**.
4. Select the **OpenSCAP profile** checkbox.
5. Click **Save**.

This action will trigger a SmartState analysis, or scan, of all images referenced by the OpenShift provider. The initial scan may take several hours to complete, depending on the number and size of images. The scan occurs in the OpenShift provider, which CloudForms receives and records in the database. OpenShift limits the number of scanning pods; only three images can be scanned simultaneously.

7.2. SCHEDULING A RECURRING SCAN

Software vulnerability databases are updated frequently. To apply these updates, a rescan is required. To schedule a recurring scan of container images:



Adding a new Schedule

Name: OpenSCAP scan

Description: Perform vulnerability scan

Active: ☑

Action: Container Image Analysis

Filter: All Container Images for Contain OpenShift

Timer Options

Run: Weekly every Week

Time Zone: (GMT+00:00) UTC * Changing the Time Zone will reset the Starting Date and Time fields below

Starting Date: 05/14/2017

Starting Time (UTC): 0 h 0 m

1. Click ⚙ (**Configuration**).
2. From **Settings** → **Zones** in the left pane of the appliance, select **Schedules**.
3. From the drop-down menu, click **Configuration** → **Add a new Schedule**.

4. Type an arbitrary **Name**.

5. Type an arbitrary **Description**.

6. Ensure the **Active** checkbox is selected.

7. In **Action**, select **Container Image Analysis**.

8. In **Filter**, select **All Container Images for Containers Provider, OpenShift**.

9. In **Run**, set the schedule as desired.

10. Set the **Time Zone**, **Starting Date**, and **Starting Time**.

11. Click **Add**.

7.3. WORKING WITH IMAGES

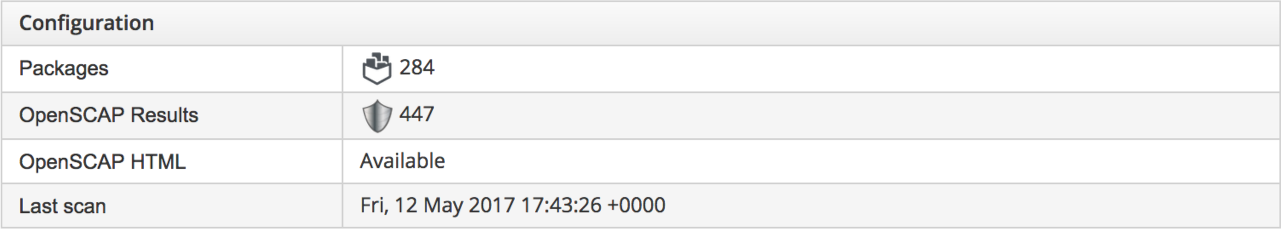
7.3.1. Viewing Results

Image scanning results are displayed in each image summary page.

1. Select **Compute → Containers → Container Images**.

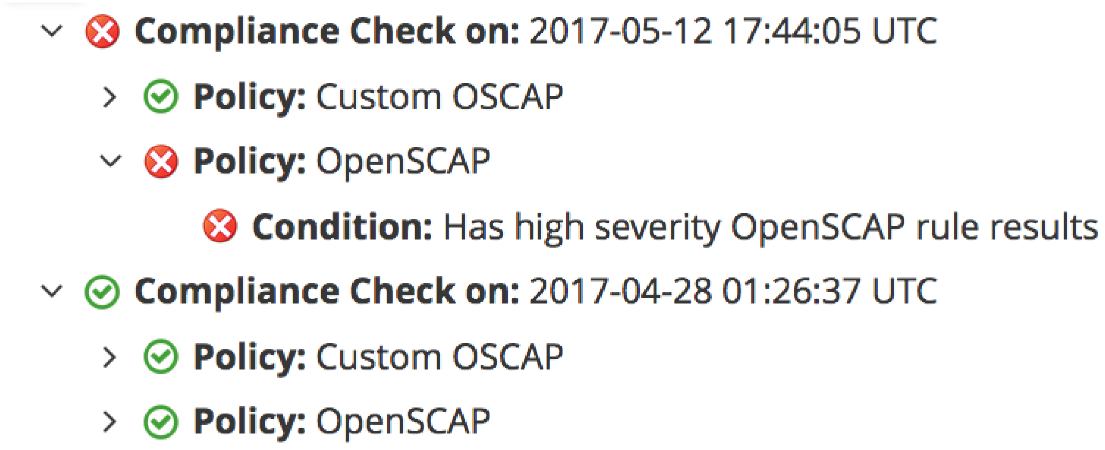
2. Click the desired image.

For an OpenSCAP HTML report, locate the **Configuration** section and select **OpenSCAP HTML**.



Configuration	
Packages	284
OpenSCAP Results	447
OpenSCAP HTML	Available
Last scan	Fri, 12 May 2017 17:43:26 +0000

For compliance and scanning history information, locate the **Compliance** section and note the **Status** field or select **Available** from the **History** field.



- ✗ **Compliance Check on:** 2017-05-12 17:44:05 UTC
  - ✓ **Policy:** Custom OSCAP
  - ✗ **Policy:** OpenSCAP
    - ✗ **Condition:** Has high severity OpenSCAP rule results
- ✓ **Compliance Check on:** 2017-04-28 01:26:37 UTC
  - ✓ **Policy:** Custom OSCAP
  - ✓ **Policy:** OpenSCAP

7.3.2. Manual Scanning

SmartState analysis scanning may be initiated manually for images. From an image summary page, select **Configuration → Perform SmartState Analysis**. Refreshing the image page will reflect the latest scan results and compliance history.

7.3.3. Evaluating Compliance

If the image scan policy has been updated since the last scan, compliance conditions may be re-evaluated. From an image summary page, select **Policy → Check Compliance of Last Known Configuration**. Refreshing the image page will reflect the latest compliance history.

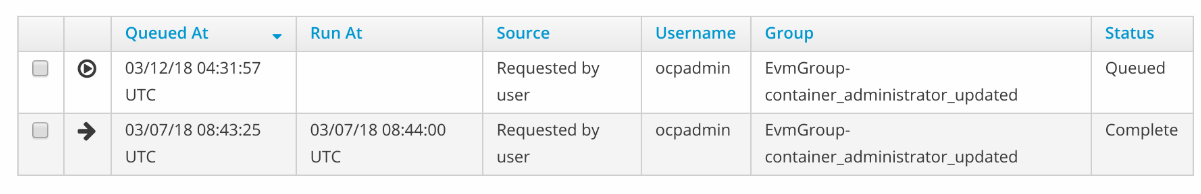
7.3.4. Generating a Report on Images

You can output the results of an OpenSCAP scan of images to a report for an overview of the security risk level of images. The **Images by Failed OpenSCAP Rule Results** is included with CloudForms and shows whether the image has passed or failed OpenSCAP policy criteria, and the security risk.

> **NOTE**
>
> You can also create a copy of this report and edit it to contain additional information, such as the project name where the image is used, to produce more useful results. See Editing a Report and See Reportable Fields in Red Hat CloudForms in *Monitoring, Alerts, and Reporting* for instructions on customizing reports.

To create a report showing image compliance:

1. Navigate to **Overview → Reports**.
2. Click the **Reports → All Reports** accordion.
3. Navigate to **Configuration Management → Containers → Images by Failed OpenSCAP Rule Results** for a report showing which images have failed the OpenSCAP compliance.
4. Click ▶ **Queue**.
5. The report generation is placed in the queue and its status shows in the reports page.



		Queued At	Run At	Source	Username	Group	Status
□	▶	03/12/18 04:31:57 UTC		Requested by user	ocpadmin	EvmGroup-container_administrator_updated	Queued
□	→	03/07/18 08:43:25 UTC	03/07/18 08:44:00 UTC	Requested by user	ocpadmin	EvmGroup-container_administrator_updated	Complete

6. Click ⟳ **(Refresh this page)** to update the status.
7. Navigate to the **Saved Reports** accordion, and click the report when it is completed.
8. Click on the report download buttons for the type of export you want. The report is automatically named with the type of report and date.

- Click 🗎 **(Download this report in text format)** to download as text.

- Click **(Download this report in CSV format)** to download as a comma-separated file.
- Click **(Download this report in PDF format)** to download as PDF.

7.4. OPENSCAP POLICY PROFILE

Red Hat CloudForms is pre-configured with a default scanning policy profile. This includes conditions to scan and identify compliance, as well as annotate compliance failure. SmartState analysis is performed when new images are added to OpenShift.

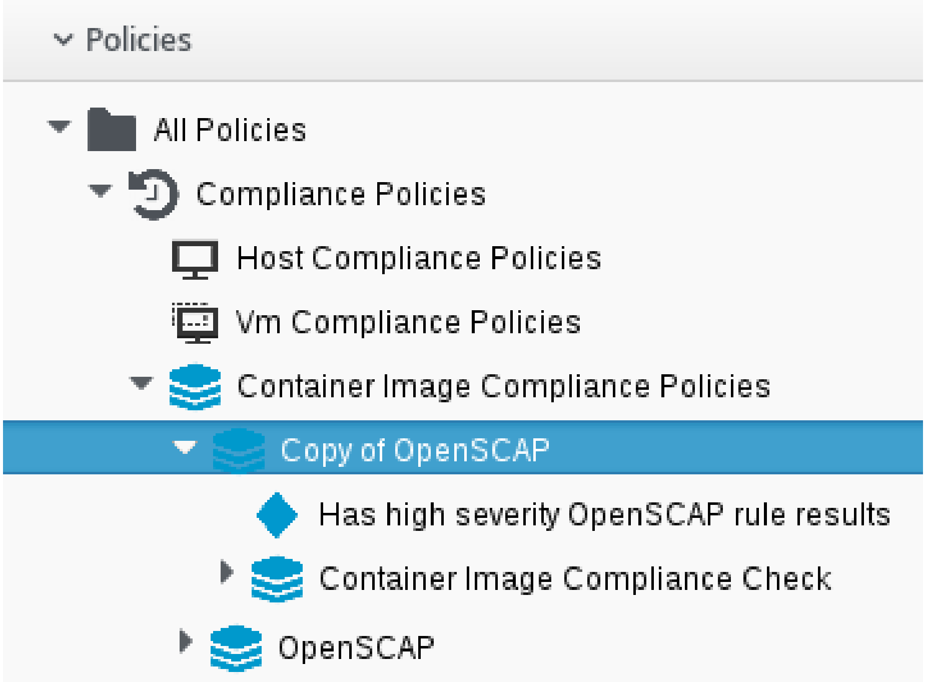
7.4.1. Customizing the Scanning Policy Profile

The built-in OpenSCAP policy profile cannot be edited. You can, however, assign *edited* copies of its policies to a new policy profile. This will allow you to create a customized version of the built-in OpenSCAP policy profile.

To do so, you will first have to copy the policy you want to customize:

1. Navigate to **Control** → **Explorer**.
2. Click the **Policies** accordion, and select **Container Image Compliance Policies**, then click **OpenSCAP**.
3. Click (**Configuration**), and an option to copy the policy should appear; for example, (**Copy this Container Image Policy**).
4. Click **OK** to confirm.

The new policy is created with a prefix of **Copy of** in its description, and it can be viewed in the Policies accordion.



You can now edit the copied policy. After editing copied policies, you can add them to a new policy profile. For instructions on how to edit policies, create a new policy profile, and add policies to it, see the

Policies and Profiles guide. Once you have a customized policy profile, you can assign it to a containers provider.

7.5. CONTROLLING OPENSHIFT POD EXECUTION

Through the default policy profile, non-compliant images receive the control policy action **Mark as Non-Compliant**. This action annotates the **image** object (not to be confused with the **imagestream** object) with *images.openshift.io/deny-execution=true*. This annotation may be used to prevent nodes from running non-compliant images. Refer to the OpenShift Container Platform *Image Policy* documentation for configuration details.

7.6. REFERENCE

More information about OpenSCAP, see visit the OpenSCAP web site.