



Red Hat OpenStack Platform 13

Introduction to the OpenStack Dashboard

An overview of the Red Hat OpenStack Platform Dashboard graphical user interface

Red Hat OpenStack Platform 13 Introduction to the OpenStack Dashboard

An overview of the Red Hat OpenStack Platform Dashboard graphical user interface

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Copyright © 2021 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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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 provides an outline of the options available in the Red Hat OpenStack Platform Dashboard user interface.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE	3
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION	4
CHAPTER 1. THE RED HAT OPENSTACK PLATFORM DASHBOARD SERVICE (HORIZON)	5
1.1. THE ADMIN TAB	5
1.1.1. View Allocated Floating IP Addresses	6
1.2. THE PROJECT TAB	6
1.3. THE IDENTITY TAB	8
CHAPTER 2. CUSTOMIZING THE DASHBOARD	9
2.1. OBTAINING THE HORIZON CONTAINER IMAGE	9
2.2. OBTAINING THE RCUE THEME	9
2.3. CREATING YOUR OWN THEME BASED ON RCUE	10
2.4. CREATING A FILE TO ENABLE YOUR THEME AND CUSTOMIZE THE DASHBOARD	10
2.5. GENERATING A MODIFIED HORIZON IMAGE	11
2.6. USING THE MODIFIED CONTAINER IMAGE IN THE OVERCLOUD	12
2.7. EDITING PUPPET PARAMETERS	12
2.8. DEPLOYING AN OVERCLOUD WITH A CUSTOMIZED DASHBOARD	12

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](#).

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Tell us how we can make it better.

Using the Direct Documentation Feedback (DDF) function

Use the **Add Feedback** DDF function for direct comments on specific sentences, paragraphs, or code blocks.

1. View the documentation in the *Multi-page HTML* format.
2. Ensure that you see the **Feedback** button in the upper right corner of the document.
3. Highlight the part of text that you want to comment on.
4. Click **Add Feedback**.
5. Complete the **Add Feedback** field with your comments.
6. Optional: Add your email address so that the documentation team can contact you for clarification on your issue.
7. Click **Submit**.

CHAPTER 1. THE RED HAT OPENSTACK PLATFORM DASHBOARD SERVICE (HORIZON)

The Red Hat OpenStack Platform (RHOSP) Dashboard (horizon) is a web-based graphical user interface that you can use to manage RHOSP services.

To access the browser dashboard, you must install the Dashboard service, and you must know the dashboard host name, or IP, and login password. The dashboard URL is:

`http://HOSTNAME/dashboard/`

1.1. THE ADMIN TAB

The **Admin** tab provides an interface where administrative users can view usage and manage instances, volumes, flavors, images, projects, users, services, and quotas.



NOTE

The **Admin** tab displays in the main window only if you have logged in as a user with administrative privileges.

The following options are available in the **Admin** tab:

Table 1.1. System Panel

Parameter Name	Description
Overview	View basic reports.
Resource Usage	Use the following tabs to view the following usages: <ul style="list-style-type: none"> ● Usage Report - View the usage report. ● Stats - View the statistics of all resources.
Hypervisors	View the hypervisor summary.
Host Aggregates	View, create, and edit host aggregates. View the list of availability zones.
Instances	View, pause, resume, suspend, migrate, soft or hard reboot, and delete running instances that belong to users of some, but not all, projects. Also, view the log for an instance or access an instance using the console.
Volumes	View, create, edit, and delete volumes, and volume types.
Flavors	View, create, edit, view extra specifications for, and delete flavors. Flavors are the virtual hardware templates in OpenStack.

Parameter Name	Description
Images	View, create, edit properties for, and delete custom images.
Networks	View, create, edit properties for, and delete networks.
Routers	View, create, edit properties for, and delete routers.
Floating IPs	View allocated floating IP addresses for all projects.
Defaults	View and edit the default quotas (maximum limits) for resources in the environment.
Metadata Definitions	Import, view, and edit metadata definition namespaces, and associate the metadata definitions with specific resource types.
System Information	<p>Contains the following tabs:</p> <ul style="list-style-type: none"> ● Services - View a list of the services. ● Compute Services - View a list of all Compute services. ● Network Agents - View the network agents. ● Block Storage Services - View a list of all Block Storage services. ● Orchestration Services - View a list of all Orchestration services.

1.1.1. View Allocated Floating IP Addresses

You can use the **Floating IPs** panel to view a list of allocated floating IP addresses. This information can also be accessed using the command line:

```
$ nova list --all-tenants
```

1.2. THE PROJECT TAB

In the **Project** tab you can view and manage project resources. Set a project as active in **Identity > Projects** to view and manage resources in that project.

The following options are available in the **Project** tab:

Table 1.2. The Compute tab

Parameter name	Description
Overview	View reports for the project.

Parameter name	Description
Instances	View, launch, create a snapshot from, stop, pause, or reboot instances, or connect to them through the console.
Volumes	Use the following tabs to complete these tasks: <ul style="list-style-type: none"> ● Volumes - View, create, edit, and delete volumes. ● Volume Snapshots - View, create, edit, and delete volume snapshots.
Images	View images, instance snapshots, and volume snapshots created by project users, and any images that are publicly available. Create, edit, and delete images, and launch instances from images and snapshots.
Access & Security	Use the following tabs to complete these tasks: <ul style="list-style-type: none"> ● Security Groups - View, create, edit, and delete security groups and security group rules. ● Key Pairs - View, create, edit, import, and delete key pairs. ● Floating IPs - Allocate an IP address to or release it from a project. ● API Access - View API endpoints, download the OpenStack RC file, download EC2 credentials, and view credentials for the logged-in project user.

Table 1.3. The Network tab

Parameter name	Description
Network Topology	View the interactive topology of the network.
Networks	Create and manage public and private networks and subnets.
Routers	Create and manage routers.
Trunks	Create and manage trunks. Requires the trunk extension enabled in OpenStack Networking (neutron).

Table 1.4. The Object Store tab

Parameter name	Description
Containers	Create and manage storage containers. A container is a storage compartment for data, and provides a way for you to organize your data. It is similar to the concept of a Linux file directory, but it cannot be nested.

Table 1.5. The Orchestration tab

Parameter name	Description
Stacks	Orchestrate multiple composite cloud applications using templates, through both an OpenStack-native REST API and a CloudFormation-compatible Query API.

1.3. THE IDENTITY TAB

In the **Identity** tab you can view and manage projects and users.

The following options are available in the **Identity** tab:

- **Projects** - View, create, edit, and delete projects, view project usage, add or remove users as project members, modify quotas, and set an active project.
- **Users** - View, create, edit, disable, and delete users, and change user passwords. The **Users** tab is available when you log in as an admin user.

For more information about managing your cloud with the Red Hat OpenStack Platform dashboard, see the following guides:

- [Instances and Images Guide](#)
- [Networking guide](#)
- [Users and Identity Management guide](#)

CHAPTER 2. CUSTOMIZING THE DASHBOARD

The Red Hat OpenStack Platform (RHOSP) dashboard (horizon) uses a default theme (RCUE), which is stored inside the horizon container. You can add your own theme to the container image and customize certain parameters to change the look and feel of the following dashboard elements:

- Logo
- Site colors
- Stylesheets
- HTML title
- Site branding link
- Help URL



NOTE

To ensure continued support for modified RHOSP container images, the resulting images must comply with the ["Red Hat Container Support Policy"](#).

2.1. OBTAINING THE HORIZON CONTAINER IMAGE

To obtain a copy of the horizon container image, pull the image either into the undercloud or a separate client system that is running podman.

Procedure

- Pull the horizon container image:

```
$ sudo docker pull registry.redhat.io/rhosp13/openstack-horizon
```

You can use this image as a basis for a modified image.

2.2. OBTAINING THE RCUE THEME

The horizon container image uses the Red Hat branded RCUE theme by default. You can use this theme as a basis for your own theme and extract a copy from the container image.

Procedure

1. Create a directory for your theme.

```
$ mkdir ~/horizon-themes  
$ cd ~/horizon-themes
```

2. Start a container that executes a null loop. For example, run the following command:

```
$ sudo docker run --rm -d --name horizon-temp registry.redhat.io/rhosp13/openstack-horizon  
/usr/bin/sleep infinity
```

3. Copy the RCUE theme from the container to your local directory:

```
$ sudo docker cp -a horizon-temp:/usr/share/openstack-  
dashboard/openstack_dashboard/themes/rcue .
```

4. Kill the container:

```
$ sudo docker kill horizon-temp
```

Result: You should have a local copy of the RCUE theme.

2.3. CREATING YOUR OWN THEME BASED ON RCUE

To use RCUE as a basis, copy the entire RCUE theme directory `rcue` to a new location. This procedure uses **mytheme** as an example name.

Procedure

- Copy the theme.

```
$ cp -r rcue mytheme
```

To change the colors, graphics, fonts, and other elements of a theme, edit the files in **mytheme**. When you edit this theme, check for all instances of `rcue` including paths, files, and directories to ensure that you change them to the new **mytheme** name.

2.4. CREATING A FILE TO ENABLE YOUR THEME AND CUSTOMIZE THE DASHBOARD

To enable your theme in the dashboard container, you must create a file to override the **AVAILABLE_THEMES** parameter.

Procedure

1. Create a new file called **_12_mytheme_theme.py** in the **horizon-themes** directory and add the following content:

```
AVAILABLE_THEMES = [('mytheme', 'My Custom Theme', 'themes/mytheme')]
```

The **12** in the file name ensures this file is loaded after the RCUE file, which uses **11**, and overrides the **AVAILABLE_THEMES** parameter.

2. Optional: You can also set custom parameters in the **_12_mytheme_theme.py** file. Use the following examples as a guide:

SITE_BRANDING

Set the HTML title that appears at the top of the browser window.

```
SITE_BRANDING = "Example, Inc. Cloud"
```

SITE_BRANDING_LINK

Changes the hyperlink of the theme logo, which normally redirects to **horizon:user_home** by default.

```
SITE_BRANDING_LINK = "http://example.com"
```

2.5. GENERATING A MODIFIED HORIZON IMAGE

When your custom theme is ready, you can create a new container image to enable and use your theme.

Procedure

1. Use a **dockerfile** to generate a new container image with the original horizon image as a basis. Use the following example **dockerfile** as a guide:

```
FROM registry.redhat.io/rhosp13/openstack-horizon
MAINTAINER Acme
LABEL name="rhosp13/openstack-horizon-mytheme" vendor="Acme" version="0"
release="1"
COPY mytheme /usr/share/openstack-dashboard/openstack_dashboard/themes/mytheme
COPY _12_mytheme_theme.py /etc/openstack-
dashboard/local_settings.d/_12_mytheme_theme.py
RUN sudo chown horizon:horizon /etc/openstack-
dashboard/local_settings.d/_12_mytheme_theme.py
```

2. Save this file in your **horizon-themes** directory as **dockerfile**.
3. Use the **dockerfile** to generate the new image:

```
sudo docker build . -t "192.168.24.1:8787/rhosp13/openstack-horizon-mytheme:0-1"
```

The **-t** option names and tags the resulting image. It uses the following syntax:

```
[LOCATION]/[NAME]:[TAG]
```

LOCATION

This is usually the location of the container registry that the overcloud uses to pull images. To push this image to the undercloud container registry, set this to the undercloud IP and port.

NAME

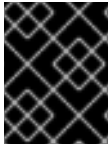
For consistency, this is usually the same name as the original container image followed by the name of your theme. In this example case, it is **rhosp13/openstack-horizon-mytheme**.

TAG

The tag for the image. You can follow the Red Hat convention by using the version and release labels as a basis for this tag. If you generate a new version of this image, increment the release, for example, 0-2.

4. Push the resulting image to the undercloud container registry:

```
$ docker push 192.168.24.1:8787/rhosp13/openstack-horizon-mytheme:0-1
```



IMPORTANT

If you update or upgrade RHOSP you must reapply the theme to the new horizon image and push a new version of the modified image to the undercloud.

2.6. USING THE MODIFIED CONTAINER IMAGE IN THE OVERCLOUD

To use the container image that you modified with your overcloud deployment, edit the environment file that contains the list of container image locations. This environment file is usually named **overcloud-images.yaml**.

Procedure

1. Edit the **DockerHorizonConfigImage** and **DockerHorizonImage** parameters to point to your modified container image:

```
parameter_defaults:
...
DockerHorizonConfigImage: 192.168.24.1:8787/rhosp13/openstack-horizon-mytheme:0-1
DockerHorizonImage: 192.168.24.1:8787/rhosp13/openstack-horizon-mytheme:0-1
...
```

2. Save this new version of the **overcloud-images.yaml** file.

2.7. EDITING PUPPET PARAMETERS

Director provides a set of dashboard parameters that you can modify with environment files.

Procedure

- Use the **ExtraConfig** parameter to set Puppet hieradata. For example, the default help URL points to <https://access.redhat.com/documentation/en/red-hat-openstack-platform>. To modify this URL, use the following environment file content and replace the URL:

```
parameter_defaults:
  ExtraConfig:
    horizon::help_url: "http://openstack.example.com"
```

2.8. DEPLOYING AN OVERCLOUD WITH A CUSTOMIZED DASHBOARD

Procedure

- To deploy the overcloud with your dashboard customizations, include the following environment files in the **openstack overcloud deploy** command:
 - The environment file with your modified container image locations.
 - The environment file with additional dashboard modifications.
 - Any other environment files that are relevant to your overcloud configuration.

```
$ openstack overcloud deploy --templates \
  -e /home/stack/templates/overcloud-images.yaml \
```



```
| -e /home/stack/templates/help_url.yaml \  
[OTHER OPTIONS]
```