Red Hat Insights 1-latest

Deploying and managing RHEL systems in hybrid clouds

Deploying and managing your customized RHEL system images in hybrid clouds

Last Updated: 2024-11-29

Red Hat Insights 1-latest Deploying and managing RHEL systems in hybrid clouds

Deploying and managing your customized RHEL system images in hybrid clouds

Legal Notice

Abstract

Red Hat Insights enables you to perform and manage multiple services as part of one larger, connected workflow. You can define your third-party content source, create customized images, use your preferred system by setting customizations in the image, and launch the image to the target public or private cloud environments, such as Amazon Web Services and Microsoft Azure. You can monitor the system infrastructure you have created, and create and apply patches when needed. You can also report and audit any issues you find in your system infrastructure to improve security and stability.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

CHAPTER 1. WORKFLOW FOR DEPLOYING AND MANAGING RHEL SYSTEMS IN OPEN HYBRID CLOUD ENVIRONMENTS

Use Red Hat Insights to launch and manage your customized RHEL systems images in the open hybrid cloud environments. Currently, you can use Red Hat Insights to deploy and manage the RHEL systems in the following clouds:

- Amazon Web Services
- Microsoft Azure
- Google Cloud Platform

You can access the Red Hat Insights with your Red Hat account, a RHEL subscription, and an activation key. No additional SKUs are required. See Creating an activation key.

As part of the deployment, by using Red Hat Insights, you can follow an end-to-end workflow to:

1. Define or use existing repositories.
2. Configure integrations to launch your images .
3. Build customized images.
4. Launch the images in your preferred cloud .

In addition, you can manage your systems by granting users access, monitoring the inventory of your system infrastructure, and applying patches to improve security and stability.

CHAPTER 2. MANAGING REPOSITORIES TO BUILD YOUR CUSTOMIZED OPERATING SYSTEMS

You can define your customized repositories with third-party content without having to manage their lifecycle. You can use your third-party content to build an image, and when you launch that image to the public cloud environment, you can use those repositories with the **dnf** tool.

2.1. ADDING A CUSTOM REPOSITORY

Define your repository to be able to add packages from this repository to your customized images.

Prerequisites

- You have a RHEL subscription.
- You have administrator access to the Red Hat Hybrid Cloud Console web user interface or **repository administrator** role.
- You have the URL link to your repository content.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Content → Repositories**.
2. Click **Add repositories**. The **Add custom repositories** wizard opens.
3. Optionally: Enable **Snapshot creation** option to create a daily snapshot of this repository. That enables you to create Image Blueprints with the consistent repository content.
4. Enter the following information:
    a. **NAME** - mandatory.
    b. **URL** - mandatory.
    c. In the **Restrict architecture** drop-down menu, select an architecture. You can allow all the architectures or restrict one to your system architecture to prevent incorrect repositories availability.
    d. In the **Restrict OS version** drop-down menu select an operating system (OS). You can allow all the RHEL versions or select one for your system version to prevent incorrect repositories being available.
    e. Optionally: Disable **Modularity filtering** option. When the **Modularity filtering** option is disabled, you can update the packages in this repository even if this package is a part of the module.
    f. **GPG key** - upload the **.txt** file with a GPG key or paste the URL or value of an existing GPG key. The GPG key verifies the signed packages of a repository. If you do not provide the GPG key for a repository, your system cannot perform the verification.
5. If you want to add another repository, click the **+ Add another repository** button and repeat step 3.

6. Click **Save**. The Red Hat Hybrid Cloud Console validates the project status. If your repository is marked as **Invalid**, check the repository URL that you added. For details about the repository status, see Repository status section.

Verification

- Open the list of custom repositories and verify that the repository you added is listed.

2.2. MODIFYING A CUSTOM REPOSITORY

You can modify a custom repository when you need to update information for that repository.

Prerequisites

- You have a RHEL subscription.
- You have administrator access to the Red Hat Hybrid Cloud Console web user interface or **repository administrator** role.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Content → Repositories**.
2. Find a repository you want to modify and click **Edit** in the **Options** menu.
3. In the **Edit custom repository** wizard, modify the information you need. Click **Save changes**.

2.3. REMOVING A CUSTOM REPOSITORY

When you no longer need a custom repository you can delete it.

Prerequisites

- You have a RHEL subscription.
- You have administrator access to the Red Hat Hybrid Cloud Console web user interface or **repository administrator** role.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Content → Repositories**.
2. Find a repository to delete and click **Delete** in the **Options** menu.

Verification

- Open the list of custom repositories, and verify that the repository no longer exists.

2.4. ADDING EXISTING REPOSITORIES FROM POPULAR REPOSITORIES TO CUSTOM REPOSITORIES

The Red Hat Hybrid Cloud Console has pre-configured repositories that you can use to build your customized RHEL image.

Prerequisites

- You have a RHEL subscription.
- You have administrator access to the Red Hat Hybrid Cloud Console web user interface or **repository administrator** role.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Content → Repositories**.
2. On the **Custom repositories** page click the **Popular repositories** tab.
3. Search for the repository you want to add and click **Add**.

Verification

- Select the **Your repositories** tab and verify that the new repository is displayed in the list of custom repositories.

2.5. UPDATING CUSTOM REPOSITORY AFTER CHANGES

When you make changes to your repository you can trigger a refresh of that repository in the Red Hat Hybrid Cloud Console.

Prerequisites

- You have a RHEL subscription.
- You have administrator access to the Red Hat Hybrid Cloud Console web user interface or **repository administrator** role.
- You updated your custom repository.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Content → Repositories**.
2. Find a repository you want to modify and click **Introspect Now** in the **Options** menu.
3. The status of that repository changes to **In progress** that indicates the Hybrid Cloud Console is connecting to the repository and checking for changes.
   The Red Hat Hybrid Cloud Console checks the status of the repositories every 24 hours and again every 8 hours if the status check fails.

2.6. REPOSITORY STATUS IN THE RED HAT HYBRID CLOUD CONSOLE

The repository status shows if the repository is available. The Red Hat Hybrid Cloud Console checks the repository status periodically and can change it. The following table describes the repository status in the Red Hat Hybrid Cloud Console.

Table 2.1. Repository status

Status	Description
Valid	The Red Hat Hybrid Cloud Console has validated the repository and you can use it.
Invalid	The Red Hat Hybrid Cloud Console never validated this repository. You cannot use it.
Unavailable	The repository was valid at least once. The Red Hat Hybrid Console cannot reach this repository at the moment. You cannot use it.
In progress	The repository validation is in progress.

CHAPTER 3. CONFIGURING INTEGRATIONS TO LAUNCH RHEL IMAGES

You can launch your customized RHEL images in a public cloud environment. To do so you must connect your public cloud account with the Hybrid Cloud Console by adding your account to Integrations. While adding your public cloud account, you may need to log in to that account and perform some actions depending on your cloud provider.

3.1. CONNECTING AN AWS ACCOUNT TO THE RED HAT HYBRID CLOUD CONSOLE

You can add your Amazon Web Services (AWS) account to the Red Hat Hybrid Cloud and configure it to launch your RHEL images in the AWS environment.

Prerequisites

- You have a RHEL subscription.
- You are an Organization Administrator or you have a non-admin user with the "Launch administrator" and the "Cloud administrator" roles assigned. See how to assign a role to a user .
- Optional: If you follow automatic access configuration, you have the **Access key ID** and the **Secret access key** for your AWS account.
- Optional: If you follow manual access configuration, ensure you have the following permissions for your AWS account:
  - **cloudformation:CreateStack**;
  - **cloudformation:DescribeStacks**;
  - **cloudformation:DeleteStack**;
  - **cloudformation:UpdateStack**;
  - **iam:CreateRole**;
  - **iam:PutRolePolicy**;
  - **iam:AttachRolePolicy**;
  - **iam:PassRole**;
  - **iam:GetRole**;
  - **iam:DeleteRole**;
  - **iam:ListRolePolicies**;
  - **iam:GetRolePolicy**;
  - **iam:DeleteRolePolicy**.
    On the information about AWS permissions, see the AWS documentation.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations**.
2. Click **Create Integration** and choose the **Cloud** option. The **Add a cloud integration** wizard opens.
3. On the **Select cloud provider** page, select **Amazon Web Service** and click **Next**.
4. On the **Name integration** page, name the integration for your AWS account in the **Integration name** field and click **Next**.
5. On the **Select configuration** page, choose between the following two options:
    a. If you select **Account authorization**, provide your **Access key ID** and **Secret access key** for your ID from your AWS account. Click **Next** and complete the following steps:
        i. On the **Select application** page, select the **Launch images** option. Click **Next**.
    b. If you select **Manual configuration**, click **Next** and complete the following steps:
        i. On the **Select application** page, select the **Launch images** option. Click **Next**.
        ii. On the **Enable account access** page in the **AWS account number**, provide your AWS account number in the **Account number** field and click **Next**.
        iii. On the **Create IAM role** page, follow the instructions on the wizard window. Click **Next**.
        iv. On the **Enter ARN** page, paste the ARN into the text field. Click **Next**.
6. On the **Review details** page, verify the details about your AWS account and click **Add** to finish the AWS integration process.

3.2. CONNECTING MICROSOFT AZURE ACCOUNT TO THE RED HAT HYBRID CLOUD CONSOLE

You can add your Microsoft Azure account to the Red Hat Hybrid Cloud Console and configure it to launch your RHEL images in the Azure environment.

Prerequisites

- You have a RHEL subscription.
- You are an Organization Administrator or you have a non-admin user with the "Launch administrator" and the "Cloud administrator" roles assigned. See how to assign a role to a user .
- You have a Microsoft Azure account.
- You have registered the following resource providers in your Microsoft Azure subscription:
    - **Microsoft.Compute**;
    - **Microsoft.Storage**;
    - **Microsoft.Network**.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations**.

2. Click **Create Integration** and choose the **Cloud** option. The **Add a cloud integration** wizard opens.
3. Select **Microsoft Azure** in the **Select integration type** page and click **Next**.
4. On the **Name integration** page, name the integration for your Microsoft Azure account in the **Integration name** field and click **Next**.
5. On the **Select application** page, select the **Launch images** option and click **Next**.
6. On the **Configure Azure Lighthouse** page, click **Take me to Lighthouse** and complete configuration steps in Azure Lighthouse according to the Microsoft instructions. Return to the **Add integration** wizard and click **Next**.
7. On the **Set subscription ID** page, fill in the **Subscription ID** field and click **Next**.
8. On the **Review details** page, verify the details about your Microsoft Azure account and click **Add** to finish adding it to the **Integrations**.

3.3. CONNECTING GCP PROJECT TO THE RED HAT HYBRID CLOUD CONSOLE

You can add your Google Cloud Platform (GCP) project to the Red Hat Hybrid Cloud and configure it to launch your RHEL images in the AWS environment.

Prerequisites

- You have a RHEL subscription.
- You are an Organization Administrator or you have a non-admin user with the "Launch administrator" and the "Cloud administrator" roles assigned. See how to assign a role to a user .
- You have a GCP project with a **default** network.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations**.
2. Click **Create Integration** and choose the **Cloud** option. The **Add a cloud integration** wizard opens.
3. Select **Google Cloud** in the **Select integration type** page and click **Next**.
4. On the **Integration name** page, name the integration for your GCP project in the **Name** field and click **Next**.
5. On the **Select application** page, select the **Launch images** option and click **Next**.
6. On the **Enable account access** page:
   a. On the **Enter Project ID** page, fill in your GCP project name that you want to add in the **Project** field. Click **Next**.
   b. On the **Create custom role** page, follow the instructions on the page. Click **Next**.

7. On the **Review details** page, verify the details about your GCP project and click **Add** to finish adding it to the **Integrations**.

Verification

- The console validates the data for your GCP project and shows a message **Configuration successful**.

CHAPTER 4. INTEGRATION WITH EXTERNAL AUTHENTICATION AND AUTHORIZATION DOMAINS

The Directory and Domain Services feature brings an additional security level by joining the identity and access management systems of your organization with the Red Hat Hybrid Cloud Console. You can register your existing identity provider domain, such as Red Hat Identity Management (IdM).

IMPORTANT

The Directory and Domain Services feature is only available in the **Preview** mode.

4.1. SECURITY CONSIDERATIONS FOR DIRECTORY AND DOMAIN SERVICES

To register an identity domain of your organization in the Red Hat Hybrid Cloud Console and enroll the machines in it, you must open ports for the required services on the server where your identity domain is deployed.

For example, to ensure your machines have access from the public cloud environment to your IdM server, you must configure access to your IdM server for the following services:

HTTPS

Allows the Directory and Domain Service to use the certificate from the RHEL subscription to enroll the image in the IdM server using IPA API.

Kerberos

Allows users and hosts to authenticate with the Kerberos authentication method.

LDAP

Allows SSSD to retrieve security policies and user information from the IdM server.

The following ports need to be open in order to provide the access to the services.

Table 4.1. IdM ports

Service	Ports	Protocol
HTTP/HTTPS	80, 443	TCP
LDAP/LDAPS	389, 636	TCP
Kerberos	88, 464	TCP and UDP
DNS	53	TCP and UDP (optional)

By using these protocols, you allow access to your organization from every enrolled machine that runs in a public cloud environment. Make sure your company security policies allow it.

Additional resources

- For more details about ports to access the IdM server, see the Port requirements for IdM.

4.2. REGISTERING AN IDENTITY DOMAIN WITH THE HYBRID CLOUD CONSOLE

You can register the identity domain of your organization in the Red Hat Hybrid Cloud Console. It enables you to use your existing identity domain with the new instances running from images in the Hybrid Cloud Console.

Currently, the Directory and Domain Services only support an IdM setup. You can only enable one domain at a time. When you enable a domain, you must disable all other domains in the Red Hat Hybrid Cloud Console.

Prerequisites

- You have the **ipa-hcc-server** package installed from the **EPEL** repository on the existing IdM server.
- You have IPA administrator permissions.
- You have Organization Administrator permissions or you have a user with the **Directory and Domain Services administrator** role. For more details, see Procedures for configuring User Access in Red Hat Hybrid Cloud Console.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations** and from the navigation panel choose **Directory and Domain Services**. The Directory and Domain Services dashboard opens.
2. Click **Register identity domain** to open the **Register identity domain** wizard.
3. Optional: On the **Preparation** page, the wizard prompts you to verify the **ipa-hcc-server** package is installed on your IdM server. Follow the instructions on the page and click **Next**.
4. On the **Registration** page.
   a. Copy the command for registration of your domain, switch to the terminal of your IdM server and run this command under the **root** privileges:

```
# ipa-hcc register <registration token>

Domain information:
   realm name:    <REALM_NAME>
   domain name:   <domain_name>
   dns domains:   <dns_domains>
```

   b. To continue registering your IdM server, type **Yes**:

```
Proceed with registration? Yes/No (default No): <Yes>
```

5. Once the registration command is complete in your IdM server terminal, switch back to the **Register identity domain** wizard and on the **Registration** page click **Test again** to verify registration. Wait for the wizard to verify your registration and click **Next**.
6. On the **Details** page, you can customize the **Display name** field for your domain. Optionally, enter the description for this domain and leave the **Domain auto-join on launch** toggle enabled if you want to make your domain available during launching images in a public cloud environment

after you complete the registration. Click **Next**.

7. On the **Review** page, review all your settings and click **Finish** to complete the registration.

Verification

- Confirm that your domain appears on the **Directory and Domain Services** dashboard.

Next steps

- You can enroll your machines to the registered domain during the launch to the environment of your choice. For that, ensure you add the **ipa-hcc-client** package from the **EPEL** repository during the blueprint creation in **Images**.

Additional resources

- Learn more about Identity Management.
- Learn more about Creating blueprints and blueprint images .
- Learn more about Adding existing repositories from popular repositories to custom repositories .
- Learn more about Launching customized RHEL images to the cloud platforms with Insights image builder.

4.3. EDITING IDENTITY DOMAIN REGISTRATIONS

You can rename and edit the description of the registered domain. You can also enable or disable the auto-join on launch feature for the registered domain.

Prerequisites

- You have Organization Administrator permissions or you have a user with the Directory and Domain Services administrator role.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations** and from the navigation panel choose **Directory and Domain Services**. The Directory and Domain Services dashboard opens.
2. From the list of domains locate the domain you want to edit, click the **Option** menu, and choose **Edit**.
3. You can edit the following parameters:

   Display name

   Changes the name of your domain.

   Description

   Changes the description of your domain.

   Domain auto-join on launch

   Enables or disables this domain for enrolling the instances during the launch to the public cloud providers.

4.4. REMOVING AUTHENTICATION DOMAIN REGISTRATION FROM HYBRID CLOUD CONSOLE

You can remove the registration of your external authentication domain from the Red Hat Hybrid Cloud Console.

Prerequisites

- You have Organization Administrator permissions or you have a user with the Directory and Domain Services administrator role.

Procedure

1. Access Hybrid Cloud Console, click **Settings → Integrations** and from the navigation panel choose **Directory and Domain Services**. The Directory and Domain Services dashboard opens.
2. From the list of domains locate the domain you want to remove, click the **Option** menu, and choose **Delete**. The **Delete identity domain registration** window opens.
3. Select the **I understand that this action cannot be undone** checkbox and click **Delete**.
4. Optional: Access your IdM server terminal and uninstall the **ipa-hcc-server** package:

   ```
   # dnf remove ipa-hcc-server
   ```

   If your IdM deployment consists of multiple servers, remove the **ipa-hcc-server** package from all of the servers.

Verification

- Open the Directory and Domain Services dashboard and verify the registration of your external authentication domain is not in the list.

CHAPTER 5. CREATING BLUEPRINTS AND BLUEPRINT IMAGES

An image blueprint is a persistent definition of the required image customizations. that enables you to create multiple builds from it, without having to configure the blueprint for each image build. You can edit, rebuild, delete, and save a blueprint to keep rebuilding images from it. You can define and manage, by editing or deleting a blueprint, and to keep rebuilding it, removing the need to configure the build each time. When you rebuild a blueprint, all targets specified in the blueprint are rebuilt. The blueprint groups the images that were built from it, so that you can have access to all the related images when dealing with large amounts of images.

The blueprints are persistent and you can manage their customizations. Even if the resulting builds, and images from those builds have different content versions, the customizations stored in that blueprint are always applied across all builds resulting from this blueprint.

When creating an image from the blueprint, unless you define a name to the image, it takes the name of the parent blueprint.

5.1. CREATING A BLUEPRINT

You can create a blueprint with a name, specify the packages that you want to install and define other customizations. You can build images from this blueprint, and the resulting images have all customizations that you specified in this blueprint.

Prerequisites

- You have a RHEL subscription.
- You have Organization Administrator permissions.
- Optional: You must have access to Integrations.
- Optional: If you want to launch your images to the public clouds, you must have your public cloud connected with the Red Hat Hybrid Cloud Console. For details, see Configuring cloud integrations for Red Hat services.
- Optional: If you plan to launch images to a public cloud environment, such as AWS, Microsoft Azure, or Google Cloud Platform, you must configure your account to **Integrations**. See Configuring integrations to launch RHEL images .
- Optional: You have the Activation key for the RHEL system. For details, see Creating an activation key.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights Images dashboard appears.
2. Click **Create blueprint**. The wizard for the blueprint creation opens.
3. On the **Image output** page, select the following options and click **Next**:
   a. From the **Release** list, select the release that you want to use.
   b. From the **Architecture** list, select the architecture that you want to use.

c. From the **Select target environments** options, select the environments that you want to use.

4. Optional: On the **Register** page, the **"Automatically register and enable advanced capabilities"** is enabled by default and the activation key drop down is automatically populated. The **Next** button gets temporarily disabled when the activation key is loading. You can disable the automatic registration by clearing the checkbox if you prefer to register your system during initial boot.

   - If you have previously added a key during recent blueprint creation, the same key gets automatically selected.
   - If you have activation keys, but have not used any key previously, the first activation key will get selected.
   - If you have no activation keys, select **"Automatically create and select a default key"**to automatically create and use an **activation-key-default-*<ID>*** default activation key.
   - You can also select any of the available activation keys.
     The following steps are optional, and if you do not want to customize the image further, you can click the **Review and finish** button to finish the image creation process. You will be redirected to the **Details** step.

5. Optional: On the **OpenSCAP profile** page, select one of the options available and click **Next**.

6. Optional: On the **File system configuration** page, select one of the following options and click **Next**:

   - **Use automatic partitioning**: The recommended partitioning, depending on the target environment.
   - **Manually configure partitions** Use to manually configure the image file system partitions, by following the steps:

     a. Click the **Manually configure partitioning** radio button to show the **Configure partitions** section and configure the partitions based on Red Hat standards and security guides.

        i. From the drop-down menu, provide details to configure the partitions:

        ii. For the **Mount point** field, select one mount point type option
        You can also add an additional path to the **Mount point**, such as **/tmp**. For example: **/var** as a prefix and **/tmp** as an additional path results in **/var/tmp**.

        NOTE

        Depending on the **Mount point** type you choose, the file system type changes, for example to **xfs**, or other type.

        iii. For the **Minimum size** partition field of the file system, enter the desired minimum partition size. In the **Minimum size** drop-down menu, you can use common size units such as **GiB**, **MiB**, or **KiB**. The default unit is **GiB**.

        iv. To add more partitions, click **Add partition** and repeat the steps for each added partition.

7. Optional: On the **Repository snapshot** page, choose one of the following options and click **Next**. Note that this step is available in the **Preview** mode only.
   - Use latest content:: the image uses the latest state of the repository.
   - Use a snapshot:: The image selects a specific date of snapshot creation. If you choose the **Use a snapshot** option, the images will have the static state of the repository based on a date you specify.
8. Optional: On the **Custom repositories** page, select the custom repositories from which you can add packages to the image and click **Next**.
9. Optional: On the **Additional Red Hat packages** page, search for the packages with the search field and choose the packages you need. Click **Next**.
10. Optional: On the **First boot configuration** page, include a file with an action script or write it in the **</> SHELL** field. That script is executed during the first boot of this image. It is useful when you want to enable your custom services or run an Ansible playbook at the boot time of the image. Click **Next**.
    You must start your script with a shebang, for example the **#!/bin/sh** for Bash shell. The first boot feature supports Python, Bash and YAML syntax.
11. On the **Details** page, the blueprint name is prefilled in the "*<distribution>-<architecture>-<datetimeString>*" format. You can enter a name for the blueprint, and the system checks for validity and duplicity against already existing blueprint names. Click **Next**.
12. On the **Review** page, verify the image details about the image creation and from the **Create blueprint** drop-down menu, choose one of the following options:

    Create blueprint

    Creates the blueprint and saves the customizations for your blueprint.

    Create blueprint and build image

    Create the blueprint, save the customizations for your blueprint and build images for the target environment or environments that you choose.
    The system verifies the build manifest of the image. After it reaches 100%, the image appears in the build queue.

Insights Images service starts to compose a RHEL image for the selected architecture. After the image build is ready, you can see the images related to the parent blueprint in the **Images** dashboard.

5.2. EDITING AN EXISTING BLUEPRINT

You can edit a blueprint. For example, to include an extra package. After you finish the blueprint editing, all the images related to the parent blueprint are rebuilt and updated with the new package.

Prerequisites

- You have created a blueprint.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights Images dashboard appears.

2. Search for the blueprint that you want to edit. You can use the **Search** field to search for the blueprint name.
3. Click the blueprint that you want to edit.
4. Click **Edit blueprint**. You are redirected to the **Review** page.
5. From the navigation menu, select the section that you want to edit.
6. After making the changes, click the **Review** page.
7. Click **Save changes to the blueprint**.
   The images related to the parent blueprint that you modified will be rebuilt and updated with the new changes.

5.3. REBUILDING AN EXISTING BLUEPRINT

Every time that you edit your blueprint, it creates a new version of that blueprint. It also impacts the images that are related to that blueprint, making them out of sync with the blueprint. To fix this, and ensure that you have the least updates available on your parent blueprint and the related images, you can edit your blueprint and rebuild it with the latest updates. This action updates all the packages specified in the blueprint and rebuild the related images with the updated packages.

Prerequisites

- You have created a blueprint.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights Images dashboard appears.
2. Search for the blueprint that you want to edit. You can use the **Search** field to search for the blueprint name.
3. Click the blueprint that you want to edit.
4. Click **Edit blueprint**. You are redirected to the **Review** page.
5. Make the changes to the blueprint and select one of the options:

   Save

   Save the customizations for your blueprint.

   Save and build image

   Save the customizations for your blueprint and build images from the target environments that you chose.

5.4. DOWNLOADING A BLUEPRINT

You can export a blueprint that you created in the Hybrid Cloud Console by downloading it in the JSON format.

WARNING

The ability to download a blueprint is only available in the "Preview" mode.

Prerequisites

- You have created a blueprint.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights Images dashboard appears.
2. Search for the blueprint that you want to download. You can use the Search field to search for the blueprint name.
3. Click the blueprint that you want to download.
4. From the **More options** menu, select the "Download blueprint" option.
   The blueprint is saved as a file in the **.json** format to the local storage that you define in your web browser.

5.5. DELETING A BLUEPRINT

If you no longer need a blueprint, you can delete it. All the images related to this blueprint will also be deleted.

Prerequisites

- You have created a blueprint.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights Images dashboard appears.
2. Search for the blueprint that you want to delete. You can use the Search field to search for the blueprint name.
3. Click the blueprint that you want to delete.
4. From the **More options** menu, select the "Delete blueprint" option.
5. Confirm the deletion by clicking **Delete**.

CHAPTER 6. BUILDING AND MANAGING CUSTOMIZED RHEL IMAGES

You can use a blueprint to build customized RHEL images for a variety of deployment types by using Insights Images. You can build Conventional (RPM-DNF) images or Immutable (OSTree) images.

- You can only launch customized images directly from the Hybrid Cloud Console to the **AWS**, **GCP**, and **Microsoft Azure** public clouds.
- The VMDK customized images must first be uploaded to VMWare vSphere, deployed there, and then you can create a VM.
- For the Guest image (**.qcow2**), and Installer (**.iso**), you can download these images and deploy them directly to virtual machines.

WARNING

Red Hat Hybrid Cloud Console does not support uploading **Amazon Web Services (AWS)**, **Google Cloud Platform (GCP), and *Microsoft Azure**images to GovCloud regions.

6.1. ABOUT BUILDING CUSTOMIZED IMAGES

You can build either Conventional (RPM-DNF) images or Immutable (OStree) images from a blueprint.

- The **Conventional (RPM-DNF)** enables you to manage or modify the system software by using the DNF package manager and updated RPM packages.
- The **Immutable (OStree)** images contain a complete operating system ready to be remotely installed and allows you to manage the system software by referencing a central image repository. For more details, see Create RHEL for Edge images and configure automated management.

The image artifacts are saved for 14 days and expire after that. To avoid losing the image, transfer the image to your account before the expiration date. If an image has already expired, you can also re-create the exact image based on an existing blueprint to reuse the previous configuration.

You can share an existing AWS image to a new region to run on your AWS account so that all regions can launch with the same configuration.

You can also download the compose request of your image and use the image builder API to automate your image building tasks.

6.2. BUILDING CUSTOMIZED RHEL SYSTEM IMAGE

Create customized RHEL system images from a blueprint by using Insights Images, and deploy the images on your target environment.

Prerequisites

- You have created a blueprint. See Creating blueprints and blueprint images .

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The list of existing blueprints appears.
2. Select the blueprint that you want to build an image from.
   a. Optionally, you can select the blueprint version from the dropdown menu.
3. Click **Build images**. A pop-up alert informs that the image is being built.
   After the image process status is marked as **Ready**, you can use it in your deployments.

6.3. ACCESSING YOUR CUSTOMIZED RHEL IMAGE FOR AWS FROM YOUR ACCOUNT

After you built your image, uploaded it to AWS, and the cloud registration process status is marked as **Ready**, you can access the image that you created and shared with your **AWS EC2** account.

The shared image expire within 14 days. To permanently access your image, copy the image to your own AWS account.

Prerequisites

- You have access to your AWS Management Console.

Procedure

1. Access your AWS account and navigate to **Service → EC2**.
2. In the navigation bar, verify if you are under the correct region: **us-east-1**.
3. Click **Images**, and choose **AMIs**. The dashboard with the **Owned by me** images opens.
4. From the drop-down menu, choose **Private images**.
   You can see the image successfully shared with the **AWS** account you specified.

6.4. LAUNCHING YOUR CUSTOMIZED RHEL SYSTEM IMAGE FOR AWS FROM YOUR AWS EC2

Launch the image that you shared with your **AWS** account to the **Amazon Elastic Compute Cloud(Amazon EC2)** compute platform.

Prerequisites

- You have access to your customized image on **AWS**. See Accessing your customized RHEL system image for AWS from your account.

Procedure

1. From the drop-down menu, under **Private images**, locate the image that you shared to the AWS account you specified.

2. Select the image you want to launch.
3. On the top of the panel, **click Launch**. You are redirected to the **Choose an Instance Type** window.
4. Choose the instance type according to the resources you need to launch your image. **Click Review and Launch**.
5. Review your instance launch details. You can edit each section, such as **Security**, **Storage**, for example, if you need to make any changes. After you finish the review, click **Launch**.
6. To launch the instance, you must select a public key to access it.
   Create a new key pair in EC2 and attach it to the new instance.
   a. From the drop-down menu list, select **Create a new key pair**.
   b. Enter the name to the new key pair. It generates a new key pair.
   c. Click **Download Key Pair** to save the new key pair on your local system.
7. Then, you can click **Launch Instance** to launch your instance.
   You can check the status of the instance, it shows as **Initializing**.
8. After the instance status is **running**, the **Connect** button turns available.
9. Click **Connect**. A popup window appears with instructions on how to connect by using SSH.
   a. Select the preferred connection method to **A standalone SSH client** and open a terminal.
   b. In the location you store your private key, make sure that your key is publicly viewable for SSH to work. To do so, run the command:

```
$ chmod 400 <your-instance-name.pem>
```

   c. Connect to your instance by using its Public DNS:

```
$ ssh -i "<_your-instance-name.pem_> ec2-user@<_your-instance-IP-address_>"
```

   d. Type **yes** to confirm that you want to continue connecting.

As a result, you are connected to your instance over SSH.

Verification

- From a terminal, check if you are able to perform any action while connected to your instance by using SSH.

6.5. COPYING YOUR CUSTOMIZED RHEL SYSTEM IMAGE FOR AWS TO A DIFFERENT REGION ON YOUR AWS EC2

You can copy the image you successfully shared with the **Amazon Web Services EC2** to your own account. Doing so, you grant that the image you shared and copied is available until you delete it, instead of expiring after some time. To copy your image to your own account, follow the steps:

Prerequisites

- You have access to your customized image on AWS.

Procedure

1. From the list of **Public images**, select the image you want to copy.
2. On the top of the panel, click **Actions**.
3. From the drop-down menu, choose **Copy AMI**. A pop-up window appears.
4. Choose the **Destination region** and click **Copy AMI**.
   After the copying process is complete, you are provided with the new **AMI ID**. You can launch a new instance in the new region.

NOTE

When you copy an image to a different region, it results in a separate and new **AMI** in the destination region, with a unique **AMI ID**.

6.6. SHARING AWS IMAGES TO OTHER REGIONS

You can share an existing AWS image to a new region. Sharing the image configures it for the new regions to run on your AWS account. After configuring new regions, all these regions launch with the same configuration as the original AWS image.

Prerequisites

- You created an AWS image.

Procedure

1. From the **Images** table, select the image you want to share with other regions.
2. From the Node options menu (▮), select **Share to new region**. The **Share to new region** wizard opens.
3. From the **Select region** dropdown menu, select the region to share the image.
   You can choose more than one region to share your image with.
4. Click **Share**.
   Your image is built, uploaded to AWS, and shared to the regions you selected.

NOTE

The shared image expires in 14 days.

5. To ensure that you can access the image permanently, copy the Red Hat image to your own AWS account.

6.7. AUTHORIZING IMAGE BUILDER TO PUSH IMAGES TO MICROSOFT AZURE CLOUD

You must authorize Insights image builder to push images to the Microsoft Azure cloud. This is a one-time action. he following are high-level steps:

- Configure Insights Images as an authorized application for your **tenant GUID**
- Give the role of **Contributor** to at least one resource group of the authorized application .
  To authorize Image Builder as an authorized application, follow the steps:

Prerequisites

- You have an existing **Resource Group** in Microsoft Azure portal.
- You have the **User Access Administrator** role rights.
- Your Microsoft Azure subscription has **Microsoft.Storage** and **Microsoft.Compute** as a resource provider.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**. The Insights image builder dashboard appears.
2. Click **Create blueprint**. The **Image output** wizard opens.

On the **Image output** page, complete the following steps:

a. From the **Release** list, select the release that you want to use.

b. From the **Select target environments** option, select **Microsoft Azure**.
Click **Next**.

   1. On the **Target Environment - Microsoft Azure**window, to add Image Builder as an authorized application, select one of the following **share method** options:

c. **Use an account configured from Sources**

   i. From the **Source name** dropdown menu, select the source that you previously configured.
   See Connecting Microsoft Azure account to the Red Hat Hybrid Cloud Console .

      1. The **Azure tenant GUID**, the **Subscription ID**, and the **Resource group** are automatically completed, and the **Authorize image builder** button becomes available.
      Image builder checks if your **Tenant GUID** is correctly formatted and the **Authorize image builder** button becomes available.

d. **Manually enter the account information**

   i. Enter your **Azure Tenant GUID**.
   Image builder checks if your **Tenant GUID** is correctly formatted and the **Authorize image builder** button becomes available.

e. One time action: Click **Authorize image builder** to authorize Image Builder to push images to the Microsoft Azure cloud.
This redirects you to the Microsoft Azure portal.

   i. Login with your credentials.

ii. Click **Accept** the **Permission requested**. Note that, if you already went through the authentication process before, you will not see the **Permission requested**. It is already granted.

f. Confirm that Image Builder is authorized for your tenant.

i. In the **search** bar, search for **Azure Active Directory**.

ii. From the **Services** menu, click **Microsoft Entra ID**, from the left menu. The **Azure Active Directory** page opens.

iii. Search for Insights image builder and confirm it is authorized.

iv. In the **Azure Active Directory**, from the **Services** list, select **Enterprise applications**.

v. In the **Enterprise applications** page, from the **Manage list** menu, click **All applications**. You can see **Red Hat Image Builder** is authorized in the Microsoft Azure cloud.

g. Add the **Red Hat Image Builder** as a contributor to your **Resource Group**.

i. In the search bar, type **Resource Groups** and select the first entry under **Services**. This redirects you to the **Resource Groups** dashboard.

ii. Search and select your **Resource Group** by name.

iii. On the lateral menu, click **Access control (IAM)** to add a permission to the **Red Hat Image Builder** application to access your resource group.

iv. From the menu, click the tab **Role assignments**.

v. Click **+Add**.

vi. From the dropdown menu, choose **Add role assignment**. A menu appears on the left side.

Select role

Assign the **Contributor** role.

Assign access to

Select the option **Assign access to user, group, and service principal**

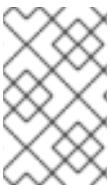
Members

Click **+Select members** and type **Red Hat** in the search bar. Click **enter**.

Select

Red Hat Image Builder application.

The Red Hat Image Builder application is now authorized to push images to the Microsoft Azure cloud.



NOTE

The Red Hat Image Builder application can locate resources only when the account administrator adds the shared application as a contributor under the **IAM** section of the resource group.

Verification

- From the menu, click the tab **Role assignments**.
  You can see Red Hat Image Builder set as a **Contributor** of the **Resource Group** you selected.

Additional resources

- Manage Microsoft Azure Resource Manager resources group by using the Microsoft Azure portal

6.8. ACCESSING YOUR CUSTOMIZED RHEL SYSTEM IMAGE FROM YOUR MICROSOFT AZURE ACCOUNT

After finishing to build and upload the image, and the cloud registration process status is marked as **Ready**, you can access the **Azure Disk Image** from your **Microsoft Azure** account.

Prerequisites

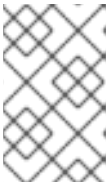
- You have access to your Microsoft Azure dashboard.

Procedure

- Access your **Microsoft Azure** dashboard and navigate to the **Resource group** page.

Verification

1. After you access your Microsoft Azure Account, you can see that the image successfully shared with the resource group account you specified.



NOTE

If the image is not visible there, you might have issues with the upload process. Return to the Insights image builder dashboard and check if the image is marked as **Ready**.

6.9. CREATING A VM INSTANCE BY USING YOUR GCP IMAGE

After the image is built, uploaded, and the cloud registration process status is **Ready**, you can create a Virtual Machine (VM) instance by using the GCP image.

Prerequisites

- You have the universally unique identifier (UUID) of the image you created.
- You have access to the image-builder service API endpoint .
- You have access to your project details at Google Cloud Platform.
- You can access Google Cloud Shell from your browser.

Procedure

1. From the Insights image builder dashboard, copy the image **UUID** of the image that you created.
2. Access /composes/{composeId} API endpoint.
3. Click **Try it Out** to activate the **composeId** string path.

4. Enter the **UUID** into the **composes/{composeId}** field in the API endpoint.

5. Click **Execute**. The API endpoint generates a response in the **Response body**, for example:

```
{
  "image_status": {
    "status": "success",
    "upload_status": {
      "options": {
        "image_name": "composer-api-03f0e19c-0050-4c8a-a69e-88790219b086",
        "project_id": "red-hat-image-builder"
      },
      "status": "success",
      "type": "gcp"
    }
  }
}
```

6. From the **Response body** field, copy the ***image_name*** and ***project_id*** to access the image from the Google Cloud Platform environment.

7. From your browser, access Google Cloud Shell and set your Google Cloud Platform **Project ID** as the default GCP project. You can find the Product ID of your project by accessing the Google Cloud Platform dashboard.

```
$ gcloud config set project PROJECT_ID
```

8. In the **Authorize Cloud Shell** window prompt, click **Authorize** to allow this and future calls that require your credentials.

9. Create a VM instance with the image by using the **gcloud** command in the Google Cloud Shell:

```
$ gcloud compute instances create INSTANCE_NAME \
  --image-project PROJECT_ID_FROM_RESPONSE \
  --image IMAGE_NAME \
  --zone GCP_ZONE
```

Where:

- *INSTANCE_NAME* is the name for your instance;
- *PROJECT_ID_FROM_RESPONSE* is the *project_id* generated by **Response body**;
- *IMAGE_NAME* is the *image_name* generated by **Response body**;
- *GCP_ZONE* is the GCP zone in which the instance will be created.

Verification

1. Verify that Compute Engine created the VM:

```
$ gcloud compute instances describe INSTANCE_NAME
```

2. Connect to the VM instance using SSH:

```
$ gcloud compute ssh --project=PROJECT_ID --zone=ZONE INSTANCE_NAME
```

Additional resources

- Google Cloud Shell documentation
- Default region and zone

6.10. COPYING THE GCE IMAGE TO YOUR PROJECT GROUP

You can create a Virtual Machine (VM) instance using the GCE image.

Prerequisites

- The universally unique identifier (UUID) of the image you created.
- Access to the Image-builder service API endpoint.
- Access to the Google Cloud Shell from your browser.

Procedure

1. From the Images dashboard, copy the **UUID** image of the image you created.
2. Access /composes/{composeId} API endpoint.
3. Click **Try it Out** to activate the **composeId** string path.
4. Enter the **UUID** into the **composes/{composeId}** field in the API endpoint.
5. Click **Execute**. The API endpoint generates a response in the **Response body**, for example:

```
{
  "image_status": {
    "status": "success",
    "upload_status": {
      "options": {
        "image_name": "composer-api-03f0e19c-0050-4c8a-a69e-88790219b086",
        "project_id": "red-hat-image-builder"
      },
      "status": "success",
      "type": "gcp"
    }
  }
}
```

6. From the **Response body** field, copy the ***image_name*** and ***project_id*** to access the image from the Google Cloud Platform environment. From the **Response body**:

```
"image_name": "composer-api-03f0e19c-0050-4c8a-a69e-88790219b086",
"project_id": "red-hat-image-builder"
```

7. From your browser, access Google Cloud Shell.

8. Set your Google Cloud Platform **Project ID** as the default GCP project. You can find the Product ID of your project by accessing the Google Cloud Platform dashboard.

```
$ gcloud config set project PROJECT_ID
```

9. In the **Authorize Cloud Shell** window prompt, click **Authorize** to allow this and future calls that require your credentials.

10. Copy the image to your project by using the gcloud command:

```
$ gcloud compute images create MY_IMAGE_NAME \
  --source-image-project red-hat-image-builder \
  --source-image IMAGE_NAME
```

Where:

- *MY_IMAGE_NAME* is the name you give to your instance;
- red-hat-image-builder is the *project_id* generated by **Response body**;
- *IMAGE_NAME* is the *image_name* generated by **Response body**;

Verification

Confirm that the image has been successfully copied to your project:

- Using the Google Cloud Platform UI, by accessing the Compute Engine / Images section.
- Using the **gcloud** tool, by running the command in Google Cloud Shell:

```
$ gcloud compute images list --no-standard-images
```

Additional resources

- Google Cloud Shell documentation

6.11. CREATING A NEW IMAGE FROM AN EXISTING BUILD

You can create a new image from an existing customized RHEL image by using Insights Images. The Insights Images re-creates the exact image, with a different UUID, which you can use to identify the image in the Hybrid Cloud Console. The new image also fetches package updates and refreshes the content with those updates. You can customize this new image to fit your requirements.

NOTE

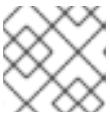
You can re-create images from failed builds.

Prerequisites

- You created an AWS image with Insights Images.

Procedure

1. From the **Images** dashboard, select the image from which you want to create your customized image.
2. Click the **Node options** menu (▮) and select **Re-create image**. The **Create image** wizard opens.



NOTE

If the image status is **Expired**, click the **Re-create image** button.

   a. Optional: You can customize the new image by using the Navigation panel to open a step and making changes. Click **Next**.

   b. On the **Review** page, click **Create image**.

The Insights Images dashboard opens. The image build starts to re-create the image and lists the following information:

- **Image name**
- **UUID**
- Cloud target environment
- Image operating system release
- Status of the image creation

Verification

- From the **Status** column, check if the image is **Ready**.
- Optional: Click **Image details** to display additional information about the re-created image.

6.12. DOWNLOADING THE JSON COMPOSE REQUEST

If you download the **.json** compose request of your image, you can use the image builder **API** to automate your image building tasks, such as:

- Customizing the image with extra packages
- Customizing the partition layout
- Embedding an activation key.

Prerequisites

- You created an image with Insights Images.

Procedure

1. From the **Images** table, select the image that you want to download as a **.json** compose request.
2. Click the **Node options** ( ) menu and select **Download compose request (.json)**.

The **.json** compose request is now saved to your host server. To use the image builder API, see Using hosted image builder via its API .

CHAPTER 7. LAUNCHING CUSTOMIZED RHEL IMAGES TO THE CLOUD PLATFORMS WITH INSIGHTS IMAGE BUILDER

7.1. LAUNCHING A CUSTOMIZED RHEL IMAGE ON AWS

You can launch a customized RHEL image on the AWS cloud environment.

Prerequisites

- You have a RHEL subscription.
- You have an AWS account created.
- You successfully built a customized RHEL image and shared it with the region you want to launch the new instance in.
- The customized RHEL image you built was shared with the same AWS integration account.
- You added an AWS integration account to the Hybrid Cloud Console.
- You have the "Launch on AWS User" role assigned. See how to assign a role to a user .

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**.
2. Find an image you want to launch in the public cloud environment and click **Launch** in the **Instance** column. The **Launch** wizard opens.
3. On the **Account and customization** page, complete the following steps:
   a. From the **Select account** drop-down menu, select the account you want to use.
   b. From the **Select region** drop-down menu, select the region to run the instance.
   c. Optionally: From the **Select template** drop-down menu, select the template you want to use.
   If you do not specify the template, you launch the image under the default security group. Ensure that the default security group allows SSH traffic.
   d. From the **Select instance type** drop-down menu, select the instance type configuration.
   e. In the **Count** field, select the number of images you want to launch. Click **Next**.
   The wizard notifies you if you try to launch too many images. Make sure you have enough resources in your AWS account.

NOTE

You must have the default Virtual Private Cloud (VPC) and Security Group on your AWS account. If you do not have them, contact the AWS support to re-create them.

4. On the **SSH key authentication** page, select one of the options:

a. **Select existing SSH public key**. From the **Select public key** drop-down menu, add an existing SSH public key.

b. **Add and save a new SSH public key** Enter a name for your new SSH public key and drag or upload a new SSH public key file. Click **Next**.

5. On the **Review** page, review the details about the image launch process and click **Launch**.

Verification

1. The **Launch** wizard shows the green checkmark with a message **System launched successfully**.
2. To verify the instance is running, copy the **ssh** command displayed on the screen to your terminal and connect to the instance.

To stop the running instance, see the AWS console documentation.

7.2. LAUNCHING A CUSTOMIZED RHEL IMAGE ON MICROSOFT AZURE

Prerequisites

- You have a RHEL subscription.
- You have a Microsoft Azure account created.
- You successfully built a customized RHEL image.
- The customized RHEL image you built was shared with the same Microsoft Azure integration account.
- You added a Microsoft Azure integration account to the Hybrid Cloud Console.
- You have the "Launch on Azure User" role assigned. See how to assign a role to a user .

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**.
2. Find an image you want to launch in the public cloud environment and click **Launch** in the **Instance** column. The **Launch** wizard opens.
3. On the **Account and customization** page, complete the following steps:

a. From the **Select account** drop-down menu, select the account you want to use.

b. From the **Azure resource group** drop-down menu, select the resource group in which you want to run your instance.
This creates the resources in the same region that this resource group is located. You can leave this field empty to run the instance in the same resource group as the image.

c. From the **Select instance size** drop-down menu, select the instance type configuration.

d. In the **Count** field, select the number of images you want to launch. Click **Next**.

The wizard notifies you if you try to launch too many images. Make sure you have enough quotas in your Microsoft Azure subscription when you are launching a large set of images.

4. On the **SSH key authentication** page, choose to use an existing SSH key or add a new SSH key:

To select an existing SSH public key, follow the steps

a. From the **Select public key** drop-down menu, choose an existing SSH public key.

b. Click **Next**.

To Add and save a new SSH public key, follow the steps

a. In the **Name** field, enter a name for your new SSH public key.

b. In the SSH public key, drag or upload a new SSH public key file.

c. Click **Next**.

NOTE

Microsoft Azure does not support the **ed25519** SSH keys.

5. On the **Review** page, review the details about the image launch process and click **Launch**.

The launching process takes a few minutes to start an instance on the Microsoft Azure cloud platform.

Verification

1. The **Launch** wizard shows the green checkmark with a message **System launched successfully**.
2. To verify the instance is running, copy the **ssh** command displayed on the screen to your terminal and connect to the instance.

7.3. LAUNCHING A CUSTOMIZED RHEL IMAGE ON THE GOOGLE CLOUD PLATFORM

Prerequisites

- You have a RHEL subscription.
- You have a Google Cloud Platform (GCP) project created .
- You successfully built a customized RHEL image.
- The customized RHEL image you built is shared with the same GCP project.
- You have the "Launch on Google User" role assigned. See how to assign a role to a user .
- You added a GCP project to the Hybrid Cloud Console.
- You have enabled the following APIs in your GCP project:
  - Compute Engine API;

- Identity and Access Management API.

Procedure

1. Access Hybrid Cloud Console, click **Services → Red Hat Enterprise Linux→ Inventory → Images**.
2. Find the image you want to launch in the public cloud environment and click **Launch** in the **Instance** column. The **Launch** wizard opens.
3. On the **Account and customization** page, complete the following steps:
   a. From the **Select account** drop-down menu, select the account you want to use.
   b. Optionally: From the **Select template** drop-down menu, select the template you want to use.
   If you do not specify the template, you launch the image to the default Virtual Private Cloud (VPC) with its firewall rules.
   c. From the **Select machine type** drop-down menu, select the machine configuration.
   d. In the **Count** field, select the number of images you want to launch and click **Next**.
4. On the **SSH key authentication** page, select one of the options:
   a. **Select existing SSH public key**. From the **Select public key** drop-down menu, choose an existing SSH public key. Click **Next**.
   b. **Add and save a new SSH public key** Enter a name for your new SSH public key and drag or upload a new SSH public key file. Click **Next**.
5. On the **Review** page, review the details about the image launch process and click **Launch**.

Verification

1. The **Launch** wizard shows the green checkmark with a message **System launched successfully**.
2. To verify the instance is running, copy the **ssh** command displayed on the screen to your terminal and connect to the instance.

7.4. CONFIGURING LAUNCH NOTIFICATIONS

You can configure notifications for various events of the launching process. For information on how to configure notifications in the Red Hat Hybrid Cloud Console, see Configuring notifications on the Red Hat Hybrid Cloud Console.

Launch events

Launch failed

If an image launch fails, a notification is sent.

Launch completed

If an image launch is successful, a notification is sent.

Note, if you choose email notifications, these notifications contain detailed information about the event. For example, if the **Launch completed** event is triggered, the email notification contains the list of the launched instances.

CHAPTER 8. DEPLOYING YOUR CUSTOMIZED IMAGES

After creating customized images for the VMWare vSphere private cloud, you can deploy the images to VMware vSphere. For the Guest image (**.qcow2**), and Installer (**.iso**), you can then download these images and deploy them to virtual machines.

NOTE

You can only launch customized images directly from Hybrid Cloud Console to the following public clouds: AWS, GCP, and Azure. The VMDK customized images must first be uploaded to VMWare vSphere, deployed there, and then you can create a VM. For the Guest image (**.qcow2**), and Installer (**.iso**), you can then download these images and deploy them directly to virtual machines.

8.1. CONFIGURING CREDENTIALS TO ACCESS YOUR DEPLOYED SYSTEMS WITH CLOUD-INIT

You cannot add a username and password to a blueprint by using the Red Hat Insights images UI interface. To add a username and password to your image, use the **cloud-init** tool.

The following example shows how to add credentials to a VMware OVA image type created with Insights image builder. You can add credentials to other image types by using this method.

The Open virtualization format (**.ova**) is a **.vmdk** image with additional metadata about the virtual hardware. The **.ova** image contains the **cloud-init** package installed that you can use to provision users by using a **user-data** file, for example.

Instead of sharing your login credentials to a hosted service, use **cloud-init** and **open-vm-tools**, that are installed within the image and enabled by default. For example, you can use it to pass the credentials to the VMware vSphere Cloud Director by using **cloud-init**.

Prerequisites

- You created an image in the **.ova** format by using the Hybrid Cloud Console.

Procedure

1. Access the directory where you downloaded your **.ova** image.
2. Create a file named **metadata.yaml** and add the following information to this file:

```
instance-id: cloud-vm
local-hostname: vmname
```

3. Create a file **userdata.yaml**. Add the following information to the file:

```
#cloud-config
users:
- name: admin
  sudo: "ALL=(ALL) NOPASSWD:ALL"
  ssh_authorized_keys:
  - ssh-rsa AAA...fhHQ== your.email@example.com
```

This file contains the administrator or root user credentials with no password that you can use to

access your system after the first boot and create additional users later. The **ssh_authorized_keys** field is your SSH public key. You can find your SSH public key in **~/.ssh/id_rsa.pub**.

Next steps

- Deploy your image to vSphere by using the CLI or to the the vSphere GUI.

8.2. CONFIGURING CREDENTIALS TO ACCESS YOUR DEPLOYED SYSTEMS WITH API

You cannot add a username and password to a blueprint by using the Red Hat Insights images UI interface. To add a username and password to your image, use the Image-builder service API.

The following example shows how to add credentials to a VMware OVA image type created with Insights image builder. You can also use this method to add credentials to other image types.

Prerequisites

- Access to Hybrid Cloud Console.
- OAuth 2.0 authorization.
- You created an offline token. See Generate an offline token.
- Access to the Image-builder service API.
- The **jq** tool is installed

Procedure

1. The offline token that you generated by using Red Hat API Tokens cannot be used directly with image builder. To use it, follow the steps:

   a. Save your offline token to a **OFFLINE_TOKEN** variable.

   b. Exchange the offline token for an access token by using the following command:

```
$ OFFLINE_TOKEN=”YOUR_OFFLINE_TOKEN”
$ curl --silent \
    --request POST \
    --data grant_type=refresh_token \
    --data client_id=rhsm-api \
    --data refresh_token=$OFFLINE_TOKEN \
    https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token \
  | jq .
```

   It generates an output similar to the following example:

```
{
  "access_token": "oiZjo1Mjhk...",
  "expires_in": 900,
  "refresh_expires_in": 0,
  "refresh_token": "eyJhbG...",
  "token_type": "bearer",
```

```
  "not-before-policy": 0,
  "session_state": "f0dbb8d4-4e4e-4654-844c-6f3704c84422",
  "scope": "offline_access"
}
```

2. Use **jq** to get the actual access token from the JSON payload and save it in a variable using the following snippet:

```
$ access_token=$( \
    curl --silent \
      --request POST \
      --data grant_type=refresh_token \
      --data client_id=rhsm-api \
      --data refresh_token=$OFFLINE_TOKEN \

https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token \
    | jq -r .access_token \
  )
```

The access token has an expiration time. If you receive an authorization error, rerun the previous command to generate a new access token.

3. In your system, create a compose request in the **.json** format. The following example creates an up-to-date RHEL 9.4 **ova** for x86_64 CPU architecture.

```
$ cat request.json
{
    "image_name": "ova_image_name",
    "distribution": "rhel-94",
    "image_requests": [
        {
            "architecture": "x86_64",
            "image_type": "vsphere-ova",
            "upload_request": {
                "type": "vmdk",
                "options": {}
            }
        }
    ],
    "customizations": {
        "users": [
            {
                "name": "user-name",
                "ssh_key": "ssh-rsa AAAAB...qfGI+vk",
                "password": "password"
            }
        ]
    }
}
```

4. Send the compose request to the image builder API:

```
$ curl --silent \
    --request POST \
    --header "Authorization: Bearer $access_token" \
```

```
    --header "Content-Type: application/json" \
    --data @request.json \
    https://console.redhat.com/api/image-builder/v1/compose
```

If the request is successful, you can see an output similar to the following, that is the image ID:

```
{"id":"fd4ecf3c-f0ce-43dd-9fcc-6ad11208b939"}
```

5. Check the status of the image building:

```
$ curl \
    --silent \
    --header "Authorization: Bearer $access_token" \
    "https://console.redhat.com/api/image-builder/v1/composes/$compose_id" \
  | image_ID.
```

If the request is successful, you can see an output similar to the following, that is the image ID:

```
{"id":"fd4ecf3c-f0ce-43dd-9fcc-6ad11208b939"}
```

You can also check the image building progress by accessing Red Hat Hybrid Cloud Console .

After the image builds, you can see the following output:

If the request is successful, you can see an output similar to the following, that is the image ID:

```
{
    "image_status": {
        "status": "success",
        "upload_status": {
            "options": {
                "url": "https://image-builder-service-production.s3.amazonaws.com/composer-api-
76...-disk.ova?e42..."
            },
            "status": "success",
            "type": "aws.s3"
        }
    }
}
```

6. After finishing the image creation, download the image.

```
$ curl --location --output vsphere-ova.vmdk  \
    “https://image-builder-service-production.s3.amazonaws.com/composer-api-76...-disk.ova?
e42...”
```

The image is saved to your system and ready to be used.

Next steps

- Deploy your image to vSphere by using the CLI or to the vSphere GUI.

Additional resources

- The Getting started with Red Hat APIs article
- The Image-builder service API blog post

8.3. UPLOADING VMDK IMAGES AND CREATING A RHEL VIRTUAL MACHINE IN VSPHERE

After creating and configuring your image, you can deploy it to **VMware vSphere** by using the CLI, and you can create a VM and log in to it.

Prerequisites

- You configured credentials to access your deployed systems by using the **cloud-init** tool. See Configuring credentials to access your deployed systems .
- You configured the **govc** VMware CLI tool client with the following values in the environment by setting the following values in the environment:

```
GOVC_URL
GOVC_DATACENTER
GOVC_FOLDER
GOVC_DATASTORE
GOVC_RESOURCE_POOL
GOVC_NETWORK
```

Procedure

1. Export the **metadata.yaml** and **userdata.yaml** files to the environment, compressed with **gzip**, encoded in **base64** as follows. They will be used in further steps.

```
export METADATA=$(gzip -c9 <metadata.yaml | { base64 -w0 2>/dev/null || base64; }) \
USERDATA=$(gzip -c9 <userdata.yaml | { base64 -w0 2>/dev/null || base64; })
```

2. Launch the image on vSphere with the **metadata.yaml** and **userdata.yaml** files:
   a. Import the **.vmdk** image in to vSphere:

```
$ govc import.vmdk ./composer-api.vmdk foldername
```

   b. Create the VM in vSphere without powering it on:

```
govc vm.create \
-net.adapter=vmxnet3 \
-m=4096 -c=2 -g=rhel8_64Guest \
-firmware=bios -disk="foldername/composer-api.vmdk" \
-disk.controller=ide -on=false \
 vmname
```

   c. Change the VM to add **ExtraConfig** variables, the **cloud-init** config:

```
govc vm.change -vm vmname \
-e guestinfo.metadata="${METADATA}" \
-e guestinfo.metadata.encoding="gzip+base64" \
```

```
-e guestinfo.userdata="${USERDATA}" \
-e guestinfo.userdata.encoding="gzip+base64"
```

d. Power-on the VM:

```
govc vm.power -on vmname
```

e. Retrieve the VM IP address:

```
HOST=$(govc vm.ip vmname)
```

f. Use SSH to log in to the VM, using the user-data specified in **cloud-init** file configuration:

```
$ ssh admin@HOST
```

Additional resources

- The govc documentation
- The VMware - cloud init 22.2 documentation

8.4. DEPLOYING OVA VMDK IMAGES TO THE VSPHERE GUI

After creating your **.vmdk** image in the **open virtualization** format (**.ova**), you can deploy it to **VMware vSphere** by using the vSphere GUI client. It will create a VM which can be customized further before booting.

Prerequisite

- You logged in to the vSphere UI in a browser.
- You downloaded your (**.ova**) image.

Procedure

1. In the vSphere Client, from the **Actions** menu, select **Deploy OVF Template**.
2. On the **Deploy OVF Template** page, complete the settings for each configuration option and click **Next**.
3. Click **Finish**. The **.ova** image starts to be deployed.
   After the image deployment is complete, you have a new virtual machine (VM) from the **.ova** image.
4. In the deployed image page, perform the following steps:
   a. From the **Actions** menu, select **Edit Setting**.
   b. On the **Virtual Hardware** tab, configure resources such as CPU, memory, add a new network adapter, between others of your choice.
      i. On the **CD/DVD drive 1** option, attach a CD or DVD Drive that contains a **cloud-init.iso**, to provision a user on startup.

The VM is now ready to boot with the username and password from the **cloud-init.iso** file.

Additional resources

- Deploy an OVF or OVA Template
- The govc documentation
- The VMware - cloud init 22.2 documentation

8.5. CREATING A VIRTUAL MACHINE FROM THE CUSTOMIZED RHEL GUEST SYSTEM IMAGE

You can create a virtual machine (VM) from the **QCOW2** image that you created by using Insights Images.

Prerequisites

- You created and downloaded a **QCOW2** image by using Insights Images.

Procedure

1. Access the directory where you downloaded your **QCOW2** image.
2. Create a file named **meta-data**. Add the following information to this file:

```
instance-id: nocloud
local-hostname: vmname
```

3. Create a file named **user-data**. Add the following information to the file:

```
#cloud-config
user: admin
password: password
chpasswd: {expire: False}
ssh_pwauth: True
ssh_authorized_keys:
  - ssh-rsa AAA...fhHQ== your.email@example.com
```

   - **ssh_authorized_keys** is your SSH public key. You can find your SSH public key in ~/**.ssh/id_rsa.pub**.

4. Use the **genisoimage** command to create an ISO image that includes the **user-data** and **meta-data** files.

```
# genisoimage -output cloud-init.iso -volid cidata -joliet -rock user-data meta-data

I: -input-charset not specified, using utf-8 (detected in locale settings)
Total translation table size: 0
Total rockridge attributes bytes: 331
Total directory bytes: 0
```

```
Path table size(bytes): 10
Max brk space used 0
183 extents written (0 MB)
```

5. Create a new VM from the KVM Guest Image using the **virt-install** command. Include the ISO image you created on step 4 as an attachment to the VM image.

```
# virt-install \
    --memory 4096 \
    --vcpus 4 \
    --name myvm \
    --disk composer-api.qcow2,device=disk,bus=virtio,format=qcow2 \
    --disk cloud-init.iso,device=cdrom \
    --os-variant rhel1-latest \
    --virt-type kvm \
    --graphics none \
    --import
```

Where,

- --graphics none - indicates that it is a headless RHEL Virtual Machine.
- --vcpus 4 - indicates that it uses 4 virtual CPUs.
- --memory 4096 - indicates that it uses 4096 MB RAM.

6. The VM installation starts:

```
Starting install...
Connected to domain myvm
...
[  OK  ] Started Execute cloud user/final scripts.
[  OK  ] Reached target Cloud-init target.

Red Hat Enterprise Linux 1-latest (Ootpa)
Kernel 4.18.0-221.el8.x86_64 on an x86_64
```

Additional resources

- Creating virtual machines using the command-line interface

8.6. INSTALLING A CUSTOMIZED RHEL ISO SYSTEM IMAGE TO A BARE METAL SYSTEM

You can create a virtual machine (VM) from the ISO image that you created using the Insights image builder.

Prerequisites

- You created and downloaded an ISO image by using Insights image builder.
- A 8 GB USB flash drive.

Procedure

1. Access the directory where you downloaded your ISO image.
2. Place the bootable ISO image file on a USB flash drive.
3. Connect the USB flash drive to the port of the computer you want to boot.
4. Boot the ISO image from the USB flash drive.
5. Perform the steps to install the customized bootable ISO image.
   The boot screen shows you the following options:
   - Install Red Hat Enterprise Linux 1-latest
   - Test this media & install Red Hat Enterprise Linux 1-latest

Additional resources

- Booting the installation

8.7. IMPORTING AND RUNNING QCOW2 IMAGES ON OCI

You can import your customized **.qcow2** image to the Oracle Cloud Infrastructure (OCI). Then, you can launch the customized **.qcow2** image on a virtual machine.

Prerequisites

- You logged in to the Oracle Cloud UI in a browser.
- You downloaded your **.qcow2** image.
- You have copied the **.qcow2** image **Image Link** URL from Insights **Instance** column, in the Images dashboard.

Procedure

1. In the Oracle Cloud UI dashboard, click **Compute > Custom Images**
2. On the **Custom Images** dashboard, click **Import image**.
3. On the **Import image** window, set the following configuration:
   a. Select the **Import from an object storage URL** option.
   b. In the **Object Storage URL** field, paste the URL given by Insights Images into it.
   c. Choose the **QCOW2** image type.
   d. Under **Launch mode**, select the **Paravirtualized mode** option.
4. Click **Import Image**.

Once the system finishes importing the image, you can run the customized image in the OCI environment.

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate and prioritize your feedback regarding our documentation. Provide as much detail as possible, so that your request can be quickly addressed.

Prerequisites

- You are logged in to the Red Hat Customer Portal.

Procedure

To provide feedback, perform the following steps:

1. Click the following link: Create Issue.
2. Describe the issue or enhancement in the **Summary** text box.
3. Provide details about the issue or requested enhancement in the **Description** text box.
4. Type your name in the **Reporter** text box.
5. Click the **Create** button.

This action creates a documentation ticket and routes it to the appropriate documentation team. Thank you for taking the time to provide feedback.

OPENING A SUPPORT CASE AT RED HAT SUPPORT

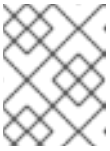
Create a support case from Red Hat Insights at Red Hat Support by performing the following steps:

Prerequisites

- You are logged in to the Red Hat Customer Portal.

Procedure

1. Access the Red Hat Hybrid Cloud Console :
2. Click **Help ?** and select **Open a support case**.
   You are redirected to the **Customer support** page.
3. From the **Get Support** page, select the type of issue that you want to report and click **Continue**.
4. From the **Summarize** page, perform the following steps:
   a. On the **Summary** field, describe the issue.



NOTE

If **Red Hat Insights** is not auto-selected, you must manually select the product.

   b. From the **Product** dropdown menu, select **Red Hat Insights**.
   c. From the **Version** dropdown menu, select the component you have issues with.
   d. From the **Review** page, click **Submit**.
   A support case is created.