



Supplement to Admin User Guide and End User Guide

SUSE OpenStack Cloud 6



Supplement to Admin User Guide and End User Guide

SUSE OpenStack Cloud 6

by Frank Sundermeyer and Tanja Roth

Publication Date: March 04, 2016

SUSE LLC

10 Canal Park Drive

Suite 200

Cambridge MA 02141

USA

<https://www.suse.com/documentation> ↗

Copyright © 2006– 2016 SUSE LLC and contributors. All rights reserved.

Except where otherwise noted, this document is licensed under **Creative Commons Attribution 3.0**

License : <http://creativecommons.org/licenses/by/3.0/legalcode> ↗

For SUSE trademarks, see <https://www.suse.com/company/legal/> ↗. All other third party trademarks are the property of their respective owners. A trademark symbol (®, ™ etc.) denotes a SUSE or Novell trademark; an asterisk (*) denotes a third party trademark.

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither SUSE LLC, its affiliates, the authors nor the translators shall be held liable for possible errors or the consequences thereof.

Contents

About This Guide iv

1 Changing the SUSE OpenStack Cloud Dashboard Theme 1

2 Managing Images 2

- 2.1 Image Requirements 3
 - General Image Requirements 3 • Image Requirements Depending on Hypervisor 3 • Images for Use With Multiple Hypervisors 4
- 2.2 Building Images with SUSE Studio 5
- 2.3 Image Properties 6
- 2.4 Uploading Images 7
- 2.5 Modifying Image Properties 10
- 2.6 Viewing Images and Image Properties, Deleting Images 12
- 2.7 Viewing and Modifying Membership of Private Images 13

3 Launching Instances from the SUSE OpenStack Cloud Dashboard 14

- 3.1 Key Parameters 14
- 3.2 Launching Instances from Images or Snapshots 15
- 3.3 Launching Instances from Volumes 16

4 Configuring Access to the Instances 19


- 4.1 Security Group Rules 20

About This Guide

SUSE® OpenStack Cloud is an open source software solution that provides the fundamental capabilities to deploy and manage a cloud infrastructure based on SUSE Linux Enterprise. SUSE OpenStack Cloud is powered by OpenStack, the leading community-driven, open source cloud infrastructure project. It seamlessly manages and provisions workloads across a heterogeneous cloud environment in a secure, compliant, and fully-supported manner. The product tightly integrates with other SUSE technologies and with the SUSE maintenance and support infrastructure.

This guide is a supplement to the SUSE OpenStack Cloud *Admin User Guide* and SUSE OpenStack Cloud *End User Guide*. It contains additional information for admin users and end users guides that is specific to SUSE OpenStack Cloud.

Many chapters in this manual contain links to additional documentation resources. These include additional documentation that is available on the system and documentation available on the Internet.

For an overview of the documentation available for your product and the latest documentation updates, refer to <http://www.suse.com/documentation> .

1 Available Documentation

The following manuals are available for this product:

Supplement to Admin User Guide and End User Guide

Gives an introduction to the SUSE® OpenStack Cloud architecture, lists the requirements, and describes how to set up, deploy, and maintain the individual components. Also contains information about troubleshooting, support, and a glossary listing the most important terms and concepts for SUSE OpenStack Cloud.

Admin User Guide

Guides you through management of projects and users, images, flavors, quotas, and networks. Also describes how to migrate instances.

To complete these tasks, either use the graphical Web interface (based on OpenStack Dashboard, code name Horizon) or the OpenStack command line clients.

End User Guide

Describes how to manage images, instances, networks, volumes, and track usage.

To complete these tasks, either use the graphical Web interface (based on OpenStack Dashboard, code name Horizon) or the OpenStack command line clients.

Supplement to Admin User Guide and End User Guide

A supplement to the SUSE OpenStack Cloud *Admin User Guide* and *SUSE OpenStack Cloud End User Guide*. It contains additional information for admin users and end users guides that is specific to SUSE OpenStack Cloud.

HTML versions of the product manuals can be found in the installed system under /usr/share/doc/manual. Additionally, you can access the product-specific manuals and the upstream documentation from the *Help* links in the graphical Web interfaces. Find the latest documentation updates at <http://www.suse.com/documentation> where you can download the manuals for your product in multiple formats.

2 Feedback

Several feedback channels are available:

Bugs and Enhancement Requests

To report bugs for a product component, or to submit enhancement requests, use <https://bugzilla.suse.com/>. For documentation bugs, submit a bug report for the component *Documentation* of the respective product.

If you are new to Bugzilla, you might find the following articles helpful:


- http://en.opensuse.org/openSUSE:Submitting_bug_reports
- http://en.opensuse.org/openSUSE:Bug_reporting_FAQ

Services and Support Options

For services and support options available for your product, refer to <http://www.suse.com/support/>.

User Comments/Bug Reports

We want to hear your comments about and suggestions for this manual and the other documentation included with this product. If you are reading the HTML version of this guide, use the Comments feature at the bottom of each page in the online documentation at <http://www.suse.com/documentation/>.

If you are reading the single-page HTML version of this guide, you can use the *Report Bug* link next to each section to open a bug report at <https://bugzilla.suse.com/> . A user account is needed for this.

Mail

For feedback on the documentation of this product, you can also send a mail to doc-team@suse.de. Make sure to include the document title, the product version, and the publication date of the documentation. To report errors or suggest enhancements, provide a concise description of the problem and refer to the respective section number and page (or URL).

3 Documentation Conventions

The following notices and typographical conventions are used in this documentation:



Warning

Vital information you must be aware of before proceeding. Warns you about security issues, potential loss of data, damage to hardware, or physical hazards.



Important

Important information you should be aware of before proceeding.



Note

Additional information, for example about differences in software versions.



Tip

Helpful information, like a guideline or a piece of practical advice.

- `tux > command`

Commands that can be run by any user, including the root user.

- `root # command`

Commands that must be run with root privileges. Often you can also prefix these commands with the sudo command to run them.

- /etc/passwd: directory names and file names
- PLACEHOLDER: replace PLACEHOLDER with the actual value
- PATH: the environment variable PATH
- ls, --help: commands, options, and parameters
- user: users or groups
- Alt, Alt-F1: a key to press or a key combination; keys are shown in uppercase as on a keyboard
- *File*, *File > Save As*: menu items, buttons
- *Dancing Penguins* (Chapter *Penguins*, ↑Another Manual): This is a reference to a chapter in another manual.

4 About the Making of This Manual

This documentation is written in SUSEDoc, a subset of DocBook 5 (<http://www.docbook.org>)⁷. The XML source files were validated by jing [<https://code.google.com/p/jing-trang/>], processed by xsltproc, and converted into XSL-FO using a customized version of Norman Walsh's stylesheets. The final PDF is formatted through FOP [<https://xmlgraphics.apache.org/fop>] from Apache Software Foundation. The open source tools and the environment used to build this documentation are provided by the DocBook Authoring and Publishing Suite (DAPS). The project's home page can be found at <https://github.com/openSUSE/daps>⁷.

The XML source code of this documentation can be found at <https://github.com/SUSE/doc-cloud>⁷.

1 Changing the SUSE OpenStack Cloud Dashboard Theme

The SUSE OpenStack Cloud Dashboard theme can now be customized. The default SUSE OpenStack Cloud theme is available in the openstack-dashboard-theme-SUSE package. If you want to replace it with a custom theme, you can explore the package contents as an example. When using a custom theme, make sure the resulting package name starts with openstack-dashboard-theme-. Apart from that, you need to adjust the *site_theme* attribute of the Horizon proposal in Crowbar accordingly.

2 Managing Images

In the SUSE OpenStack Cloud context, images are virtual disk images that represent the contents and structure of a storage medium or device, such as a hard disk, in a single file. Images are used as a template from which a virtual machine can be started. For starting a virtual machine, SUSE OpenStack Cloud always uses a copy of the image.

Images have both content and metadata; the latter are also called image properties.

Permissions to manage images are defined by the cloud operator during setup of SUSE OpenStack Cloud. Image upload and management may be restricted to cloud administrators or cloud operators only.

Managing images for SUSE OpenStack Cloud requires the following basic steps:

1. *Building Images with SUSE Studio.*

For general and hypervisor-specific requirements, refer to *Section 2.1, “Image Requirements”*.

2. *Uploading Disk Images to SUSE OpenStack Cloud.*

Images can either be uploaded to SUSE OpenStack Cloud with the **glance** command line tool or with the SUSE OpenStack Cloud Dashboard. As the Dashboard comes with some limitations with regards to image upload and modification of properties, it is recommended to use the **glance** command line tools for comprehensive image management.

3. **Specifying Image Properties.** You can do so during image upload (using **glance image-create**) or with **glance image-update** after the image has already been uploaded. Refer to *Procedure 2.2, “Uploading Disk Images to SUSE OpenStack Cloud”* and *Procedure 2.3, “Modifying Image Properties”*.



Important: Properties for Architecture, Hypervisor and VM Mode

OpenStack Image does *not* check nor automatically detect any image properties. Therefore you need to specify the image's properties manually.

This is especially important when using mixed virtualization environments to make sure that an image is only launched on appropriate hypervisors. The properties can specify a certain architecture, hypervisor type, or application binary interface (ABI) that the image requires.

2.1 Image Requirements

To build the images to use within the cloud, use SUSE Studio or SUSE Studio Onsite as they provide automatic insertion of management scripts and agents. Make sure any images that you build for SUSE OpenStack Cloud fulfill the following requirements.

2.1.1 General Image Requirements

- The network is set to DHCP.
- The image does not include YaST2 Firstboot.
- The image does not include any end-user license agreement (EULA) dialogs.
- The image contains the `cloud-init` package. The package will be automatically added to the image if the following check box in SUSE Studio or SUSE Studio Onsite is enabled: *Integrate with SUSE OpenStack Cloud/OpenStack*.
The `cloud-init` package contains tools used for communication with the instance metadata API, which is provided by Compute. The metadata API is only accessible from inside the VM. The package is needed to pull keypairs into the virtual machine that will run the image.

If you intend to manage the image by the Orchestration module, you also need to include the following package: `openstack-heat-cfn-tools` (part of the SUSE OpenStack Cloud ISO).

2.1.2 Image Requirements Depending on Hypervisor

For a list of supported VM guests, refer to the SUSE® Linux Enterprise Server *Virtualization Guide*, section *Supported VM Guests*. It is available at https://www.suse.com/documentation/sles-12/book_virt/data/virt_support_guests.html.

Depending on the virtualization platform on which you want to use the image, make sure the image also fulfills the following requirements.

KVM

Appliance format: If you are using SUSE Studio or SUSE Studio Onsite 1.3 to build images, use the SUSE OpenStack Cloud/OpenStack/KVM (.qcow2) format.

Xen

Appliance format: If you are using SUSE Studio or SUSE Studio Onsite 1.3 to build images, use the SUSE OpenStack Cloud/OpenStack/KVM (.qcow2) format.

VMware

Appliance format: If you are using SUSE Studio or SUSE Studio Onsite 1.3 to build images, use the VMware/VirtualBox/KVM (.vmdk) format.

If you are using SUSE Studio or SUSE Studio Onsite to build images, the resulting image will be a monolithic sparse file.



Note: Image Conversion

Sparse images can be uploaded to OpenStack Image. However, it is recommended to convert sparse images into a different format before uploading them to OpenStack Image (because starting VMs from sparse images may take longer).

For a list of supported image types, refer to <http://docs.openstack.org/liberty/config-reference/content/vmware.html>, section *Supported image types*.

For details on how to convert a sparse image into different formats, refer to <http://docs.openstack.org/liberty/config-reference/content/vmware.html>, section *Optimize images*.

Hyper-V

Appliance format: If you are using SUSE Studio or SUSE Studio Onsite 1.3 to build images, use the Hyper-V (.vhd), or SUSE OpenStack Cloud/OpenStack/KVM (.qcow2) format. As Hyper-V only supports images in *.vhd format, convert any qcow2 images before uploading them to SUSE OpenStack Cloud. For details, refer to *Procedure 2.1*.

To ensure that an image is bootable on Hyper-V, check the INITRD_MODULES lines in /etc/sysconfig/kernel for the parameters hv_storvsc and mkinitrd. If the parameters are missing, add them. Otherwise the Kernel module for storage devices will not be loaded and the root file system will not be available.

2.1.3 Images for Use With Multiple Hypervisors

If you build the images for SUSE OpenStack Cloud in SUSE Studio or SUSE Studio Onsite, they are compatible with multiple hypervisors by default—even if you may need to convert the resulting image formats before uploading them to OpenStack Image. See *Procedure 2.1, “Converting Disk Images to Different Formats”* for details.

If your image is not made in SUSE Studio or SUSE Studio Onsite, configure the image as follows to create an image that can be booted on KVM, Xen, and Hyper-V, for example:

/etc/sysconfig/kernel

```
INITRD_MODULES="virtio_blk virtio_pci ata_piix ata_generic hv_storvsc"
```

The resulting disk device will be called /dev/sda on Hyper-V.

/boot/grub/menu.lst

To name the partition that should be booted, use:

```
root=UUID=...
```

To find the respective UUID value to use, execute the following command:

```
tune2fs -l /dev/sda2 | grep UUID
```

/etc/fstab

Do not use device names (/dev/...), but UUID=... or LABEL=root entries. For the latter, add the label root to the root file system of your image (in this case, /dev/sda2):

```
tune2fs -L root /dev/sda2
```

Disk Format

Use *.qcow2 as disk format for your image.

Image Properties in OpenStack Image

To upload the image to SUSE OpenStack Cloud only once and to use the same image for KVM, Xen, and Hyper-V, specify the following image options during or after upload:

```
--is-public=True --container-format=bare \  
--property architecture=x86_64 \  
--property vm_mode=hvm \  
--disk-format=qcow2
```

2.2 Building Images with SUSE Studio

When creating an appliance for SUSE OpenStack Cloud the following steps are essential:

1. In SUSE Studio or SUSE Studio Onsite, switch to the *Configuration > Appliance* tab.
2. Enable the *Integrate with SUSE OpenStack Cloud/OpenStack* check box.

3. On the *Build* tab, choose the respective appliance format. It mainly depends on the hypervisor on which you want to use the image—see *Section 2.1.2, “Image Requirements Depending on Hypervisor”*.

For more detailed information on how to build appliance images, refer to the *SUSE Studio Onsite Quick Start* or the *SUSE Studio Onsite User Guide*, available at http://www.suse.com/documentation/suse_studio/.

2.3 Image Properties

Images have both contents and metadata; the latter are also called properties. The following properties can be attached to an image in SUSE OpenStack Cloud. Set them from the command line when uploading or modifying images.

IMAGE PROPERTIES

Name (`--name` , optional)

Specifies a name with which the image will be listed in the SUSE OpenStack Cloud Dashboard and in the command line interface.

Kernel ID (optional)

The image's kernel ID. This parameter is only needed if an external Kernel is associated with the image. The ID points to the Kernel glance image.

Ramdisk ID (optional)

The image's ramdisk ID. This parameter is only needed if an external ramdisk is associated with the image. The ID points to the ramdisk glance image.

Container Format (`--container-format` , optional)

Indicates if the VM image's file format contains metadata about the actual virtual machine. Set it to `bare` as the container format string is not currently used in any OpenStack components anyway. For details, refer to <http://docs.openstack.org/developer/glance/formats.html>.

Disk Format (`--disk-format` , required)

Specifies the image's disk format. Example formats include `raw`, `qcow2`, `ami`, `vhd`, and `vmdk`. For details, refer to <http://docs.openstack.org/developer/glance/formats.html>.

Public (`--is-public` , optional)

Boolean value, default: `false`. If set to `true`, the image is publicly available.

Hypervisor Type (optional)

If your cloud consists of “mixed” hypervisor nodes (KVM, Xen, VMware, Hyper-V), specify at least the hypervisor type the image requires—otherwise it might be scheduled on an incompatible node. For example:

```
hypervisor_type=xen
```

Further examples are: kvm, qemu, vmware, hyperv, xenapi, and powervm.

Architecture (optional)

Specifies the architecture the image requires. For example:

```
architecture=x86_64
```

VM Mode (optional)

Specify the hypervisor ABI (application binary interface) with the vm_mode flag. At the moment, this is only useful for images on the Xen hypervisor. For Xen PV image import, use vm_mode=xen, for Xen HVM image import use vm_mode=hvm. For KVM, VMware, and Hyper-V the correct mode is selected automatically.

2.4 Uploading Images

If you have created an image for SUSE OpenStack Cloud/OpenStack/KVM with SUSE Studio or with SUSE Studio Onsite 1.3, you can upload the image directly as described in *Procedure 2.2, “Uploading Disk Images to SUSE OpenStack Cloud”*.

If you use Hyper-V, convert any qcow2 images before uploading them to SUSE OpenStack Cloud.

PROCEDURE 2.1: CONVERTING DISK IMAGES TO DIFFERENT FORMATS

1. Make sure the virt-utils package is installed on the machine used for conversion.
2. Download the image from SUSE Studio.
3. To convert qcow2 to vhd images, use the following command:

```
qemu-img convert -O vpc CURRENT_IMAGE_FILE FINAL_IMAGE_FILE.vhd
```

PROCEDURE 2.2: UPLOADING DISK IMAGES TO SUSE OPENSTACK CLOUD

Upload a disk image using the glance command line tool that is contained in the package python-glanceclient.

Images are owned by projects and can be private (accessible to members of the particular project only) or public (accessible to members of all projects). Private images can also be explicitly shared with other projects, so that members of those projects can access the images, too. Any image uploaded to OpenStack Image will get an owner attribute. By default, ownership is set to the primary project of the user that uploads the image.

Set or modify hypervisor-specific properties with the `--property key=value` option. This can be done directly during image upload (as shown in the examples below). To change the properties after image upload, refer to *Procedure 2.3, "Modifying Image Properties"*.

1. In a shell, source the OpenStack RC file for the project that you want to upload an image to. For details, refer to http://docs.openstack.org/user-guide/common/cli_set_environment_variables_using_openstack_rc.html.
2. Upload the image using **glance image-create**. Find some example commands for different hypervisors below:

- For KVM:

```
glance image-create --name="IMAGE_NAME" --progress \  
  --is-public=True --container-format=bare \  
  --property architecture=x86_64 \  
  --property hypervisor_type=kvm \  
  --disk-format=qcow2 < PATH_TO_FINAL_IMAGE_FILE.qcow2
```

- For Xen:

```
glance image-create --name="IMAGE_NAME" --progress \  
  --is-public=True --container-format=bare \  
  --property architecture=x86_64 \  
  --property hypervisor_type=xen \  
  --property vm_mode=xen \  
  --disk-format=qcow2 < PATH_TO_FINAL_IMAGE_FILE.qcow2
```



Note: Value of `vm_mode`

For Xen PV image import, use `vm_mode=xen`, for Xen HVM image import use `vm_mode=hvm`.

- For Hyper-V:

```
glance image-create --name="IMAGE_NAME" --progress \  
  --is-public=True --container-format=bare \  
  --property architecture=x86_64 \  
  --property hypervisor_type=vmtoolsd \  
  --disk-format=qcow2 < PATH_TO_FINAL_IMAGE_FILE.qcow2
```

```
--is-public=True --container-format=bare \  
--property architecture=x86_64 \  
--property hypervisor_type=hyperv \  
--property vm_mode=hvm \  
--disk-format=vhd < PATH_TO_FINAL_IMAGE_FILE.vhd
```

- For VMware:

```
glance image-create --name="IMAGE_NAME" --progress \  
--is-public=true --container-format=bare \  
--property vmware_adaptype="lsiLogic" \  
--property vmware_disktype="preallocated" \  
--property hypervisor_type=vmware \  
--disk-format=vmdk --file PATH_TO_FINAL_IMAGE_FILE.vmdk
```



Note: Value of vmware_disktype

Depending on which disk type you use, adjust the value of `vmware_disktype` accordingly. For an overview of which values to use, refer to <http://docs.openstack.org/liberty/config-reference/content/vmware.html>, Table 2.8. *OpenStack Image Service disk type settings*.

- For Docker:

Find an image in the Docker registry you want to use and save it locally with **`docker pull IMAGE_NAME`**, where `IMAGE_NAME` is the name of the image in the Docker registry. The same name needs to be used with the `--name` parameter when uploading the image with the following command:

```
docker save IMAGE_NAME | glance image-create \  
--is-public=true --property hypervisor_type=docker \  
--container-format=docker --disk-format=raw --name "IMAGE_NAME"
```

! Important: Docker Images Need to Run a Long-Living Process

Docker instances will only be able to spawn successfully, when running a long-living process, for example `sshd`. Such a process can be configured with `CMD` (<https://docs.docker.com/engine/reference/builder/#cmd>) or `ENTRYPOINT` (<https://docs.docker.com/engine/reference/builder/#entrypoint>) in the `Docker`.

Alternatively, such a process can be specified on the command line with the `glance image` property `os_command_line`.

```
glance image-update --property os_command_line='/usr/sbin/sshd -D' \  
IMAGE_ID
```

If the image upload has been successful, a message appears, displaying the ID that has been assigned to the image.



Note: Updating Images

After having uploaded an image to SUSE OpenStack Cloud, the image contents cannot be modified—only the image's metadata, see [Procedure 2.3](#). To update image contents, you need to delete the current image and upload a modified version of the image. You can also launch an instance from the respective image, change it, create a snapshot of the instance and use the snapshot as a new image.

2.5 Modifying Image Properties

Set or modify hypervisor-specific properties with the `--property key=value` option. This can be done directly during image upload (see [Procedure 2.2](#)) or after the image has been uploaded (as described below).

PROCEDURE 2.3: MODIFYING IMAGE PROPERTIES

1. In a shell, source the OpenStack RC file for the project that you want to upload an image to. For details, refer to http://docs.openstack.org/user-guide/common/cli_set_environment_variables_using_openstack_rc.html.

2. If you do not know the ID or the exact name of the image whose properties you want to modify, look it up with:

```
glance image-list
```

3. Use the **glance image-update** command to set the properties for architecture, hypervisor type, and virtual machine mode. In the following, find some examples with properties for different hypervisors:

- For KVM:

```
glance image-update IMAGE_ID_OR_NAME \  
  --property architecture=x86_64 \  
  --property hypervisor_type=kvm
```

- For Xen:

```
glance image-update IMAGE_ID_OR_NAME \  
  --property architecture=x86_64 \  
  --property hypervisor_type=xen \  
  --property vm_mode=xen
```



Note: Value of vm_mode

For Xen PV image import, use vm_mode=xen, for Xen HVM image import use vm_mode=hvm.

- For VMware:

```
glance image-update IMAGE_ID_OR_NAME \  
  --property vmware_adaptype="lsiLogic" \  
  --property vmware_disktype="preallocated" \  
  --property hypervisor_type=vmware
```



Note: Value of `vmware_disktype`

Depending on which disk type you use, adjust the value of `vmware_disktype` accordingly. For an overview of which values to use, refer to <http://docs.openstack.org/liberty/config-reference/content/vmware.html>, *Table 2.8. OpenStack Image Service disk type settings*.

- For Hyper-V:

```
glance image-update IMAGE_ID_OR_NAME \  
  --property architecture=x86_64 \  
  --property hypervisor_type=hyperv \  
  --property vm_mode=hvm
```

For more information about the `architecture`, `hypervisor_type`, and `vm_mode` properties, refer to <http://docs.openstack.org/image-guide/content/image-metadata.html>.

2.6 Viewing Images and Image Properties, Deleting Images

In the following, find some examples on how to view images or image properties or how to remove images from OpenStack Image.

Listing Images

```
glance image-list
```

Lists ID, name, disk format, and container format for all images in Image that the current user can access.

Showing Metadata for a Particular Image

```
glance image-show IMAGE_ID_OR_NAME
```

Shows metadata of the specified image.

Removing Image Properties

```
glance image-update IMAGE_ID_OR_NAME --purge-props
```

Deleting an Image

```
glance image-delete IMAGE_ID_OR_NAME
```

Removes the specified image from OpenStack Image.

2.7 Viewing and Modifying Membership of Private Images

In the following, find some examples on how to view or modify membership of private images:

Listing Members a Private Image is Shared With

```
glance member-list --image-id IMAGE_ID
```

Lists the IDs of the projects whose members have access to the private image.

Listing Private Images Shared With a Member

```
glance member-list --tenant-id PROJECT_ID
```

Lists the IDs of private images that members of the specified project can access.

Granting Members Access to a Private Image

```
glance member-create [--can-share] IMAGE_ID_OR_NAME PROJECT_ID_OR_NAME
```

Grants the specified member access to the specified private image.

By adding the `--can-share` option, you can allow the members to further share the image.

Revoking Member Access to a Private Image

```
glance member-delete IMAGE_ID_OR_NAME PROJECT_ID_OR_NAME
```

Revokes the access of the specified member to the specified private image.

3 Launching Instances from the SUSE OpenStack Cloud Dashboard

Instances are virtual machines that run inside the cloud. To start an instance, a virtual machine image must exist that contains the following information: which operating system to use, a username and password with which to log in to the instance, file storage, etc. The cloud contains a pool of such images that have been uploaded to OpenStack Image and are accessible to members of different projects.

3.1 Key Parameters

When starting an instance, specify the following key parameters:

Flavor

In OpenStack, flavors define the compute, memory, and storage capacity of nova computing instances. To put it simply, a flavor is an available hardware configuration for a server. It defines the “size” of a virtual server that can be launched.

For more details and a list of default flavors available, refer to the *Admin User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Manage Flavors*. The guide is available from <http://www.suse.com/documentation>.

Key Pair (optional, but recommended)

Key Pairs are SSH credentials that are injected into images when they are launched. For this to work, the image must contain the cloud-init package.

It is recommended to create at least one key pair per project. If you already have generated a key pair with an external tool, you can import it into OpenStack. The key pair can be used for multiple instances belonging to that project.

For details, refer to the *End User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Configure access and security for instances*. The guide is available from <http://www.suse.com/documentation>.

Security Group

In SUSE OpenStack Cloud, security groups are used to define which incoming network traffic should be forwarded to instances. Security groups hold a set of firewall policies (security group rules).

For details, refer to the *End User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Configure access and security for instances*. The guide is available from <http://www.suse.com/documentation>.

Network

Instances can belong to one or multiple networks. By default, each instance is given a fixed IP address, belonging to the internal network.

Boot Source of the Instance

You can launch instances from the following sources:

- Images that have been uploaded to SUSE OpenStack Cloud—see the *End User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Launch and manage instances* or *Launch an instance from an image*, respectively. The guide is available from <http://www.suse.com/documentation>.
- Volumes that contain images—see the *End User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Launch and manage instances* or *Launch an instance from a volume*. The guide is available from <http://www.suse.com/documentation>.
- Instance snapshots—see the *End User Guide*, chapter *OpenStack dashboard* or chapter *OpenStack command-line clients*, section *Launch and manage instances* or *Create volume from image and boot instance*, respectively. The guide is available from <http://www.suse.com/documentation>.
- Volume snapshots.

If needed, you can assign a floating (public) IP address to a running instance and attach a block storage device (volume) for persistent storage. For details, refer to *Allocate a floating IP address to an instance* (http://docs.openstack.org/user-guide/content/Launching_Instances_using_Dashboard.html)[↗] and *Create and manage volumes* (http://docs.openstack.org/user-guide/content/dashboard_manage_volumes.html)[↗].

3.2 Launching Instances from Images or Snapshots

For instructions on how to launch instances from images or snapshots, see *Launch an Instance* (http://docs.openstack.org/user-guide/content/dashboard_launch_instances_from_image.html)[↗].

3.3 Launching Instances from Volumes

If you want to launch an instance from a volume, the volume needs to contain an image. You can directly create a volume from an image, as described in *Create a volume* (http://docs.openstack.org/user-guide/content/dashboard_manage_volumes.html#dashboard_create_volumes)⁷, by using the respective image as volume source.

In case the image you want to transfer to a volume is not available in OpenStack Image yet, you can create an empty volume and manually copy an image to it as shown in *Procedure 3.1*.

PROCEDURE 3.1: MANUALLY PREPARING A BOOTABLE VOLUME

1. Log in to SUSE OpenStack Cloud Dashboard and select a project from the drop-down box at the top-level row.
2. Create an empty volume as described in *Create a volume* (http://docs.openstack.org/user-guide/content/dashboard_manage_volumes.html#dashboard_create_volumes)⁷, by using the option *No source, empty volume*. The volume size must be large enough to store an unzipped image.
3. Create an image with SUSE Studio or SUSE Studio Onsite. For details, refer to *Section 2.2, "Building Images with SUSE Studio"*.
4. To be able to copy the image to the empty volume, you need to execute the following steps first:
 - a. Launch an instance from an arbitrary image as described in *Launch an Instance* (http://docs.openstack.org/user-guide/content/dashboard_launch_instances_from_image.html)⁷.



Note

- Which image you select there is only important if you want to boot a *Xen* image from the volume later on. In that case, the image you select in *Launch an Instance* (<http://docs.openstack.org/user-guide/>)

[content/dashboard_launch_instances_from_image.html](http://docs.openstack.org/user-guide/content/dashboard_launch_instances_from_image.html) ↗, Step 3 needs to be of the same type as the one on the volume: fully virtualized or paravirtualized.

- In any other case the image you select for starting the instance does *not* matter. It will be replaced by the image on the volume that you specify in *Launch an Instance* (http://docs.openstack.org/user-guide/content/dashboard_launch_instances_from_image.html) ↗, Step 3, by using the option *Boot from volume*.
- Make sure to select an appropriate flavor for the instance. The instance's hard disk must be big enough to store the image that you want to copy onto the volume later on.

b. Attach the empty volume to the running instance as described in *Attach a volume to an instance* (http://docs.openstack.org/user-guide/content/dashboard_manage_volumes.html#dashboard_create_volumes) ↗.

5. Copy the image that you created in *Step 3* to the running instance (for example, by using scp).
6. Log in to the instance by using SSH or the VNC console.
7. Assuming that the attached volume is mounted as /dev/vdb, use one of the following commands to copy the image to the attached volume:

- For a raw image:

```
cat IMAGE >/dev/vdb
```


(alternatively, use dd)


- For a non-raw image:

```
qemu-img convert -O raw IMAGE /dev/vdb
```

- For a *.tar.bz2 image:

```
tar xvj0 IMAGE >/dev/vdb
```

8. As the image comes with a predefined disk size (that might be smaller than the size of the volume it has been copied to), the image will not use the full size of the volume. To change this, adjust the partition table within the image to match the size of the volume.
9. As only *detached* volumes are available for booting, detach the volume. For details on how to do so, refer to *Detach a volume from an instance* (http://docs.openstack.org/user-guide/content/dashboard_manage_volumes.html#dashboard_create_volumes) .

Now, you can launch an instance from the image that has been copied to the volume. The instance will be booted from the volume, which is provided by OpenStack Block Storage through iSCSI. To do so, proceed as described in *Launch an Instance* (http://docs.openstack.org/user-guide/content/dashboard_launch_instances_from_image.html) , using the option *Boot from volume*.

After you have launched an instance, switch to the *Instances* category in the Dashboard. There, you can view the *Instance Name*, the *Image Name*, its (private or public) *IP address*, its *Size*, the *Key Pair* associated with it, the image's *Status*, its *Availability Zone*, its *Task*, its *Power State*, and its *Uptime*.

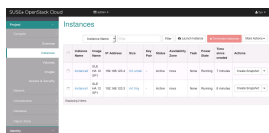



FIGURE 3.1: SUSE OPENSTACK CLOUD DASHBOARD—LIST OF LAUNCHED INSTANCES

If you did not provide a key pair on starting and have not touched security groups or rules so far, by default the instance can only be accessed from inside the cloud via VNC at this point. Even pingging the instance from within the cloud is not possible. To change this, proceed with *Configure access and security for instances* (http://docs.openstack.org/user-guide/content/Launching_Instances_using_Dashboard.html) .

4 Configuring Access to the Instances

Access to an instance is mainly influenced by the following parameters:

Security Groups and Rules

In SUSE OpenStack Cloud, security groups are used to define which incoming network traffic should be forwarded to instances. Security groups hold a set of firewall policies (security group rules).

For instructions on how to configure security groups and security group rules, see *Create a security group* (http://docs.openstack.org/user-guide-admin/content/dashboard_manage_projects_users.html) ↗, *Add a security group rule* (http://docs.openstack.org/user-guide-admin/content/dashboard_manage_projects_users.html) ↗, and *Section 4.1, “Security Group Rules”*.

Key Pairs

Key Pairs are SSH credentials that are injected into images when they are launched. For this to work, the image must contain the `cloud-init` package.


It is recommended to create at least one key pair per project. If you already have generated a key pair with an external tool, you can import it into OpenStack. The key pair can be used for multiple instances belonging to that project.

For details on how to create or import keypairs, see *Add a key pair* and *Import a key pair* (http://docs.openstack.org/user-guide-admin/content/dashboard_manage_projects_users.html) ↗.

IP Addresses

Each instance can have two types of IP addresses: private (fixed) IP addresses and public (floating) ones. Private IP addresses are used for communication between instances, and public ones are used for communication with the outside world. When an instance is launched, it is automatically assigned private IP addresses in the networks to which it is assigned. The private IP stays the same until the instance is explicitly terminated. (Rebooting the instance does not have an effect on the private IP addresses.)

A floating IP is an IP address that can be dynamically added to a virtual instance. In OpenStack Networking, cloud administrators can configure pools of floating IP addresses. These pools are represented as external networks. Floating IPs are allocated from a subnet that is associated with the external network. You can allocate a certain number of floating IPs to a project—the maximum number of floating IP addresses per project is defined by the quota. From this set, you can then add a floating IP address to an instance of the project.

For information on how to assign floating IP addresses to instances, see [Allocate a floating IP address to an instance](http://docs.openstack.org/user-guide/content/Launching_Instances_using_Dashboard.html) (http://docs.openstack.org/user-guide/content/Launching_Instances_using_Dashboard.html) .

4.1 Security Group Rules

You can adjust rules of the default security group and rules of any other security group that has been created. As soon as the rules for a group are modified, the new rules are automatically applied to all running instances belonging to that security group.

Adjust the rules in a security group to allow access to instances via different ports and protocols. This is necessary to be able to access instances via SSH, to ping them, or to allow UDP traffic (for example, for a DNS server running on an instance).

Rules in security groups are specified by the following parameters:

IP Protocol

Protocol to which the rule will apply. Choose between TCP (for SSH), ICMP (for pings), and UDP.

Port/Port Range

For TCP or UDP, define a single port or a port range to open on the virtual machine. ICMP does not support ports. In that case, enter values that define the codes and types of ICMP traffic to be allowed.

Source of traffic (*Remote* in the SUSE OpenStack Cloud Dashboard)

Decide whether to allow traffic to instances only from IP addresses inside the cloud (from other group members) or from *all* IP addresses. Specify either an IP address block (in CIDR notation) or a security group as source. Using a security group as source will allow any instance in that security group to access any other instance.

If no further security groups have been created, any instances are automatically assigned to the default security group (if not specified otherwise). Unless you change the rules for the default group, those instances cannot be accessed from any IP addresses outside the cloud.

PROCEDURE 4.1: CONFIGURING SECURITY GROUP RULES

For quicker configuration, the Dashboard provides templates for rules that are often-used, including rules for well-known protocols on top of TCP (such as HTTP or SSH), or rules to allow all ICMP traffic (for pings).

1. Log in to SUSE OpenStack Cloud Dashboard and select a project from the drop-down box at the top-level row.
2. Click *Project > Compute > Access & Security*. The view shows the following tabs: *Security Groups*, *Key Pairs*, *Floating IPs*, and *API Access*.
3. On the *Security Group* tab, click *Manage Rules* for the security group you want to modify. This opens the *Security Group Rules* screen that shows the existing rules for the group and lets you add or delete rules.
4. Click *Add Rule* to open a new dialog.

From the *Rule* drop-down box, you can select templates for rules that are often-used, including rules for well-known protocols on top of TCP (such as HTTP or SSH), or rules to allow all ICMP traffic (for pings). In the following steps, we will focus on the most commonly-used rules only:
5. To enable SSH access to the instances:
 - a. Set *Rule* to SSH.
 - b. Decide whether to allow traffic to instances only from IP addresses inside the cloud (from other group members) or from *all* IP addresses.
 - To enable access from *all* IP addresses (specified as IP subnet in CIDR notation as 0.0.0.0/0), leave the *Remote* and *CIDR* fields unchanged.
 - Alternatively, allow only IP addresses from other security groups to access the specified port. In that case, set *Remote* to Security Group. Select the desired *Security Group* and *Ether Type* (IPv4 or IPv6).
6. To enable pinging the instances:
 - a. Set *Rule* to ALL ICMP.
 - b. Decide whether to allow traffic to instances only from IP addresses inside the cloud (from other group members) or from *all* IP addresses.

- To enable access from *all* IP addresses (specified as IP subnet in CIDR notation as 0.0.0.0/0), leave the *Remote* and *CIDR* fields unchanged.
 - Alternatively, allow only IP addresses from other security groups to access the specified port. In that case, set *Remote* to Security Group. Select the desired *Security Group* and *Ether Type* (IPv4 or IPv6).
7. To enable access via a UDP port (for example, for syslog):
- a. Set *Rule* to Custom UDP.
 - b. Leave the *Direction* and *Open Port* values untouched.
 - c. In the *Port* text box, enter the value *514*.
 - d. Decide whether to allow traffic to instances only from IP addresses inside the cloud (from other group members) or from *all* IP addresses.
 - To enable access from *all* IP addresses (specified as IP subnet in CIDR notation as 0.0.0.0/0), leave the *Remote* and *CIDR* fields unchanged.
 - Alternatively, allow only IP addresses from other security groups to access the specified port. In that case, set *Remote* to Security Group. Select the desired *Security Group* and *Ether Type* (IPv4 or IPv6).