SUSE Cloud

1.0

December 04, 2012

User Guide for Administrators

User Guide for Administrators

List of Authors: Tanja Roth, Frank Sundermeyer

Contents

About This Guide

SUSE® 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 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 helps cloud administrators in the execution of their daily tasks like managing projects and users, images, or flavors, and setting quotas. Most of these tasks can either be achieved with the Web interface (the SUSE Cloud Dashboard) or the OpenStack command line tools.

Many chapters in this manual contain links to additional documentation resources. These include additional documentation that is available on the system as well as 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/suse_cloud10.

For the documentation on SUSE Linux Enterprise Server and Subscription Management Tool (SMT) and the latest documentation updates, refer to http://www.suse.com/documentation/.

1 Available Documentation

The following manuals are available for this product:

Deployment Guide (↑*Deployment Guide*)
> Gives an introduction to the SUSE® Cloud architecture and describes how to set up, deploy, and maintain the individual components.

User Guide for Administrators (page i)
>Guides you through management of projects and users, images, flavors, and quotas with SUSE Cloud Dashboard or the command line interface.

End User Guide (↑*End User Guide*)
>Describes how to launch instances, manage volumes, and track usage.

HTML versions of the product manuals can be found in the installed system under `/usr/share/doc/manual`. 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
>For services and support options available for your product, refer to http://www.suse.com/support/.
>
>To report bugs for a product component, log in to the Novell Customer Center from http://www.suse.com/support/ and select *My Support > Service Request*.

User Comments
>We want to hear your comments about and suggestions for this manual and the other documentation included with this product. Use the User Comments feature at the bottom of each page in the online documentation or go to http://www.suse.com/documentation/feedback.html and enter your comments there.

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 typographical conventions are used in this manual:

- `/etc/passwd`: directory names and filenames

- *`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

- This paragraph is only relevant for the architectures `amd64`, `em64t`, and `ipf`. The arrows mark the beginning and the end of the text block.

  This paragraph is only relevant for the architectures `System z` and `ipseries`. The arrows mark the beginning and the end of the text block.

- *Dancing Penguins* (Chapter *Penguins*, ↑Another Manual): This is a reference to a chapter in another manual.

4 About the Making of This Manual

This book is written in Novdoc, a subset of DocBook (see http://www.docbook.org). The XML source files were validated by `xmllint`, processed by `xsltproc`, and converted into XSL-FO using a customized version of Norman Walsh's stylesheets. The final PDF is formatted through XEP from RenderX.

Using SUSE Cloud Dashboard

1

The SUSE® Cloud Dashboard is a Web interface that allows cloud administrators and users to manage various OpenStack services. It is based on OpenStack Dashboard (also known under its codename `Horizon`).

After a short introduction to the Dashboard, learn how to execute key administration tasks such as managing projects and users, adding images, adding flavors, and setting quotas.

1.1 Requirements

The following requirements need to be fulfilled to access the SUSE Cloud Dashboard:

- The cloud operator has set up SUSE Cloud.

- You have a recent Web browser that supports HTML5. It must have cookies and JavaScript enabled. For using the Dashboard's VNC client, which is based on `noVNC`, your browser needs to support HTML5 Canvas and HTML5 WebSockets. For more details and a list of browsers that support `noVNC`, refer to `https://github.com/kanaka/noVNC/blob/master/README.md`, and `https://github.com/kanaka/noVNC/wiki/Browser-support`, respectively.

1.2 SUSE Cloud Dashboard—Overview

Learn how to log in to SUSE Cloud Dashboard and get a short overview of its Web interface.

1.2.1 Logging in to the SUSE Cloud Dashboard

To access the SUSE Cloud Dashboard, ask the cloud operator for the following information:

• Hostname or (public) IP address of the SUSE Cloud Dashboard. (The Dashboard is available on the node that has the `nova-dashboard server` role.)

• Username and password of the cloud administrator or cloud user with which you can log in to SUSE Cloud Dashboard.

1 Start a Web browser and make sure that JavaScript and cookies are enabled.

2 As a URL, enter the hostname or IP address that you got from the cloud operator.

```
https://IP_ADDRESS_OR_HOSTNAME/
```

NOTE: Certificate Warning

Depending on your browser and browser options, you may get a certificate warning when trying to access the URL for the first time. (In case no certificate is provided when setting up the Dashboard, SUSE Cloud uses a self-signed certificate that is not considered trustworthy by default).

In this case, verify the certificate.

To proceed anyway, you can add an exception in the browser to bypass the warning.

---

3 On the SUSE Cloud Dashboard login screen, enter the *User Name* and *Password* and click *Sign In*.

Figure 1.1: *SUSE Cloud Dashboard—Login Screen*

After logging in, the Dashboard's Main Screen (Administrator's View) appears.

1.2.2 Main Screen (Administrator's View)

The top-level row of the main screen shows the username with which you are logged in. It also allows you to access the *Settings* or to *Sign Out* of the Web interface.

NOTE: Available Functions

The visible tabs and functions in the Dashboard depend on the access permissions of the user that is logged in. They are defined by roles.

If you are logged in as an administrator, the main screen shows the following tabs: *Project* and an *Admin*.

1.2.2.1 Main Screen—*Project* Tab

The *Project* tab shows details for the projects (or `tenants`) that the user who is logged in belongs to.

Figure 1.2:   *SUSE Cloud Dashboard—Project Tab*

Select a *Project* from the drop-down list on the left-hand side to access the following categories:

Overview
> Shows basic reports on the project.

Instances & Volumes
> Lists instances and volumes created by users of the project. From here, you can terminate, pause, or reboot any instances or connect to them via VNC.
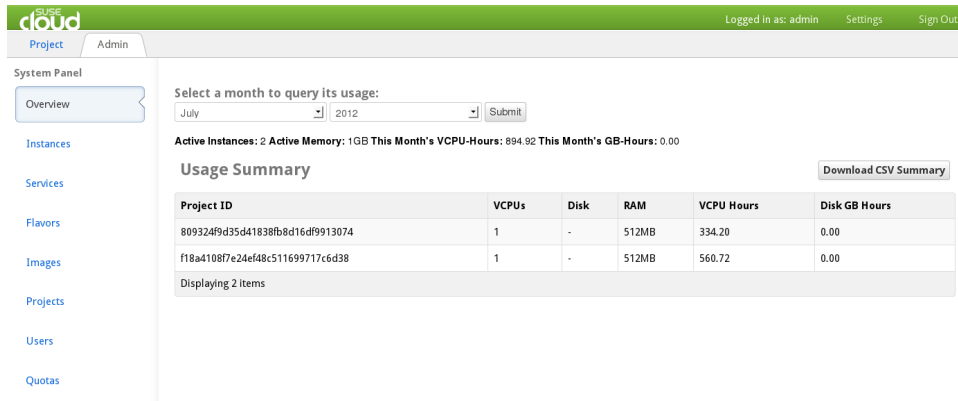
Images & Snapshots
> Lists images and snapshots created by users of the project, plus any images that are publicly available.

Access & Security
> Allows to allocate or release floating IP addresses, manage security groups and keypairs.

1.2.2.2 Main Screen—*Admin* Tab

Figure 1.3: *SUSE Cloud Dashboard—Admin Tab*

On this tab, you can access the following categories:

Overview
Shows basic reports.

Instances
Lists all currently running instances belonging to various users and tenants. (Not all tenants may be visible to the administrator, though.)

Services
Lists the defined services.

Flavors
Lists the available "sizes" of the VMs that users may launch.

Images
Shows the custom images that have been uploaded. Lets you edit image properties or delete images, if needed.

Project
Lists the available tenants. From here, you can create new projects and assign users to the projects.

Users
> Gives an overview of all users.

Quotas
> Lists the default quota values (hard-coded in OpenStack Nova). Includes parameters such as the number of CPUs, RAM, or instances.

1.3 Managing Projects and Users

SUSE Cloud allows you to manage projects independently from each other. Projects represent different organizational units in the cloud to which users can be assigned. Both project and user management are cloud administrator's tasks.

During the basic system setup, the cloud operator needs to minimally define one project, one user, and one role to link the project and user. Roles define which actions users are allowed to perform.

As a SUSE Cloud administrator, you can create additional projects and users as needed. The following procedures guide you through the main management tasks like adding, modifying, or deleting projects and users. Learn how to assign users to one or multiple projects, or how to change or remove the assignment.

Procedure 1.1: *Creating or Deleting A Project*

Projects can be created, deleted, or temporarily disabled by cloud administrators. Disabling a project has the following consequences:

Consequences of Disabling a Project

- In the SUSE Cloud Dashboard, the project can no longer be accessed from the *Project* drop-down list on the *Project* tab.

- Users that are only members of the disabled project can no longer log in.

- It is impossible to launch new instances for a disabled project. Instances already running are not automatically terminated though—you have to stop them manually.

- All data of a disabled project is kept so that the project can be re-enabled at any time.

1 On the *Admin* tab, select the *Projects* category.

2 To add a new project:

> **2a** Click *Create New Project*.
>
> **2b** In the window that opens, enter a *Name* and *Description* for the project and confirm your changes.
>
> The Dashboard automatically assigns an ID and shows the newly created project in the *Projects* category.

3 To delete one or multiple projects:

> **3a** Activate the check boxes in front of the projects that you want to delete.
>
> **3b** Click *Delete Projects* and confirm your choice in the pop-up that appears.
>
> A message on the Web page shows if the action has been successful.

4 To temporarily deactivate a project: Click *Edit Project* and deactivate the *Enabled* check box.

Figure 1.4: *SUSE Cloud Dashboard—List of Projects (Administrator's View)*
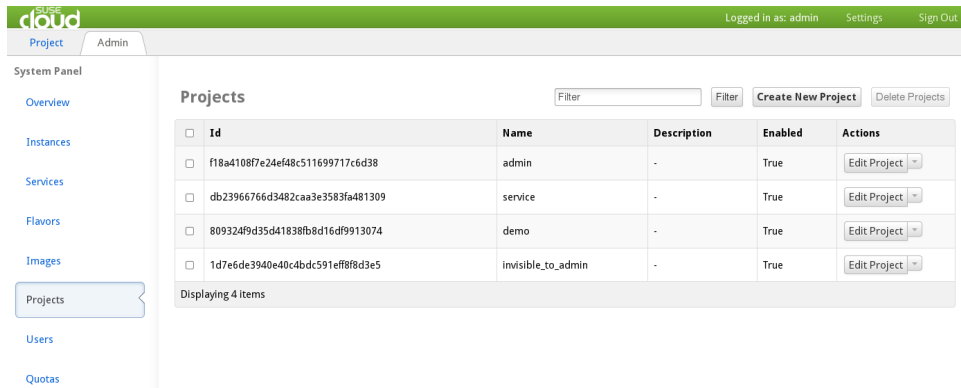
Procedure 1.2: *Creating Or Deleting Users Accounts*

Users are members of one or multiple projects. Both project and user management are cloud administrator's tasks. When a new user account is created, the user has to be assigned to a primary project. The user can also be assigned to additional projects.

User accounts can be created, deleted, or temporarily deactivated. If a user account is deactivated, the user can no longer log in, but his data is kept so that the account can be re-enabled at any time. Before a user account can be deleted, the user needs to be removed from his primary project.

1 On the *Admin* tab, select the *Users* category.

2 To add a new user account:

> **2a** Click *Create User*.
>
> **2b** In the window that opens, enter a *User Name* and an *Email* address for the user.
>
> **2c** Set a preliminary *Password* for the user and confirm it.
>
> **2d** Assign the user to a *Primary Project* by selecting the respective project from the drop-down list.
>
> **2e** Confirm your changes.
>
> The Dashboard automatically assigns an ID and shows the newly created user account in the *Users* category.

3 To delete one or multiple users accounts:

> **3a** Activate the check boxes in front of the user accounts that you want to delete.
>
> **3b** Click *Delete Users* and confirm your choice in the pop-up that appears.
>
> A message on the Web page shows if the action has been successful.

> ---
> **NOTE: Remove User from Primary Project**
>
> If a user account cannot be deleted, it is because it is still assigned to its primary project.
>
> 1. *Edit* the user data to view which *Primary Project* he is assigned to.

2. Remove the user from his primary project as described in Proce-
dure 1.3, "Modifying User Assignments for a Project" (page 9),
Step 1 to Step 4.

4 To temporarily deactivate a user account: Select the user, and from the *Actions* drop-
down list, select *Disable*.

Figure 1.5: *SUSE Cloud Dashboard—List of Users (Administrator's View)*

Procedure 1.3: *Modifying User Assignments for a Project*

When creating new users, you must assign them to a primary project as described in
Procedure 1.2, "Creating Or Deleting Users Accounts" (page 7). To assign users to
additional projects or to modify and remove assignments, proceed as follows:

1 On the *Admin* tab, select the *Projects* category.

2 Select the project for which to modify user assignments.

3 From the *Actions* drop-down list for the project, select *Modify Users*.

The Dashboard shows two lists of users: *Users For Project* shows the users assigned
to the current project, *Add New Users* shows other existing users, which can be as-
signed to the current project.

4 To remove users from the current project, select one or multiple users and click *Remove User* or *Remove Users*.

5 To assign a user to the current project:

5a Select the user and click *Add To Project*.

5b In the window that appears, set the user's role with which to add him to the project and click *Add*. Roles define the actions that the user is allowed to perform. Roles are configured by the cloud operator in OpenStack Identity (Keystone). Actions are defined per OpenStack service in the respective `/etc/[SERVICE_CODENAME]/policy.json` file, for example in `/etc/nova/policy.json` for the Compute (Nova) service. For details, refer to `http://docs.openstack.org/essex/openstack` `-compute/install/content/keystone-concepts.html`.

1.4 Managing Images

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

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

After uploading an image to Nova, it cannot be changed any more ("golden image").

Whereas nearly all key tasks can either be executed from the SUSE Cloud Web interface or from the command line, images can only be uploaded with a command line tool, `glance image-create`. For details, refer to Section 2.4.2, "Adding Images" (page 26).

Images have both contents and meta-data; the latter are also called properties. The following properties can be attached to an image in SUSE Cloud:

Image Properties

Name (`--name`, optional)
> Specifies a name with which the image will be listed in the SUSE 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.

Architecture (optional)
> The image's architecture.

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`, and `ami`. 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.

VM Mode (optional)

Specify the hypervisor ABI (application binary interface) with the `vm_mode` flag. It can take the values `pv`, `hvm`, or `xen`. Use `vm_mode=xen` for XEN PV image import, or `vm_mode=hvm` for XEN HVM image import. For KVM, the correct mode is selected automatically.

After images have been added from the command line, they appear in the SUSE Cloud Dashboard on the *Admin* tab. View them in the *Images* category. From there, you can also *Edit* the image properties.

If you need to delete an image, proceed as follows.

Procedure 1.4: *Deleting Images*

1 On the *Admin* tab, select the *Images* category.

2 To delete one or multiple images, activate the check boxes in front of the images that you want to delete.

3 Click *Delete Images* and confirm your choice in the pop-up that appears.

A message on the Web page shows if the action has been successful.

1.5 Managing Flavors

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.

A flavor consists of the following parameters:

Flavor Parameters

Flavor ID
> Automatically proposed by SUSE Cloud.

Name
> Name for the new flavor.

VCPUs
> Number of virtual CPUs to use.

Memory MB
> Amount of RAM to use (in megabytes).

Root Disk GB
> Amount of disk space (in gigabytes) to use for the root (/) partition.

Ephemeral Disk GB
> Amount of disk space (in gigabytes) to use for the ephemeral partition. If unspecified, the value is `0` by default.
>
> Ephemeral disks offer machine local disk storage linked to the lifecycle of a VM instance. When a VM is terminated, all data on the ephemeral disk is lost. Ephemeral disks are not included in any snapshots.

Swap
> Amount of swap space (in megabytes) to use. If unspecified, the value is `0` by default.

The default flavors are:

Default Flavors

- m1.tiny (1 VCPU/0 GB Disk/512 MB RAM)

- m1.smaller (1 VCPU/0 GB Disk/1024 MB RAM)

- m1.small (1 VCPU/10 GB Disk/2048 MB RAM)

- m1.medium (2 VCPU/10 GB Disk/3072 MB RAM)

- m1.large (4 VCPU/10 GB Disk/8192 MB RAM)

• m1.xlarge (8 VCPU/10 GB Disk/8192 MB RAM)

Via the SUSE Cloud Dashboard, you can create new flavors or delete existing ones.

Procedure 1.5: *Creating or Deleting Flavors*

1 On the *Admin* tab, select the *Flavors* category.

2 To create a new flavor:

    2a Click *Create Flavor*.

    2b In the window that opens, specify the required parameters for the flavor.

    2c Confirm your changes.

    The newly created flavor will appear in the list of flavors and can be used when launching new instances.

3 To delete one or multiple flavors:

    3a Activate the check boxes in front of the flavors that you want to delete.

    3b Click *Delete Flavors* and confirm your choice in the pop-up that appears.

    A message on the Web page shows if the action has been successful.

Figure 1.6: *SUSE Cloud Dashboard—List of Flavors (Administrator's View)*

ID	Flavor Name	VCPUs	Memory	Root Disk	Ephemeral Disk	Actions
5	m1.xlarge	8	16384	10	160	Delete Flavor
4	m1.large	4	8192	10	80	Delete Flavor
3	m1.medium	2	4096	10	40	Delete Flavor
2	m1.small	1	2048	10	20	Delete Flavor
12	smaller	1	768	.	20	Delete Flavor
1	m1.tiny	1	512	.	.	Delete Flavor

Displaying 6 items

1.6 Setting Quotas

To prevent system capacities from being exhausted without notification, cloud administrators can set up quotas. In OpenStack, quotas are currently defined per project.

Quotas contain the following parameters:

Quota Parameters

ID (Name)
    ID for the quota settings. Automatically proposed by SUSE Cloud.

Metadata Items
    Number of metadata items per instance.

Injected Files
    Number of injected files.

Injected File Content Bytes
    Number of bytes per injected file.

VCPUs
    Number of virtual CPUs that can be allocated in total.

Instances
    Total number of instances.

Volumes
    Total number of volumes.

Gigabytes
    Total size of all volumes, measured in gigabytes.

RAM (in MB)
    Total RAM size of all instances, measured in megabytes.

Floating IPs
    Total number of floating IP addresses.

Security Group Rules
    Number of security rules per security group.

Security Groups
>   Number of security groups.

NOTE: Contents of the *Quota* Category

The *Quota* category does not allow you to set any values, it just shows the global default quota values that are hard-coded in OpenStack Nova.

Procedure 1.6: *Setting Quotas for a Project*

1 On the *Admin* tab, select the *Projects* category.

2 Select the project for which to set or change quota values.

3 From the *Actions* drop-down list, select *Modify Quota*.

The window that opens shows the default quota values per project, which are hard-coded in OpenStack Nova.

4 Change the values for the quota parameters as desired.

Update Quota   ×

ID (name)   **Description:**
809324f9d35d41838fb8d16df9913074   From here you can edit quotas (max limits) for
                                       the project .
Metadata Items
128

Injected Files
5

Injected File Content Bytes
10240

VCPUs
20

Instances
10

Volumes
10

Gigabytes
1000

RAM (in MB)
51200

Floating IPs
10

(Cancel) (Update Quota)

5 Confirm your changes.

Using OpenStack Command Line Interfaces

2

The OpenStack project provides a variety of command line tools with which you can manage the services within your cloud and automate tasks by using scripts. Each of the core OpenStack components has its own command line tool.

2.1 OpenStack Commands—Overview

The following command line tools are available for the respective services' APIs:

keystone
> For managing users and projects. Provided by the `python-keystoneclient` package.

nova
> For managing instances and flavors. Provided by the `python-novaclient` package.

glance
> For managing images. Provided by the `python-glanceclient` package.

swift
> For managing the object store. Provided by the `python-swiftclient` package.

All of them have tab completion.

Help and detailed information about the individual commands and their arguments are available with

```
COMMAND help
```

For help on subcommands, use

```
COMMAND help SUBCOMMAND
```

For example: `glance help` or `glance help image-create`

2.2 OpenStack RC File

To set the necessary environment variables for the OpenStack command line tools, you need to download and source an environment file, `openrc.sh`. It is project-specific and contains the credentials used by OpenStack Compute, Image, and Identity services. You can download it from the SUSE Cloud Dashboard (either as user `admin` or as any other user).

Procedure 2.1: *Downloading the OpenStack RC File*

1 Log in to the SUSE Cloud Dashboard.

2 In the top-level row of the main screen, click *Settings > OpenStack Credentials*.

3 Select the project for which you want to download the OpenStack RC file, click *Download RC File* and save the file.

4 Copy the `openrc.sh` file to the machine on which you want to execute OpenStack commands (for example, uploading an image with the `glance` command).

5 On any shell that you want to execute OpenStack commands from, source the `openrc.sh` file for the respective project:

```
source openrc.sh
```

You will be prompted for an OpenStack password.

6 Enter the OpenStack password of the user who downloaded the `openrc.sh` file.

With sourcing the file and entering the password, environment variables are set for that shell. They allow the commands to communicate to the OpenStack services running in the cloud.

2.3 Managing Projects and Users

SUSE Cloud allows you to manage projects independently from each other. Projects represent different organizational units in the cloud to which users can be assigned. Both project and user management are cloud administrator's tasks.

During the basic system setup, the cloud operator needs to minimally define one project, one user, and one role to link the project and user. Roles define which actions users are allowed to perform.

The `python-keystoneclient` provides the `keystone` command line tool which you can use to manage projects and users from any machine outside the cloud. Prior to using the tool, download and source an OpenStack RC file. For details, refer to Section 2.2, "OpenStack RC File" (page 20).

NOTE: Administrator Credentials

Administering projects and users requires administrator credentials. Make sure to download and source the OpenStack RC file as administrator prior to running `keystone` commands. Alternatively, export the respective environment variables, using the token or password authentication method. For details, refer to http://docs.openstack.org/essex/openstack-compute/admin/content/adding-users-tenants-and-roles-with-python-keystoneclient.html

---

2.3.1 Viewing, Creating, Disabling, or Deleting Projects

Find examples for the key administration tasks below.

Listing All Projects

```
keystone tenant-list
```

Lists all projects with their ID, name, and the information if they are enabled or not.

Creating a Project

```
keystone tenant-create --name PROJECT_NAME
```

Creates a new project with the specified name.

Temporarily Disabling a Project

```
keystone tenant-update PROJECT_ID --enabled false
```

For the details of the impact, refer to Consequences of Disabling a Project (page 6)
.

Deleting a Project

```
keystone tenant-delete PROJECT_ID
```

Deletes the specified project.

2.3.2 Viewing, Creating, Disabling, or Deleting User Accounts

Find examples for the key administration tasks below.

Listing All Users

```
keystone user-list
```

Lists all user accounts with their ID, name, e-mail address, and the information if they are enabled or not.

Creating a User Account

```
keystone user-create --name USER_NAME --tenant_id PROJECT_ID \
  --pass PRELIM_PASSWD
```

Creates a new user with the specified name. While the only required argument is
--name, at least specify the optional parameters --tenant_id and --pass.
Otherwise the newly created user cannot log in to the SUSE Cloud Dashboard.

Temporarily Disabling a User Account

```
keystone user-update USER_ID --enabled false
```

If you disable a user account, the user can no longer log in, but his data is kept so that the account can be re-enabled at any time.

Deleting a User Account

```
keystone user-delete USER_ID
```

Deletes the specified user account.

2.3.3 Managing Roles

Roles define the actions that the user is allowed to perform. Configure roles in OpenStack Identity (Keystone). Actions are defined per OpenStack service in the respective `/etc/[SERVICE_CODENAME]/policy.json` file, for example in `/etc/nova/policy.json` for the Compute (Nova) service.

Find examples for the key administration tasks below.

Listing All Roles

```
keystone role-list
```

Lists all roles with their ID and name.

Creating a Role

```
keystone role-create --name=ROLE_NAME
```

Creates a role with the specified name.

Deleting a Role

```
keystone role-delete ROLE_ID
```

Deletes the specified role.

2.3.4 Modifying User Assignments for a Project

Whereas each user is assigned to a primary project when his user account is created, users can be members of multiple projects. The keystone client does not allow to directly assign users to additional projects. Instead you need to define a role and grant that role to a user-project pair.

1 On a shell, source the OpenStack RC file. For details, refer to Section 2.2, "OpenStack RC File" (page 20).

2 Check if there is already a `member` role defined:

```
keystone role-list
```

3 If not, create it:

```
keystone role-create --name=member
```

4 To grant the role to a user-project pair (and to thus assign a user to this project), you need to know the IDs of the role, the user, and the project. You can look them up with `keystone role-list`, `keystone user-list`, and `keystone tenant-list`.

5 To grant the user membership of a project:

```
keystone user-role-add --role-id=ROLE_ID --tenant_id=TENANT_ID \
  --user_id=USER_ID
```

6 To assign the user to multiple projects, repeat the last step.

7 To verify the assignments, use:

```
keystone user-role-list --user_id=USER_ID --tenant_id=TENANT_ID
```

2.4 Managing Images

In the SUSE Cloud context, images are virtual disk images that represent the contents and structure of a storage medium or device, such as a hard drive, in a single file. Images

are used as a template from which a virtual machine can be started. For starting a virtual machine, SUSE Cloud always uses a copy of the image.

Images can only be uploaded to SUSE Cloud with the `glance` command line tool. 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 Glance will get an `owner` attribute. By default, ownership is set to the primary project of the user that uploads the image.

2.4.1 Building Images with SUSE Studio

To build the images to use within the cloud, use SUSE Studio or SUSE Studio Onsite. For 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/.

NOTE: Image Requirements

Make sure any images that you build for the cloud fulfill the following 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 following package: `cloud-init` (part of the SUSE Cloud ISO). It contains tools used for communication with the instance metadata API, which is provided by Nova. The instance metadata API is specific to each virtual machine and is only accessible from inside the VM. The package is needed to pull keypairs into the virtual machine that will run the image.

- The image has one of the following formats: `SUSE Cloud/OpenStack/KVM` or `USB Stick/Hard Disk Image`.

As announced in [http://blog.susestudio.com/2012/10/kvm-build
-format-suse-cloud-support.html](http://blog.susestudio.com/2012/10/kvm-build-format-suse-cloud-support.html), SUSE Studio now supports building native KVM images in `qcow2` format. They can be directly uploaded to SUSE Cloud.

If you are building images with SUSE Studio Onsite, you need to use the format `USB Stick/Hard Disk Image` and convert the image to `qcow2` before uploading it. Future versions of SUSE Studio Onsite will also support building native KVM images in `qcow2` format.

2.4.2 Adding Images

If you have created an image for `SUSE Cloud/OpenStack/KVM` with SUSE Studio, you can upload it directly as described in Procedure 2.3, "Uploading Disk Images to SUSE Cloud" (page 26). If you used SUSE Studio Onsite and have created an image for `USB Stick/Hard Disk Image`, convert it first as described in Procedure 2.2, "Converting Disk Images to `qcow2` Format" (page 26).

Procedure 2.2:   *Converting Disk Images to qcow2 Format*

The `qcow2` format helps to optimize disk space as it consumes disk space only when contents are written on it.

1 Make sure the `virt-utils` package is installed on the machine used for conversion.

2 Download the disk image from SUSE Studio.

3 Convert the image:

```
qemu-img convert -c -f raw -O qcow2 IMAGE_FILE FINAL_IMAGE_FILE
```

Procedure 2.3:   *Uploading Disk Images to SUSE Cloud*

The following procedure shows how to upload a disk image using the `glance` variant that is contained in the package `python-glanceclient`.

1 In a shell, source the OpenStack RC file for the project that you want to upload an image to. For details, refer to Section 2.2, "OpenStack RC File" (page 20).

2 Upload the image with the following command:

```
glance image-create --name="IMAGE_NAME" --is-public=True
--container-format=bare \
  --disk-format=qcow2 < PATH_TO_FINAL_IMAGE_FILE
```

NOTE: Image File Required

The glance client does not support piping images from standard input. Therefore make sure that `PATH_TO_FINAL_IMAGE_FILE` really specifies an image file.

If you need to pipe images, use the former glance client version `/usr/bin/glance.essex`, which is available from the `openstack-glance` package.

Glance does not check any image properties during upload, therefore you need to specify the image's properties as command line options.

Images have both contents and meta-data; the latter are also called properties. The following properties can be attached to an image in SUSE Cloud:

Image Properties

Name (`--name`, optional)
> Specifies a name with which the image will be listed in the SUSE 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.

Architecture (optional)
> The image's architecture.

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`, and `ami`. 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.

VM Mode (optional)
Specify the hypervisor ABI (application binary interface) with the `vm_mode` flag. It can take the values `pv`, `hvm`, or `xen`. Use `vm_mode=xen` for XEN PV image import, or `vm_mode=hvm` for XEN HVM image import. For KVM, the correct mode is selected automatically.

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 Cloud, the image contents cannot be modified (only the image's metadata). To update image contents, you need to delete the current image and upload a modified version of the image.

To modify the metadata of an image, use the `glance image-update` command.

2.4.3 Viewing Images and Image Properties

In the following, find some examples on how to view images or image properties that are available in Glance.

Listing Images

```
glance image-list
```

Lists ID, name, disk format, and container format for all publicly available images in Glance.

Showing Metadata for a Particular Image

```
glance image-show IMAGE_ID
```

Shows metadata of the specified image.

2.4.4 Removing Disk Images from Glance

To delete an image, you need to know the image's ID. If necessary, look it up as described in Section 2.4.3, "Viewing Images and Image Properties" (page 28). Delete the image with the following command:

```
glance image-delete IMAGE_ID
```

If you need to delete all images and all image metadata, use the `glance clear` command.

2.4.5 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 PROJECT_ID
```

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 PROJECT_ID
```

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

2.5 Managing Flavors

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.

A flavor consists of the following parameters:

Flavor Parameters

Flavor ID
  Automatically proposed by SUSE Cloud.

Name
  Name for the new flavor.

VCPUs
  Number of virtual CPUs to use.

Memory MB
  Amount of RAM to use (in megabytes).

Root Disk GB
  Amount of disk space (in gigabytes) to use for the root (/) partition.

Ephemeral Disk GB
  Amount of disk space (in gigabytes) to use for the ephemeral partition. If unspecified, the value is 0 by default.

  Ephemeral disks offer machine local disk storage linked to the lifecycle of a VM instance. When a VM is terminated, all data on the ephemeral disk is lost. Ephemeral disks are not included in any snapshots.

Swap

> Amount of swap space (in megabytes) to use. If unspecified, the value is `0` by default.

Default Flavors

- m1.tiny (1 VCPU/0 GB Disk/512 MB RAM)

- m1.smaller (1 VCPU/0 GB Disk/1024 MB RAM)

- m1.small (1 VCPU/10 GB Disk/2048 MB RAM)

- m1.medium (2 VCPU/10 GB Disk/3072 MB RAM)

- m1.large (4 VCPU/10 GB Disk/8192 MB RAM)

- m1.xlarge (8 VCPU/10 GB Disk/8192 MB RAM)

Flavors can be managed with the `nova flavor-*` commands, provided by the `python-novaclient` package.

Find examples for the key administration tasks below.

Listing Flavors

```
nova flavor-list
```

> Lists all flavors with their ID and name, the amount of memory, the amount of disk space for the root partition and for the ephemeral partition, the swap, and the number of virtual CPUs.

Creating a Flavor

```
nova flavor-create FLAVOR_NAME FLAVOR_ID RAM_IN_MB ROOT_DISK_IN_GB \
  NUMBER_OF_VCPUS
```

> When creating a flavor, you need to specify at least the parameters listed above. For optional parameters, refer to `nova help flavor-create`.

Deleting a Flavor

```
nova flavor-delete FLAVOR_ID
```

Deletes the specified flavor.

2.6 Setting Quotas

To prevent system capacities from being exhausted without notification, cloud administrators can set up quotas. In OpenStack, quotas are currently defined per project.

Quotas contain the following parameters:

Quota Parameters

ID (Name)
  ID for the quota settings. Automatically proposed by SUSE Cloud.

Metadata Items
  Number of metadata items per instance.

Injected Files
  Number of injected files.

Injected File Content Bytes
  Number of bytes per injected file.

VCPUs
  Number of virtual CPUs that can be allocated in total.

Instances
  Total number of instances.

Volumes
  Total number of volumes.

Gigabytes
  Total size of all volumes, measured in gigabytes.

RAM (in MB)
  Total RAM size of all instances, measured in megabytes.

Floating IPs
  Total number of floating IP addresses.

Security Group Rules
: Number of security rules per security group.

Security Groups
: Number of security groups.

Quotas can be managed with the `nova quota-*` commands, provided by the `python-novaclient` package.

Find examples for the key administration tasks below.

Showing Default Quota Values

```
nova quota-defaults PROJECT_ID
```

Lists the default quotas for a tenant. They are hard-coded in OpenStack Nova.

Showing Quota Values for a Project

```
nova quota-show PROJECT_ID
```

Lists the currently set quota values for a project.

Setting Quota Values for a Project

```
nova quota-update --instances 2 PROJECT_ID
```

Sets the quota value for the `instances` parameter to `2`. For a list of further options, refer to `nova help quota-update`.

Using OpenStack APIs

3

The OpenStack project provides comprehensive API documentation, including an API Quick Start, a detailed API reference, plus API guides dealing with individual OpenStack components like Nova, Keystone, Glance, or Swift. In addition to that, developer documentation for python developers and advanced users is available.

For details, refer to http://docs.openstack.org/api/ and http://docs.openstack.org/developer/.