



# OpenShift Container Platform 4.3

## Getting started with cost management

Learn about and configure cost management



## OpenShift Container Platform 4.3 Getting started with cost management

---

Learn about and configure cost management

## Legal Notice

Copyright © 2020 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## Abstract

This guide describes the initial steps to begin using cost management.

---

## Table of Contents

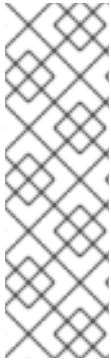
<b>CHAPTER 1. INTRODUCTION TO COST MANAGEMENT</b> .....	<b>3</b>
1.1. ABOUT COST MANAGEMENT	3
1.1.1. Terminology	3
1.2. PLANNING FOR COST MANAGEMENT	4
<b>CHAPTER 2. LIMITING ACCESS TO COST MANAGEMENT RESOURCES</b> .....	<b>5</b>
2.1. ADDING A ROLE	5
2.2. ADDING A ROLE TO A GROUP	6
<b>CHAPTER 3. ADDING SOURCES TO COST MANAGEMENT</b> .....	<b>8</b>
3.1. ADDING AN AMAZON WEB SERVICES (AWS) SOURCE TO COST MANAGEMENT	8
3.1.1. Creating an S3 bucket for reporting	8
3.1.2. Activating AWS tags for cost management	9
3.1.3. Enabling account access for cost and usage consumption	9
3.2. ADDING A MICROSOFT AZURE SOURCE TO COST MANAGEMENT	11
3.2.1. Creating an Azure resource group and storage account	11
3.2.2. Configuring an Azure service principal	12
3.2.3. Configuring a daily Azure data export schedule	13
3.3. ADDING AN OPENSIFT CONTAINER PLATFORM SOURCE TO COST MANAGEMENT	13
3.3.1. Installing the OpenShift Container Platform prerequisites	14
3.3.2. Obtaining Metering Operator login credentials	15
3.3.3. Downloading and configuring the Usage Collector	16
3.3.4. Uploading data with Usage Collector	17
<b>CHAPTER 4. ORGANIZING COST DATA USING TAGS</b> .....	<b>19</b>
4.1. HOW COST MANAGEMENT ASSOCIATES TAGS BETWEEN SOURCES	19
4.2. ADDING TAGS TO AN AWS RESOURCE	19
4.3. ADDING LABELS TO AN OPENSIFT NAMESPACE	20
4.4. NEXT STEPS	20



# CHAPTER 1. INTRODUCTION TO COST MANAGEMENT

This document provides instructions to begin using cost management, including prerequisites and instructions for connecting your cloud environments, and configuring users and permissions.

After completing the setup described in this guide, you will be able to track cost and usage data for your Amazon Web Services (AWS), Microsoft Azure, and OpenShift Container Platform environments.



## NOTE

Cost management is currently Technology Preview. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

For more information about the support scope of Red Hat Technology Preview features, see [Technology Preview Features Support Scope](#).

If you have a suggestion for improving this guide or have found an error, please submit a Bugzilla report at <http://bugzilla.redhat.com> against *Cloud Software Services (cloud.redhat.com)* for the **Cost Management** component.

## 1.1. ABOUT COST MANAGEMENT

Cost management is an OpenShift Container Platform service that enables you to better understand and track costs for clouds and containers. It is based on the upstream project Koku.

You can access the cost management application from <https://cloud.redhat.com/beta/cost-management/>.

Cost management allows you to simplify management of resources and costs across various environments, including:

- Public clouds such as Amazon Web Services (AWS) and Microsoft Azure
- Container platforms such as OpenShift Container Platform

The cost management application allows you to:

- Visualize, understand and analyze the use of resources and costs
- Forecast your future consumption and compare them with budgets
- Optimize resources and consumption
- Identify patterns of usage that should be investigated
- Integrate with third party tools that can benefit from cost and resourcing data

### 1.1.1. Terminology

#### Source

A cloud provider account that is connected to cost management to be monitored, for example, an AWS or Azure account, or an OpenShift Container Platform deployment.

### Organization Administrator

The highest permission level for Red Hat accounts, with full access to content and features. This is the only role that can manage users and control their access and permissions on an account. An account may have multiple Organization Administrators.

See [Roles and Permissions for Red Hat Subscription Management](#) for more details.

## 1.2. PLANNING FOR COST MANAGEMENT

When configuring cost management for your needs, consider the scope of your environments that you want to manage costs for, and the users who will have access to the data.

Some considerations in creating a new Red Hat organization and users for different customer types include:

### Scope:

- Customer company wide
- Customer division or organization wide
- Partner company managing several tenants

### Data:

- How does your business need the data? Do you want information about projects, or users, for example?
- Planning AWS tags to reflect previous use cases.
- Enforcement: Is there any way for you to ensure that the proper tags and metadata are included in each item of the inventory?

Determine the required level of access for your users based on the following predefined roles in cost management:

- Cost Administrator: grants read and write permissions
- Cost Cloud Viewer: grants read permissions on cost reports related to cloud sources
- Cost OpenShift Viewer: grants read permissions on cost reports related to OpenShift sources
- Cost Price List Administrator: grants read and write permissions on price list rates
- Organization Administrator: a Red Hat account user who can manage users and control their access and permissions



## CHAPTER 2. LIMITING ACCESS TO COST MANAGEMENT RESOURCES

You may not want users to have access to all cost data, but instead only data specific to their projects or organization. Using role-based access control, you can limit the visibility of resources involved in cost management reports.

Role-based access control works by organizing users into groups, which can be associated with one or more roles. A role defines a permission and a set of resource definitions.

By default, a user who is not an account administrator will not have access to data, but instead must be granted access to resources. Account administrators can view all data without any further role-based access control configuration.

### Prerequisites

- At least one Red Hat account user with Organization Administrator entitlements. You will use this Red Hat login to look up users, add them to groups, and to assign roles that control visibility to resources.
- Additional Red Hat user(s) without Organization Administrator privileges. See [How To Create and Manage Users](#) in the Customer Portal for details.



### NOTE

To learn more about Red Hat account roles, see [Roles and Permissions for Red Hat Subscription Management](#).


## 2.1. ADDING A ROLE

Create a new role to manage and limit the scope of information that users can see within cost management.

### Prerequisites

- You must be an Account Administrator or a member of a group with the *RBAC Administrator* role to create a role.

### Procedure

1. Click  (**Settings**) to navigate to **User Access Management**.
2. Click the **Roles** tab.
3. Click **Add Role** to open the **Add role** wizard.
4. In the **Name and Description** screen, enter a name for the new role, and optionally, a description. Click **Next**.
5. In the **Permission** screen, specify the Red Hat Cloud Services application you are creating the role for (in this case, cost management) as well as the resource and permission type:
  - a. For **Application**, enter *cost-management*.

- b. For **Resource type**, specify the resource this permission will be used to access from the following list:
    - *aws.account*
    - *azure.subscription\_guid*
    - *openshift.cluster*
    - *openshift.node*
    - *openshift.project*
  - c. For **Permission**, specify *read* as all cost resource data is read-only. For example, to create a role with read-only permissions to AWS account data, set *aws.account* as the **Resource type** and *read* as the **Permission**. In the next step, you can specify the AWS account to apply this role to.
6. In the **Resource definitions** screen, you can provide more details about the resources the permission will be used for. For example, to grant this role access to a specific AWS account, enter the following and click **Add to definitions**:
- **Key:** *aws.account*
    - Options for **Key** are: *aws.account*, *azure.subscription\_guid*, *openshift.cluster*, *openshift.node*, *openshift.project*
  - **Operation:** *equal*
    - Use *equal* if you know the exact value, or *list* to see a list of values that will work for this role.
  - **Value:** Your AWS account number or account alias.
    - This is specific to the resource defined in the **Key** field. Examples include the AWS account ID or alias, Azure subscription ID, OpenShift cluster ID, OpenShift node name, or OpenShift project name.  
You can also enter *\** in this field as a wildcard to create a role that matches everything of the resource type defined in **Key**.
7. Review the details for this role and click **Confirm** to create the role.

Your new role will be listed in the **Roles** tab on the **User Access Management** screen.

### Next steps

- Add this role to a group to provide the role with access to resources.


## 2.2. ADDING A ROLE TO A GROUP

Add your role to a group to manage and limit the scope of information that users in that group can see within cost management.

### Prerequisites

- You must be an Account Administrator or a member of a group with the *RBAC Administrator* role to create a role.

## Procedure

1. Click  (**Settings**) to navigate to **User Access Management**.
2. Click the **Groups** tab.
3. Click **Create a group**.
4. In the **General information** screen, enter a name for the new group, and optionally, a description. Click **Next**.
5. In the **Add members** screen, select the user(s) in your organization to add to the new group. Click **Next**.
6. (Optional) In the **Select roles** screen, select one or more role(s) to add to the group. Default roles available for cost management are:
  - *Cost Administrator* : grants read and write permissions
  - *Cost Cloud Viewer* : grants read permissions on cost reports related to cloud sources
  - *Cost OpenShift Viewer* : grants read permissions on cost reports related to OpenShift sources
  - *Cost Price List Administrator* : grants read and write permissions on price list rates
7. Review the details for this group and click **Confirm** to create the group.

Your new group will be listed in the **Groups** list on the **User Access Management** screen.

To verify your configuration, log out of the cost management application and log back in as a user added to the group.

## CHAPTER 3. ADDING SOURCES TO COST MANAGEMENT

To use cost management to monitor your cloud costs, you must first connect a data source to the cost management application.

Currently, cost management can track costs for Amazon Web Services (AWS), Microsoft Azure, and Red Hat OpenShift Container Platform sources.

### 3.1. ADDING AN AMAZON WEB SERVICES (AWS) SOURCE TO COST MANAGEMENT

To add an AWS account to cost management, you must configure your AWS account to provide metrics, then add your AWS account as a source from the cost management user interface.

This creates a read-only connection to AWS in order to collect cost information hourly in cost management, but does not make any changes to the AWS account.



#### IMPORTANT

You must use an AWS master account for this procedure, as a linked AWS account does not have sufficient access to create billing accounts. After you add the master account as a source, cost management will collect data from any linked accounts as well.

Before you can add your AWS account to cost management as a data source, you must configure the following services on your AWS account to allow cost management access to metrics:

1. An S3 bucket to store cost and usage data reporting for cost management
2. An Identity Access Management (IAM) policy and role for cost management to process the cost and usage data

As you will complete some of the following steps in the AWS console, and some steps in the cost management user interface, keep both applications open in a web browser.

Add your AWS source to cost management from the settings area at <https://cloud.redhat.com/beta/settings/sources/>.



#### NOTE

As non-Red Hat products and documentation can change without notice, instructions for configuring the third-party sources provided in this guide are general and correct at the time of publishing. See the [AWS documentation](#) for the most up-to-date and accurate information.

#### 3.1.1. Creating an S3 bucket for reporting

Cost management requires an Amazon S3 bucket with permissions configured to store billing reports.

Log into your AWS master account to begin configuring cost and usage reporting:

1. In the AWS S3 console, create a new S3 bucket or use an existing bucket. If you are configuring a new S3 bucket, accept the default settings.

2. In the AWS Billing console, create a Cost and Usage report that will be delivered to your S3 bucket. Specify the following values (and accept the defaults for any other values):

Report name: `<any-name>` (note this name as you will use it later)  
 Additional report details: Include resource IDs  
 S3 bucket: `<the S3 bucket you configured previously>`  
 Time granularity: Hourly  
 Enable report data integration for: Amazon Redshift, Amazon QuickSight  
 Report path prefix: (leave blank)



#### NOTE

See the *AWS Billing and Cost Management* documentation for more details on configuration.

3. In the cloud.redhat.com platform, open the **Sources** menu (<https://cloud.redhat.com/beta/settings/sources/>) to begin adding an AWS source to cost management:
  - a. Navigate to **Sources** and click **Add a source** to open the Sources wizard.
  - b. Enter a name for your source and click **Next**.
  - c. Select **Cost Management** as the application and **Amazon Web Services (AWS)** as the source type. Click **Next**.
  - d. Paste the name of your S3 bucket and click **Next**.

### 3.1.2. Activating AWS tags for cost management

To use tags to organize your AWS resources in the cost management application, activate your tags in AWS to allow them to be imported automatically.

#### Procedure

1. In the AWS Billing console:
  - a. Open the *Cost Allocation Tags* section.
  - b. Select the tags you want to use in the cost management application, and click **Activate**.
2. In the cloud.redhat.com **Sources** wizard, click **Next** to move to the next screen.

### 3.1.3. Enabling account access for cost and usage consumption

To provide data within the web interface and API, cost management needs to consume the Cost and Usage reports produced by AWS. For cost management to obtain this data with a minimal amount of access, create an IAM policy and role for cost management to use. This configuration will only provide access to the stored information and nothing else.

To configure account access:

1. From the AWS Identity and Access Management (IAM) console, create a new IAM policy for the S3 bucket you configured previously.

- a. Select the JSON tab and paste the following content in the JSON policy text box:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3:::bucket",
        "arn:aws:s3:::bucket/*"
      ]
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets",
        "iam:ListAccountAliases",
        "s3:HeadBucket",
        "cur:DescribeReportDefinitions",
        "organizations:List*",
        "organizations:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```



#### NOTE

- Including the Action *iam:ListAccountAliases* allows cost management to display the AWS account alias, rather than the account ID.
- (Optional) Including Actions *organization:List\** and *organizations:Describe\** allows cost management to obtain the display names of AWS member accounts if you are using consolidated billing rather than the account ID.

- b. Provide a name for the policy and complete the creation of the policy. Keep the AWS IAM console open as you will need it for the next step.
2. In the cloud.redhat.com **Sources** wizard, click **Next** to move to the next screen.
  3. In the AWS IAM console, create a new IAM role:
    - a. For the type of trusted entity, select **Another AWS account**
    - b. Enter *589173575009* as the Account ID to provide the cost management application with read access to the AWS account cost data.

- c. Attach the IAM policy you just configured.
  - d. Enter a role name (and description if desired) and finish creating the policy.
4. In the cloud.redhat.com **Sources** wizard, click **Next** to move to the next screen.
  5. In the AWS IAM console under Roles, open the summary screen for the role you just created and copy the Role ARN (a string beginning with *arn:aws:*).
  6. In the cloud.redhat.com **Sources** wizard, paste your Role ARN and click **Next**.
  7. Review the details and click **Finish** to add the AWS account to cost management.

Cost management will begin collecting cost and usage data from your master AWS account and any linked AWS accounts.

The data can take a few days to populate before it shows on the cost management dashboard (<https://cloud.redhat.com/beta/cost-management/>).

You have completed adding your AWS account as a source.

You have completed adding your AWS account as a source.

## 3.2. ADDING A MICROSOFT AZURE SOURCE TO COST MANAGEMENT

This section describes how to configure your Microsoft Azure account to allow cost management access.

Configuring your Azure account to be a cost management source requires:

1. Creating a storage account and resource group
2. Setting up an application and service principal for access
3. Scheduling daily cost exports



### NOTE

As non-Red Hat products and documentation can change without notice, instructions for configuring the third-party sources provided in this guide are general and correct at the time of publishing. See the [Microsoft Azure documentation](#) for the most up-to-date and accurate information.

Add your Azure source to cost management from <https://cloud.redhat.com/beta/settings/sources/>.

### 3.2.1. Creating an Azure resource group and storage account

Cost export data is written to a storage account, which exists within a resource group. The resource group must be accessible by cost management in order to read the Azure cost data.

Create a new storage account in Azure to contain the cost data and metrics that cost management will collect. This requires a resource group; Red Hat recommends creating a dedicated resource group for this storage account.

1. In the cloud.redhat.com platform, open the **Sources** menu (<https://cloud.redhat.com/beta/settings/sources/>) to begin adding an Azure source to cost management:
  - a. Navigate to **Sources** and click **Add a source** to open the Sources wizard.
  - b. Enter a name for your source and click **Next**.
  - c. Select **Cost Management** as the application and **Microsoft Azure** as the source type. Click **Next**.
2. Create a resource group and storage account in your Azure account using the instructions in the See [Azure documentation](#) *Create a storage account*.

Make a note of the resource group and storage account. They will be needed in subsequent steps.

1. In the cloud.redhat.com **Sources** wizard, enter the **Resource group name** and **Storage account name** and click **Next**.

### 3.2.2. Configuring an Azure service principal

Configuring an Azure Active Directory application and service principal provides cost management with the necessary (read-only) access to obtain cost data for Azure resources.

To configure this access, see the Azure guide [How to: Use the portal to create an Azure AD application and service principal that can access resources](#), and complete the steps in these sections:

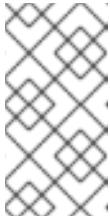
1. *Create an Azure Active Directory application*
2. *Assign the application to a role*
3. *Certificates and secrets*

Any additional details or customization for cost management is described below.

1. In Azure, create a new Azure Active Directory application.
  - Enter a display name of your choice and default values for all other fields.
  - Copy the display name to use in the next step.
2. In Azure, assign the application to a role at the subscription scope.
  - Use the subscription you used to create the storage account.
  - Add a new role assignment with the settings:
    - Role: *Storage Blob Data Reader*
    - Assign access to: Azure AD user, group, or service principal
    - Select: <type the application display name from the previous step>
3. In Azure, create a new application secret.
  - From your application's page, create a new client secret.
  - Enter a description and click **Add**.



- Copy the *Value* for the secret.



#### NOTE

Save your secret to a secure location, as you will need the secret value and application ID to sign in as the application. As the secret value cannot be retrieved again later, if you lose the secret, you will need to create a new one and update it where it has been used.

4. In Azure Cloud Shell, run the following command to obtain your Subscription ID:

```
$ az account show --query "{subscription_id: id}"
```

5. In the cloud.redhat.com **Sources** wizard, enter your **Subscription ID**.

6. In Azure Cloud Shell, run the following command to obtain your tenant ID, client (application) ID, and client secret:

```
$ az ad sp create-for-rbac --query '{"tenant": tenant, "client_id": appld, "secret": password}'
```

7. In the cloud.redhat.com **Sources** wizard, enter your Azure Active Directory **Tenant ID**, **Client ID**, and **Client Secret**, and click **Next**.

### 3.2.3. Configuring a daily Azure data export schedule

Create a recurring task to export your cost data on a daily basis automatically to your Azure storage account, where cost management will retrieve the data.

1. In Azure, add a new export as described in the instructions in the [Azure article](#) *Create and manage exported data*.
  - For **Export type**, select **Daily export of billing-period-to-date costs**
  - For **Storage account**, select the account you created earlier.
  - Enter any value for the container name and directory path for the export. These values provide the tree structure in the storage account where report files are stored.
  - Click **Run now** to start exporting data to the Azure storage container.
2. In the cloud.redhat.com **Sources** wizard, click **Next** when you have created the export schedule and review the source details.
3. Click **Finish** to complete adding the Azure source to cost management.

After the schedule is created, cost management will begin polling Azure for cost data, which will appear on the cost management dashboard (<https://cloud.redhat.com/beta/cost-management/>).

You have completed adding your Azure account as a source.

## 3.3. ADDING AN OPENSIFT CONTAINER PLATFORM SOURCE TO COST MANAGEMENT

This section describes how to connect your OpenShift Container Platform cluster to the cost management application.

To add an OpenShift Container Platform account to cost management, you must configure your OpenShift cluster to provide Operator Metering usage data (metrics), then add the source in the cost management user interface.

For your OpenShift Container Platform cluster to provide metrics to cost management:

1. Install and configure the prerequisites.
2. Obtain a login token for your reporting-operator service account.
3. Download and configure the Usage Collector Ansible playbook to generate report resources for your OpenShift cluster.
4. Create a cron job to regularly collect and send Operator Metering usage data to the upload service.

As you will complete some of the following steps in OpenShift Container Platform, and some steps in the cloud.redhat.com platform (<https://cloud.redhat.com/beta/settings/sources/>), have both applications open in a web browser, as well as a terminal to access the command line interface (CLI).

### 3.3.1. Installing the OpenShift Container Platform prerequisites

Begin adding your OpenShift Container Platform cluster as a source to cost management, then install the prerequisites.

1. In the cloud.redhat.com platform, open the **Sources** menu (<https://cloud.redhat.com/beta/settings/sources/>) to begin adding an AWS source to cost management:
  - a. Navigate to **Sources** and click **Add a source** to open the Sources wizard.
  - b. Enter a name for your source and click **Next**.
  - c. Select **Cost Management** as the application and **OpenShift Container Platform** as the source type. Click **Next**.

Install the prerequisites as described in the Sources wizard.

To obtain the OpenShift Operator Metering usage data, cost management requires the following:

#### On your OpenShift cluster

- OpenShift Container Platform 4.3
- Operator Metering  
Operator Metering is not deployed out of the box in OpenShift Container Platform. See [Installing Metering](#) in the OpenShift documentation for instructions to install metering from OperatorHub.

#### On a Red Hat Enterprise Linux (RHEL) system with network access to your OpenShift cluster

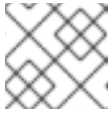
- Red Hat Insights client - installed and registered:

1. Register your RHEL system with Red Hat Subscription Manager and automatically attach subscriptions:

```
# subscription-manager register --auto-attach
```

2. Install the Insights client if using a RHEL 7 system:

```
# yum install -y insights-client
```



#### NOTE

The Insights client is installed by default on Red Hat Enterprise Linux 8.

Register the system to Red Hat Insights:

```
# insights-client --register
```

3. Open the **insights-client.conf** config file, and change the authentication method to **BASIC** and provide the credentials for your Red Hat account. To do this, edit the following lines:

- Uncomment the line **authmethod=BASIC**
- Uncomment and add your Red Hat login username as **username=<yourRedHatlogin>**
- Uncomment and add the password for your Red Hat login as **password=<password>**

```
# sudo vi /etc/insights-client/insights-client.conf
```

Save your changes to the file.



#### NOTE

See [Get Started with Red Hat Insights](#) for more information.

- Ansible (included in the RHEL “extras” repository): <https://access.redhat.com/articles/3174981>  
See [How do I Download and Install Red Hat Ansible Engine?](#) for more information.
- OpenShift command line tools (oc)  
See [Installing the CLI](#) in the *OpenShift documentation*.

When you have finished installing all of the listed prerequisites, check the boxes in the Sources wizard and click **Next**.

### 3.3.2. Obtaining Metering Operator login credentials

During the installation process for the metering operator, the reporting-operator service account is created in the Metering Operator namespace.

1. Log in to your OpenShift cluster with an account that has access to the Metering Operator namespace (for example, *metering*).

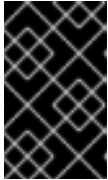
```
# oc login
```

2. Switch to the *metering* project:

```
# oc project metering
```

3. Obtain the login token for the reporting-operator service account and copy the token to a file named *ocp\_usage\_token*:

```
# oc serviceaccounts get-token reporting-operator > ocp_usage_token
```



### IMPORTANT

As the reporting-operator service account token does not expire, store the token file on a file system with limited access to maintain security. This token is used to obtain metering data on a scheduled basis using a cron job.

4. In the cloud.redhat.com **Sources** wizard, click **Next** to move to the next screen.

### 3.3.3. Downloading and configuring the Usage Collector

The Usage Collector connects to your OpenShift cluster and collects usage data required for cost management.

The Usage Collector is an Ansible playbook wrapped in a minimal script that performs setup and collecting tasks. During the setup phase, configuration is stored for connectivity to the OpenShift cluster and usage report resources are created to enable the ongoing collection of usage data.

During the collection phase, usage data is retrieved from the Operator Metering endpoint and compressed into a package that is uploaded for processing by cost management via the Red Hat Insights Client.

1. Download the Usage Collector from <https://github.com/project-koku/korekuta/archive/master.zip> on the same system where the Red Hat Insights Client was installed, using your browser or with the following command:

```
# curl -LOk https://github.com/project-koku/korekuta/archive/master.zip
```

2. Unzip the tool:

```
# unzip master.zip
```

3. Navigate to the *korekuta-master* directory to find the *ocp\_usage.sh* script.

```
# cd korekuta-master
```

4. Run the *ocp\_usage.sh* script with your cluster's variables to set up the Usage Collector.

**NOTE**

To configure the Usage Collector, you need the following information:

- OpenShift API endpoint (e.g. <https://api.openshift-prod.mycompany.com>)
- OpenShift reporting-operator token file path
- OpenShift Operator Metering namespace (e.g. metering)
- The URL of the route exposed for the reporting operator in the Operator Metering namespace (e.g. <https://metering.metering.api.ocp.com>)
- sudo password for installing dependencies

Use the following example, substituting values for your OpenShift API endpoint, Operator Metering namespace, metering-operator token file path, and Metering API endpoint:

```
# ./ocp_usage.sh --setup
-e OCP_API="https://api.openshift-prod.mycompany.com"
-e OCP_METERING_NAMESPACE="openshift-metering"
-e OCP_TOKEN_PATH="/path/to/ocp_usage_token"
-e METERING_API="https://metering.metering.api.ocp.com"
```

**NOTE**

If the `oc` command line is installed in a different location from `/usr/bin/oc` on your system, specify the path using `-e OCP_CLI=</path/to/oc>` when executing the `ocp_usage.sh` command.

You will be prompted for your sudo password and the Ansible playbook will execute to capture the configuration information and create the usage reports on your OpenShift cluster. When complete you will see the following message:

```
TASK [setup : Display New Cluster Identifier] *****
ok: [localhost] => {
  "msg": "Use the following value, <YOUR_OCP_IDENTIFIER>, for the
cluster identifier when configuring an OCP provider in cost management."
}
```

5. Copy the cluster identifier to enter into the [cloud.redhat.com](https://cloud.redhat.com) **Sources** screen and click **Next**.

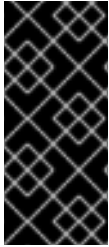
**NOTE**

The cluster identifier value is also stored in `~/.config/ocp_usage/config.json`.

### 3.3.4. Uploading data with Usage Collector

The Usage Collector retrieves usage data from the Operator Metering endpoint and compresses it into a package (an OpenShift Container Platform metrics report) that is uploaded for processing by cost management via the Red Hat Insights Client. Data collection is performed via the `ocp_usage.sh` script.

Create a cron job to regularly upload data collected by the Usage Collector to cost management.



## IMPORTANT

The cron user requires:

- access to the file with metering-operator token
- sudo access to interact with the Red Hat Insights client.

1. To configure sudo access, add the following to the `/etc/sudoers` file, substituting `ocpcollector` with the cron user:

```
ocpcollector  ALL=(ALL)  NOPASSWD: ALL
```

2. Open the crontab for the user that will execute the scheduled upload:

```
# crontab -u <username> -e
```

3. Create a crontab entry to run the Usage Collector every 45 minutes:

```
*/45 * * * * /path/to/ocp_usage.sh --collect --e OCP_CLUSTER_ID=  
<YOUR_OCP_IDENTIFIER>
```

4. In the `cloud.redhat.com` **Sources** wizard, click **Next** to move to the confirmation screen.
5. In the `cloud.redhat.com` **Sources** wizard, review the details and click **Finish** to finish adding the OpenShift Container Platform cluster.

Cost management will begin collecting cost and usage data from your OpenShift Container Platform cluster. The data can take a few days to populate before it shows on the cost management dashboard.

You have completed adding your OpenShift Container Platform cluster as a source.

## CHAPTER 4. ORGANIZING COST DATA USING TAGS

After adding your AWS and OpenShift Container Platform sources, in addition to showing cost data by source, cost management will automatically show AWS cost and usage related to running your OpenShift Container Platform clusters on AWS.

The cost management application tracks these costs using tags (called labels in OpenShift), which you can refine to filter and attribute to resources. Tags in cost management allow you to organize your resources by cost and to allocate the costs to different parts of your cloud infrastructure.



### IMPORTANT

Tags and labels can only be configured directly on a source. You cannot edit tags and labels in the cost management application.

### 4.1. HOW COST MANAGEMENT ASSOCIATES TAGS BETWEEN SOURCES

AWS tags and OpenShift labels both consist of key:value pairs. When the key:value pairs match, the AWS and OpenShift costs are automatically associated by cost management. Note that tag matching is not case sensitive: for example, an AWS resource AWS tagged “APP” and an OpenShift resource tagged “app” are a match.

By default, cost management tracks AWS compute usage and costs by associating the Amazon EC2 instance ID with the OpenShift Container Platform node running on that instance.

You can further organize the information you want to view in cost management by customizing your resource tagging to identify AWS resources associated with an OpenShift cluster running on AWS. For information on the hierarchy of tag matching and tips on planning your tagging strategy, see [Using tagging to manage cost data](#).

To import your tags and labels into cost management, follow the instructions for each source.

### 4.2. ADDING TAGS TO AN AWS RESOURCE

Amazon creates certain tags automatically, like the EC2 instance resource identifier (a number such as i-123456789).

You can also add your own tags at the individual resource level. These tags must be activated for Cost and Usage reporting to export them to the cost management application.

Configure AWS tags for cost management using the following steps:

#### Procedure

1. Create and apply tags to your AWS resources.  
See [User-Defined Cost Allocation Tags](#) in the AWS documentation for instructions.
2. Activate the tags you want to be collected by the cost management application via the Cost and Usage Report. In the AWS Billing console, select the tags you want to activate from the Cost Allocation Tags area.  
See [Activating the AWS-Generated Cost Allocation Tags](#) in the AWS documentation for instructions.

## 4.3. ADDING LABELS TO AN OPENSIFT NAMESPACE

The AWS tag equivalent in OpenShift is a label, which also consists of a key:value pair. Cost management collects OpenShift tag data from pods and persistent volumes (or persistent volume claims) using Prometheus metrics and Operator Metering.

One method of adding tags to OpenShift resources is to specify labels to add in a template.

Or to add labels manually:

### Procedure

To add a label to a namespace in OpenShift:

1. In the OpenShift web console, navigate to Administration > Namespaces.
2. Click your Namespace to open the Namespace overview.
3. From the Namespace Overview page, click Actions > Edit labels to open the dialog box showing labels assigned to your project.
4. Enter your labels, for example: openshift.io/cluster-monitoring=true
5. Click Save.
6. Run the Usage Collector script again to refresh the tags:

```
█ /path/to/ocp_usage.sh --collect
```

Wait an hour for the data to refresh and be visible in cost management.

### Verification steps

To view your tags, navigate to a resource in the OpenShift web console. Any assigned labels are listed under the Labels heading, for example: openshift.io/cluster-monitoring=true.

### Additional resources

Information about creating OpenShift labels can be found here:

- [https://docs.openshift.com/container-platform/4.1/nodes/nodes/nodes-nodes-working.html#nodes-nodes-working-updating\\_nodes-nodes-working](https://docs.openshift.com/container-platform/4.1/nodes/nodes/nodes-nodes-working.html#nodes-nodes-working-updating_nodes-nodes-working)
- [https://docs.okd.io/latest/architecture/core\\_concepts/pods\\_and\\_services.html#labels](https://docs.okd.io/latest/architecture/core_concepts/pods_and_services.html#labels)

## 4.4. NEXT STEPS

- Refine your tags and tagging strategy to better organize your view of cost data. See the [Managing cost data using tagging](#) guide for more details.