SUSE Cloud

2.0

www.suse.com

September 24, 2013

Deployment Guide

Deployment Guide

List of Authors: Tanja Roth, Frank Sundermeyer

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

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 provides cloud operators with the information needed to deploy and maintain SUSE Cloud administrative units, the Administration Server, and the Control Node, as well as the Compute and Storage Nodes. The Administration Server provides all services needed to manage and deploy all other nodes in the cloud. The Control Node hosts all OpenStack services needed to operate virtual machines deployed on the Compute Nodes in the SUSE Cloud. Each virtual machine (instance) started in the cloud will be hosted on one of the Compute Nodes. Object storage is managed by the Storage Nodes.

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_cloud20`.

1 Available Documentation

The following manuals are available for this product:

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

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

End User Guide (↑*End User Guide*)
Describes how to manage images, networks, instances, 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
- *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.

1 The SUSE Cloud Architecture

SUSE Cloud is a cloud infrastructure solution that can easily be deployed and managed. It offers a cloud management solution that helps organizations to centralize virtual machine deployment. SUSE Cloud 2.0 provides the following features:

- Open source software that is based on the OpenStack Grizzly release.

- Centralized resource tracking providing insight into activities and capacity of the cloud infrastructure for optimized automated deployment of services.

- A self-service portal enabling end users to configure and deploy services as necessary, also offering the ability to track resource consumption (Horizon).

- An image repository from which standardized, pre-configured virtual machines can be published(Glance).

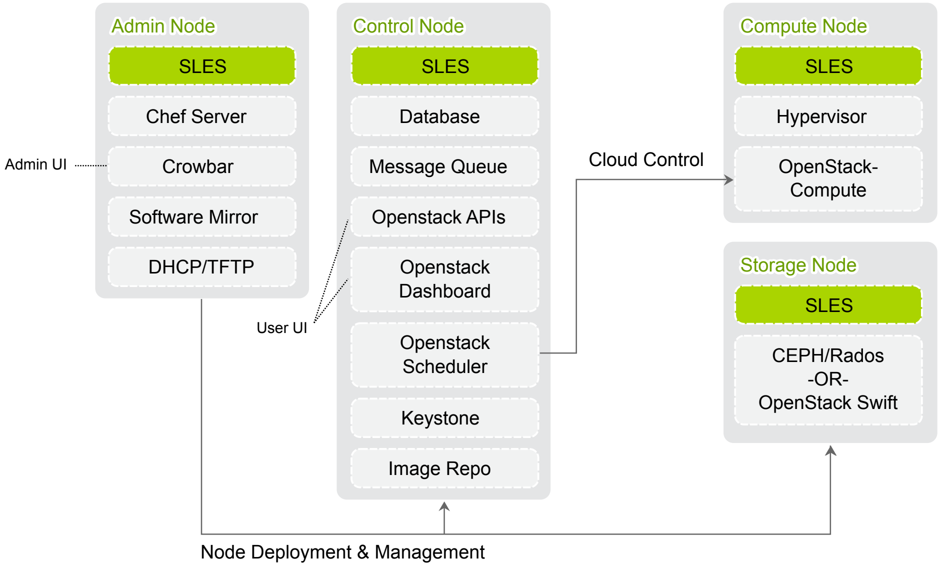
- Automated installation processes via Crowbar utilizing predefined scripts for configuring and deploying Compute and Storage Nodes.

- Multi-tenant, role-based provisioning and access control for multiple departments and users within your organization.

- APIs enabling the integration of third-party software, such as identity management and billing solutions.

SUSE Cloud is based on SUSE Linux Enterprise Server, OpenStack, Crowbar, and Chef. SUSE Linux Enterprise Server is used as the underlying operating system for all cloud infrastructure machines (also called nodes), whereas OpenStack, the cloud management layer, works as the “Cloud Operating System”. Crowbar and Chef are used to automatically deploy and manage the OpenStack nodes from a central Administration Server.

Figure 1.1: *SUSE Cloud Infrastructure*

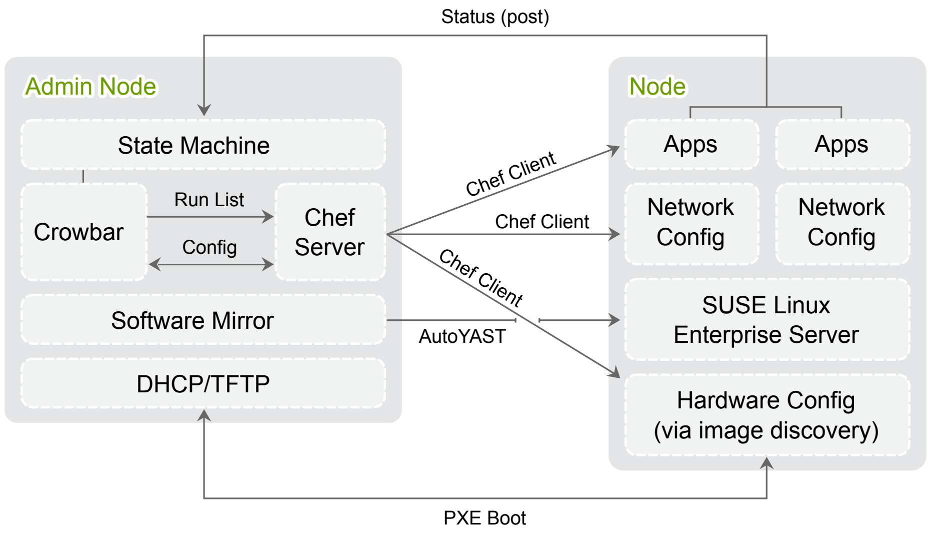


SUSE Cloud is deployed to four different types of machines:

- one Administration Server for node deployment and management
- one or more Control Nodes hosting the cloud management services
- several Compute Nodes on which the instances are started
- several Storage Nodes for block and object storage

1.1 The Administration Server

The Administration Server provides all services needed to manage and deploy all other nodes in the cloud. Most of these services are provided by the Crowbar tool that automates in conjunction with Chef all the required installation and configuration tasks. Among the services provided by the server are DHCP, DNS, NTP, PXE, TFTP.



The Administration Server also hosts the software repositories for SUSE Linux Enterprise Server and SUSE Cloud, since they are needed for node deployment. Optionally (if no other sources for the software repositories are available) it can also host the Subscription Management Tool (SMT), providing up-to-date repositories with updates and patches for all nodes.

1.2 The Control Node(s)

The Control Node(s) hosts all OpenStack services needed to orchestrate virtual machines deployed on the Compute Nodes in the SUSE Cloud. OpenStack on SUSE Cloud uses a PostgreSQL database, which is also hosted on the Control Node. The following OpenStack components and dependencies run on the Control Node(s):

- PostgreSQL database
- Image (Glance) for managing virtual images
- Identity (Keystone), providing authentication and authorization for all OpenStack services

- Networking (Neutron), providing “networking as a service” between interface devices managed by other OpenStack services

- Block Storage (Cinder), providing block storage

- Nova Dashboard (Horizon), providing the Dashboard, which is a user Web interface for the OpenStack services

- Compute (Nova) management (Nova multi-controller) including API and scheduler

- Message broker (RabbitMQ)

- Swift; proxy server plus dispersion tools

- Swift Ring

- Ceph master cluster monitor

Being a central point in the SUSE Cloud architecture that runs a lot of services, a single Control Node can quickly become a performance bottleneck, especially in large SUSE Cloud deployments. It is possible do distribute the services listed above on more than one Control Node, up to a setup where each service runs on its own node.

Deploying Networking (Neutron) on a distinct node is a general recommendation for production clouds. It will separate the network management and most of the network traffic and will also allow to define a distinct network mode for this node (see Appendix B, *The Network Barclamp Template File* (page 105) for details).

Hosting Identity (Keystone) on a distinct node enables you to separate authentication and authorization services from other cloud services for security reasons. Another “good candidate” to be hosted on a separate node is Block Storage (Cinder, particularly the cinder-volume role) when using local disks for storage. Deploying it on one or more separate node enables you to equip the node with storage and network hardware best suiting the service.

NOTE: Moving Services in an Existing Setup

In case you plan to move a service in an already deployed SUSE Cloud from one Control Node to another, it is strongly recommended to shut down or save *all* instances before doing so (and restart them after having successfully re-deployed the services).

1.3 The Compute Nodes

The Compute Nodes are the pool of machines on which the instances are running. These machines need to be equipped with a sufficient number of CPUs and enough RAM to start several instances. They also need to provide sufficient hard disk space, see Section 2.3.2.3, “Compute Nodes” (page 22) for details. The Control Node effectively distributes instances within the pool of Compute Nodes and provides the necessary network resources. The OpenStack service Compute (Nova) runs on the Compute Nodes and provides means for setting up, starting, and stopping virtual machines.

SUSE Cloud supports several hypervisors such as KVM, QEMU or Xen. Each image that can be started with a instance is bound to one hypervisor. Each Compute Node can only run one hypervisor at a time. You can choose which hypervisor to run on which Compute Node when deploying the Nova Barclamp.

1.4 The Storage Nodes

The Storage Nodes are the pool of machines providing storage. SUSE Cloud offers two different types of storage: object and block storage. Object storage is provided by the OpenStack Swift component, while block storage is provided by Cinder. Deploying Swift is optional.

Considerations and Requirements

2

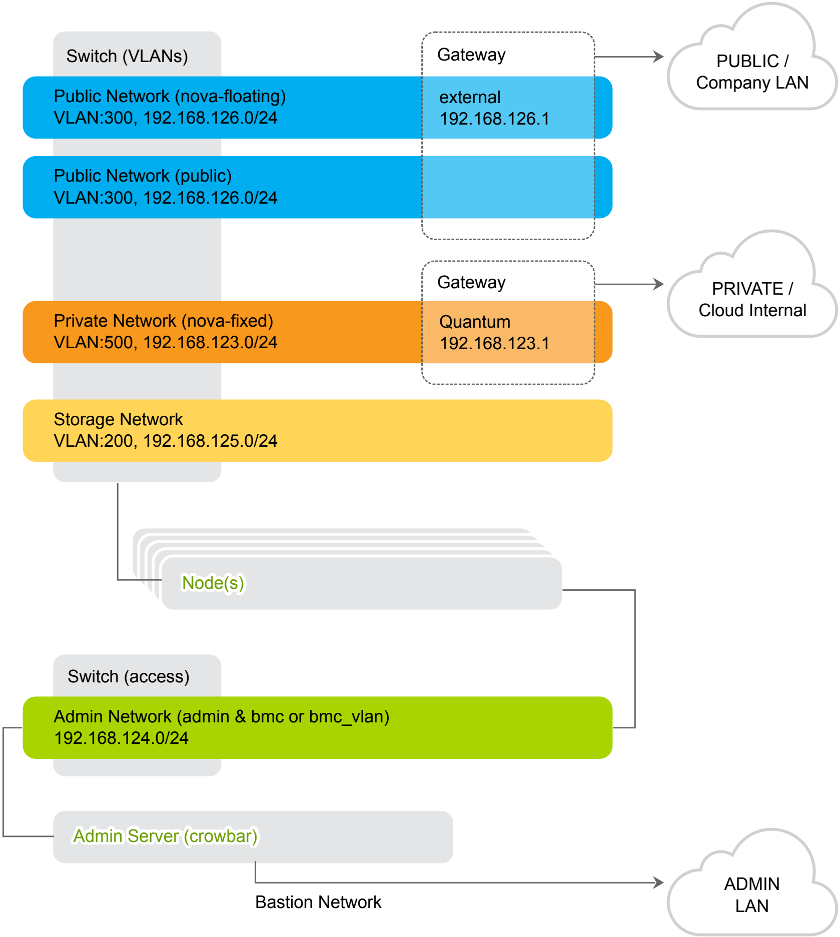
Before deploying SUSE Cloud, there are a few requirements to be met and considerations to be made. Make sure to thoroughly read this chapter—some decisions need to be made *before* deploying SUSE Cloud, since you cannot change them afterwards.

2.1 Network

SUSE Cloud requires a complex network setup consisting of several networks that are configured during installation. These networks are for exclusive cloud usage. In order to access them from an existing network, a router is needed.

The network configuration on the nodes in the SUSE Cloud network is entirely controlled by Crowbar. Any network configuration not done with Crowbar (e.g. with YaST) will automatically be overwritten. Once the cloud is deployed, network settings cannot be changed anymore!

Figure 2.1: *SUSE Cloud Network: Overview*



The following networks are pre-defined when setting up SUSE Cloud. The IP addresses listed are the default addresses and can be changed using the YaST Crowbar module (see Section 3.1.9, “Crowbar Setup” (page 36)). It's also possible to completely customize the network setup. This requires to manually edit the network Barclamp template. See Appendix B, *The Network Barclamp Template File* (page 105) for detailed instructions.

Admin Network (`192.168.124/24`)

A private network to access the Administration Server and all nodes for administration purposes. The default setup lets you also access the BMC (Baseboard Management Controller) data via IPMI (Intelligent Platform Management Interface) from this network. If required, BMC access can be swapped to a separate network.

You have the following options for controlling access to this network:

- do not allow access from the outside and keep the admin network completely separated
- allow access to the Administration Server from a single network (e.g. your company's administration network) via the “bastion network” option configured on an additional network card with a fixed IP address
- allow access from one or more networks via a gateway

NOTE: VLAN support for the Admin Network

As of SUSE Cloud 2.0, a virtual VLAN is not supported for the admin network. If you need VLAN support for the admin network, it must be handed by a “real” switch.

Storage Network (`192.168.125/24`)

Private, SUSE Cloud internal virtual network. This network is used by Ceph and Swift only. It should not be accessed by users.

Private Network (nova-fixed, `192.168.123/24`)

Private, SUSE Cloud internal virtual network. This network is used for inter-instance communication only. The gateway required is also automatically provided by SUSE Cloud.

Public Network (nova-floating, public, `192.168.126/24`)

The only public network provided by SUSE Cloud. You can access the Nova Dashboard as well as instances (provided they have been equipped with a floating IP) on this network. This network can only be accessed via a gateway, which needs to be provided externally. All SUSE Cloud users and administrators need to be able to access the public network.

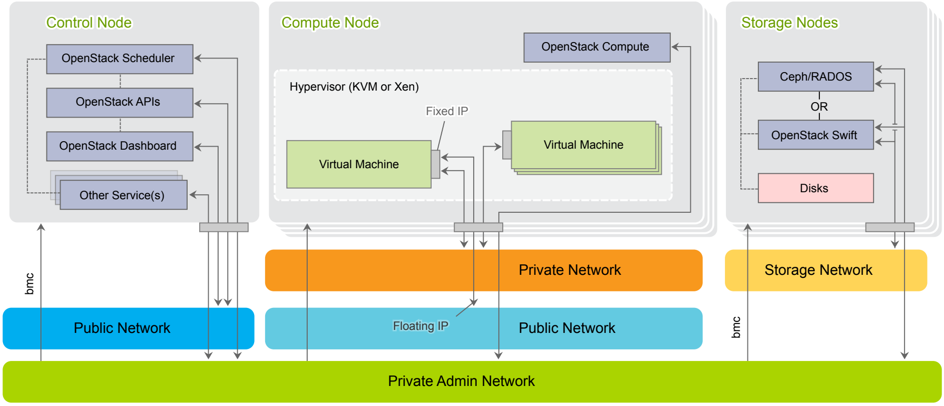
Software Defined Network (os_sdn, `192.168.130/24`)
: Private, SUSE Cloud internal virtual network. This network is used by Neutron only. It should not be accessed by users.

NOTE: No IPv6 support

As of SUSE Cloud 2.0, IPv6 is not supported. This applies to the cloud internal networks as well as to the instances.

The following diagram shows the pre-defined SUSE Cloud network in more detail. It demonstrates how the OpenStack nodes and services use the different networks.

Figure 2.2: *SUSE Cloud Network: Details*



2.1.1 Network Address Allocation

The default networks set up in SUSE Cloud are class C networks with 256 IP addresses each. This limits the maximum number of instances that can be started simultaneously. Addresses within the networks are allocated as outlined in the following table. Use the YaST Crowbar module to customize (see Section 3.1.9, “Crowbar Setup” (page 36)). The `.255` address for each network is always reserved as the broadcast address. This assignment cannot be changed.

NOTE: Limitations of the Default Network Proposal

The default network proposal as described below limits the maximum number of Compute Nodes to 49, the maximum number of floating IP addresses to 61 and the maximum number of instances to 240.

To overcome this limitations you need to reconfigure the network setup by using a class B or class C networks and appropriate network ranges. Do this by either using the YaST Crowbar module as described in Section 3.1.9, “Crowbar Setup” (page 36) or by manually editing the network template file as described in Appendix B, *The Network Barclamp Template File* (page 105).

Table 2.1: *192.168.124.0/24 (Admin/BMC) Network Address Allocation*

Function	Address	Remark
router	`192.168.124.1`	Provided externally.
admin	`192.168.124.10 - 192.168.124.11`	Fixed addresses reserved for the Administration Server.
dhcp	`192.168.124.21 - 192.168.124.80`	Address range reserved for node allocation/installation. Determines the maximum number of parallel allocations/installations.
host	`192.168.124.81 - 192.168.124.160`	Fixed addresses for the OpenStack nodes. Determines the maximum number of OpenStack nodes that can be deployed.
bmc vlan host	`192.168.124.161`	Fixed address for the BMC VLAN.
bmc host	`192.168.124.162 -192.168.124.240`	Fixed addresses for the OpenStack nodes. Determines the maximum number of OpenStack nodes that can be deployed.

Function	Address	Remark
switch	192.168.124.241 -192.168.124.250	

Table 2.2: 192.168.125/24 (Storage) Network Address Allocation

Function	Address	Remark
host	192.168.125.10 - 192.168.125.239	

Table 2.3: 192.168.123/24 (Private Network/nova-fixed) Network Address Allocation

Function	Address	Remark
router	192.168.123.1 - 192.168.123.49	Each Compute Node also acts as a router for “its” instances, getting an address from this range assigned. This effectively limits the maximum number of Compute Nodes that can be deployed with SUSE Cloud to 49.
dhcp	192.168.123.50 - 192.168.123.254	Address range for instances.

Table 2.4: 192.168.126/24 (Public Network nova-floating, public) Network Address Allocation

Function	Address	Remark
public host	192.168.126.2 - 192.168.126.49	Public address range for external SUSE Cloud services such as the Nova Dashboard or the API.
public dhcp	192.168.126.50 - 192.168.126.127	Public address range for instances. These addresses are automatically assigned to the instances to e.g. al-

Function	Address	Remark
		low them to access the internet. Not to be confused with the floating IP addresses that are assigned by users and allow to login to the instance from the outside. Determines the maximum number of instances that can be started concurrently.
floating host	`192.168.126.129 -192.168.126.191`	Floating IP address range. Floating IPs can be manually assigned to a running instance to allow to access the guest from the outside. Determines the maximum number of instances that can concurrently be accessed from the outside. The nova_floating network is set up with a netmask of 255.255.255.192, allowing a maximum number of 61 IP addresses. This range is pre-allocated by default and managed by Neutron.

Table 2.5: *192.168.130/24 (Software Defined Network) Network Address Allocation*

Function	Address	Remark
host	`192.168.130.10 - 192.168.130.254`	

2.1.2 Network Modes

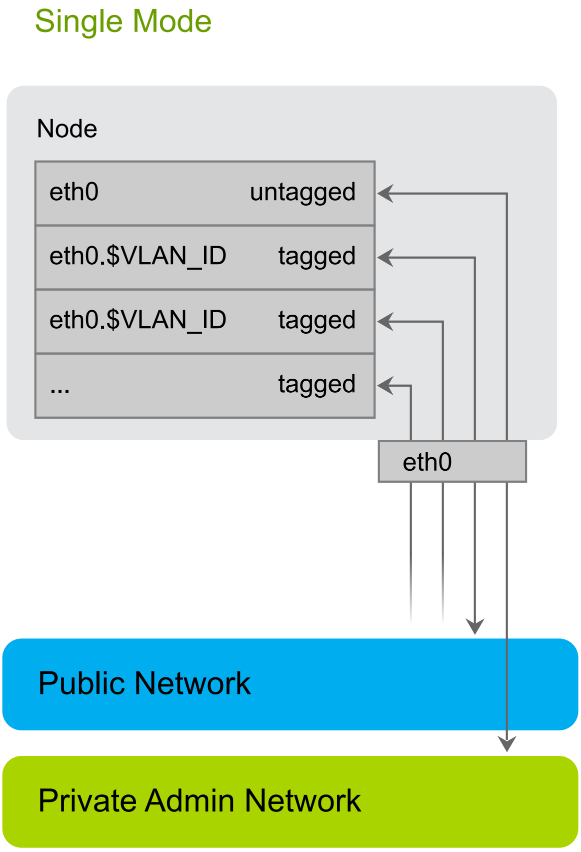
SUSE Cloud supports different network modes: single, dual, and teaming. As of SUSE Cloud 2.0 the networking mode is applied to all nodes as well as the Administration Server. That means that all machines need to meet the hardware requirements for the chosen mode. The network mode can be configured using the YaST Crowbar module

(Section 3.1.9, “Crowbar Setup” (page 36)). The network mode cannot be changed once the cloud is deployed.

Other, more flexible network mode setups can be configured by manually editing the Crowbar network configuration files. See Appendix B, *The Network Barclamp Template File* (page 105) for more information. SUSE can assist you in creating a custom setup within the scope of a Level 3 support contract.

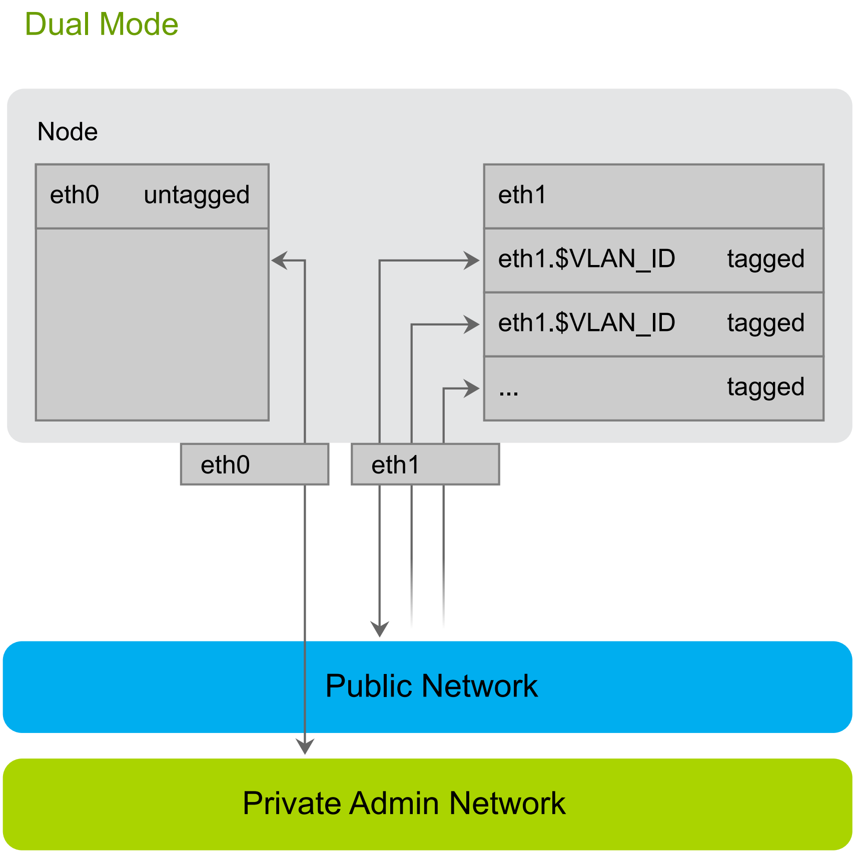
2.1.2.1 Single Network Mode

In single mode you just use one ethernet card for all the traffic:



2.1.2.2 Dual Network Mode

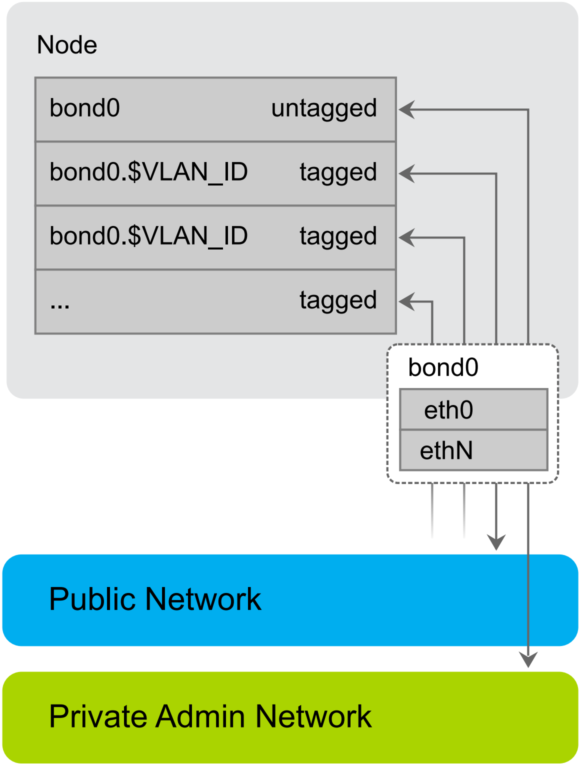
Dual mode needs two ethernet cards (on all nodes but Administration Server) and allows you to completely separate traffic to/from the Admin Network and to/from the public network:



2.1.2.3 Teaming Network Mode

Teaming mode is almost identical to single mode, except for the fact that you combine several ethernet cards to a “bond” in order to increase the performance. Teaming mode needs two or more ethernet cards.

Team Mode



2.1.3 Accessing the Administration Server via a Bastion Network

If you want to enable access to the Administration Server from another network, you can do so by providing an external gateway. This option offers maximum flexibility, but requires additional machines and may be less secure than you require. Therefore SUSE Cloud offers a second option for accessing the Administration Server: the bastion network. You just need a dedicated ethernet card and a static IP address from the external network to set it up.

The bastion network setup enables you to login to the Administration Server via SSH. A direct login to other nodes in the cloud is not possible. However, the Administration Server can server as a “jump host”: To log in to a node, first log in to the Administration Server via SSH. From there, you can “ssh” to other nodes.

2.1.4 DNS and Hostnames

The Administration Server acts as a name server for all nodes in the cloud. If you allow access to the admin network from outside, you may want to add additional name servers to your network setup prior to deploying SUSE Cloud. If additional name servers are found on cloud deployment, the name server on the Administration Server will automatically be configured to forward requests for non-local records to those servers.

The Administration Server needs to be configured to have a fully qualified hostname. This hostname must not be changed after SUSE Cloud has been deployed. The OpenStack nodes will be named after their MAC address by default, but you can provide aliases, which are easier to remember when allocating the nodes. The aliases for the OpenStack nodes can be changed any time. It is useful to have a list of MAC addresses and the intended use of the corresponding host at hand when deploying the OpenStack nodes.

2.2 Product and Update Repositories

In order to deploy SUSE Cloud and to be able to keep a running SUSE Cloud up-to-date, a total of seven software repositories is needed. This includes the static product repositories, which do not change over the product life cycle and the update repositories which constantly change. The following repositories are needed:

SUSE Linux Enterprise Server 11 SP3 Product

> The SUSE Linux Enterprise Server 11 SP3 product repository is a copy of the installation media (DVD1) for SUSE Linux Enterprise Server. As of SUSE Cloud 2.0 it is required to have it available locally on the Administration Server. This repositories requires approximately 3.5 GB of hard disk space.

SUSE Cloud 2.0 Product

> The SUSE Cloud 2.0 product repository is a copy of the installation media for SUSE Cloud. It can either be made available remote via http or locally on the Administration Server. The latter is recommended, since it makes the setup of the

Administration Server easier. This repositories requires approximately 250 MB of hard disk space.

Cloud-PTF
: A repository created automatically on the Administration Server upon the SUSE Cloud add-on product installation. It serves as a repository for “Program Temporary Fixes” (PTF) which are part of the SUSE support program.

SLES11-SP3-Pool and SUSE-Cloud-2.0-Pool
: The SUSE Linux Enterprise Server and SUSE Cloud repository containing all binary RPMs from the installation media, plus pattern information and support status metadata. These repositories are served from Novell Customer Center and need to be kept in sync with it's source. They can be made available remotely via an existing SMT or SUSE Manager server or locally on the Administration Server by installing a local SMT server, by mounting or syncing a remote directory or by regularly copying them.

SLES11-SP3-Updates and SUSE-Cloud-2.0-Updates
: These repositories contain maintenance updates to packages in the corresponding Pool repositories. These repositories are served from Novell Customer Center and need to be kept in sync with it's source. They can be made available remotely via an existing SMT or SUSE Manager server or locally on the Administration Server by installing a local SMT server, by mounting or syncing a remote directory or by regularly copying them.

Since the product repositories (for SUSE Linux Enterprise Server 11 SP3 and SUSE Cloud 2.0), do not change during the life cycle of a product they can be copied to the destination directory from the installation media. The update repositories however, need to be kept in sync with it's source, the Novell Customer Center. SUSE offers two products taking care of synchronizing repositories and making them available within your organization: SUSE Manager (`http://www.suse.com/products/suse-manager/` and Subscription Management Tool (SMT, `http://www.suse.com/solutions/tools/smt.html`.

All repositories need to be served via `http` in order to be available for SUSE Cloud deployment. Repositories that are directly available on the Administration Server are made served by the Apache Web server running on the Administration Server. If your organization already uses SUSE Manager or SMT, you can use the repositories served from theses servers.

Making the repositories locally available on the Administration Server makes setting up the Administration Server more complicated, but has the advantage of a simple network setup within SUSE Cloud. It also allows you to seal off the SUSE Cloud network from other networks in your organization. Using a remote server as a source for the repositories has the advantage of using existing resources and services. It also makes setting up the Administration Server much easier, but requires a custom network setup for SUSE Cloud.

Using a remote SMT Server

> If you already run an SMT server within your organization, you can use it within SUSE Cloud. When using a remote SMT server, update repositories are served directly from the SMT server. Each node is configured with this repositories upon it's initial setup. The SMT server needs to be accessible from the Administration Server and all nodes in SUSE Cloud (via one or more gateways). Resolving the server's hostname also needs to work.

Using a SUSE Manager Server

> Each client that is managed by SUSE Manager needs to register with the SUSE Manager server. Therefore the SUSE Manager support can only be installed after the nodes have been deployed. In order to also be able to use repositories provided by SUSE Manager during node deplyoment, SUSE Linux Enterprise Server 11 SP3 must be set up for autoinstallation on the SUSE Manager server.
>
> The server needs to be accessible from the Administration Server and all nodes in SUSE Cloud (via one or more gateways). Resolving the server's hostname also needs to work.

Installing a Subscription Management Tool (SMT) Server on the Administration Server

> The SMT server, a free add-on product for SUSE Linux Enterprise Server, regularly synchronizes repository data from Novell Customer Center with your local host. Installing the SMT server on the Administration Server is recommended if you do not have access to update repositories from elsewhere within your organization. This option requires the Administration Server to be able to access the Internet. Subscription Management Tool 11 SP3 is available for free from `http://www.suse.com/solutions/tools/smt.html`.

“Sneakernet”

> If you choose to completely seal off your admin network from all other networks, you need to manually update the repositories from removable media. For this pur-

pose copy the repositories from an existing SMT or SUSE Manager server to the removable media.

Utilizing Existing Repositories

If you can access existing repositories from within your company network from the Administration Server, you can either mount or sync these repositories from an existing SMT or SUSE Manager server to the required locations on the Administration Server.

2.3 Persistent Storage

When talking about “persistent storage” on SUSE Cloud, there are two completely different aspects to discuss: the block and object storage services SUSE Cloud offers on the one hand and the hardware related storage aspects on the different node types.

NOTE: Persistent vs. Ephemeral Storage

Block and object storage are persistent storage models where files or images are stored until they are explicitly deleted. SUSE Cloud also offers ephemeral storage for images attached to instances. These ephemeral images only exist during the life of a instance and are deleted once the guest is terminated. See Section 2.3.2.3, “Compute Nodes” (page 22)for more information.

2.3.1 Cloud Storage Services

As mentioned above, SUSE Cloud offers two different types of services for persistent storage: object and block storage. Object storage lets you upload and download files (similar to an FTP server), whereas a block storage provides mountable devices (similar to a hard-disk partition). Furthermore SUSE Cloud provides a repository to store the virtual disk images used to start instances.

Object Storage with Swift

The object OpenStack storage service is called Swift. The storage component of Swift (swift-storage) needs to be deployed on dedicated nodes where no other cloud services run. In order to be able to store the objects redundantly, it is required to deploy at least two Swift nodes. SUSE Cloud is configured to always use all unused disks on a node for storage.

Swift can optionally be used by Glance, the service that manages the images used to boot the instances. Offering object storage with Swift is optional.

Block Storage

Block storage on SUSE Cloud is provided by Cinder. Cinder can use a variety of storage backends, among them network storage solutions like NetApp or EMC. It is also possible to use local disks for block storage.

Alternatively, Cinder can use Ceph RBD as a backend. Ceph offers data security and speed by storing the devices redundantly on different servers. Ceph needs to be deployed on dedicated nodes where no other cloud services run. In order to be able to store the objects redundantly, it is required to deploy at least two Ceph nodes.

IMPORTANT: Ceph not Supported

Ceph is included in SUSE Cloud 2.0 as a technology preview. Customers can use this in test environments but it is not recommended for production. Supported block storage is provided by Cinder.

The Glance Image Repository

Glance provides a catalog and repository for virtual disk images used to start the instances. Glance is installed on a Control Node. It either uses Swift or a directory on the Control Node to store the images. The image directory can either be a local directory or an NFS share.

2.3.2 Storage Hardware Requirements

Apart from sufficient disk space to install the SUSE Linux Enterprise Server operating system, each node in SUSE Cloud has to store additional data. Requirements and recommendations for the various node types are listed below.

IMPORTANT: Choose a Hard Disk for the Operating System Installation

The operating system will always be installed on the *first* hard disk, the one that is recognized as `/dev/sda`. This is the disk that is listed *first* in the BIOS, the one from which the machine will boot. If you have nodes with a certain hard disk you want the operating system to be installed on, make sure it will be recognized as the first disk.

2.3.2.1 Administration Server

If you store the update repositories directly on the Administration Server (see Section 2.2, "Product and Update Repositories" (page 17) for details), it is recommended to mount `/srv` to a separate partition or volume with a minimum of 30 GB space.

2.3.2.2 Control Nodes

Depending on how the services are set up, Glance and Cinder may require additional disk space on the Control Node on which they are running. Both services may be configured to use a local image file for storage. For performance and scalability reasons this is only recommended for test setups. Make sure there is sufficient free disk space available if you use a local file for storage.

Cinder may be configured to use local disks for storage (configuration option `raw`). If you choose this setup, it is recommended to deploy the *cinder-volume* role to one or more dedicated Control Nodes equipped with several disks providing sufficient storage space. It may also be necessary to equip this node with two or more bonded network cards (requiring a special setup for this node, refer to Appendix B, *The Network Barclamp Template File* (page 105) for details), since it will generate heavy network traffic.

2.3.2.3 Compute Nodes

Unless an instance is started via "Boot from Volume", it is started with at least one disk—a copy of the image from which it has been started. Depending on the flavor you start, the instance may also have a second, so-called "ephemeral" disk. The size of the root disk depends on the image itself, while ephemeral disks are always created as sparse image files that grow (up to a defined size) when being "filled". By default ephemeral disks have a size of 10 GB.

Both disks, root images and ephemeral disk, are directly bound to the instance and are deleted when the instance is terminated. Therefore these disks are bound to the Compute Node on which the instance has been started. The disks are created under `/var/lib/nova` on the Compute Node. Your Compute Nodes should be equipped with enough disk space to store the root images and ephemeral disks.

NOTE: Ephemeral Disks vs. Block Storage

Do not confuse ephemeral disks with persistent block storage. In addition to an ephemeral disk, which is automatically provided with most instance flavors, you can optionally add a persistent storage device provided by Cinder. Ephemeral disks are deleted when the instance terminates, while persistent storage devices can be reused in another instance.

The maximum disk space required on a compute node depends on the available flavors. A flavor specifies the number of CPUs, as well as RAM and disk size of an instance. Several flavors ranging from *tiny* (1 CPU, 2512 MB RAM, no ephemeral disk) to *xlarge* (8 CPUs, 8 GB RAM, 10 GB ephemeral disk) are available by default. Adding custom flavors as well as editing and deleting existing flavors is also supported.

To calculate the minimum disk space needed on a compute node, you need to determine the highest "disk space to RAM" ratio from your flavors. Example:

Flavor small: 2 GB RAM, 100 GB ephemeral disk => 50 GB disk /1 GB RAM
Flavor large: 8 GB RAM, 200 GB ephemeral disk => 25 GB disk /1 GB RAM

So, 50 GB disk /1 GB RAM is the ratio that matters. If you multiply that value by the amount of RAM in GB available on your compute node, you have the minimum disk space required by ephemeral disks. Pad that value with sufficient space for the root disks plus a buffer that enables you to create flavors with a higher disk space to RAM ratio in the future.

WARNING: Overcommitting Disk Space

The scheduler that decides in which node an instance is started does not check for available disk space. If there is no disk space left on a compute node, this will not only cause data loss on the instances, but the compute node itself will also stop operating. Therefore you must make sure all compute nodes are equipped with enough hard disk space!

2.3.2.4 Storage Nodes

The block-storage service Ceph RBD and the object storage service Swift need to be deployed onto dedicated nodes—it is not possible to mix these services. Each storage service requires at least two machines (more are recommended) to be able to store data redundantly.

Each Ceph/Swift Storage Node needs at least two hard disks. The first one (`/dev/sda`) will be used for the operating system installation, while the others can be used for storage purposes. It's recommended to equip the storage nodes with as much disks as possible.

Using RAID on Swift storage nodes is not supported. Swift takes care of redundancy and replication on its own. Using RAID with Swift would also result in a huge performance penalty.

IMPORTANT: Ceph not Supported

Ceph is included in SUSE Cloud 2.0 as a technology preview. Customers can use this in test environments but it is not recommended for production. Supported block storage is provided by Cinder.

2.4 SSL Encryption

Whenever non-public data travels over a network it needs to be encrypted. Encryption protects the integrity and confidentiality of data. Therefore you should enable SSL support when deploying SUSE Cloud to production (it is not enabled by default). The following services (and their APIs if available) can make use of SSL:

- Neutron
- Keystone
- Glance
- Cinder
- Nova
- VNC
- Nova Dashboard

Using SSL requires an SSL certificate either for each node on which the services that uses encryption run (services sharing a certificate) or, alternatively, a dedicated certificate for each service. A single certificate for the Control Node is the minimum requirement, where all services listed above are installed on the Control Node and are sharing the certificate.

Certificates must be signed by a trusted authority. Refer to `http://www.suse.com/documentation/sles11/book_sle_admin/data/sec_apache2_ssl.html` for instructions how to create and sign them.

IMPORTANT: Host Names

Each SSL certificate is issued for a certain host name and, optional, for alternative hostnames (via the `AlternativeName` option). Each node publicly available node in SUSE Cloud has got two hostnames—an internal and a public one. The SSL certificate needs to be issued for both names.

The internal name has the following scheme:

```
dMAC ADRESS.FQDN
```

`MAC ADRESS` is the MAC address of the interface used to PXE boot the machine with lowercase letters and colons replaced with dashes, for example `52-54-00-8e-ce-e3`. *`FQDN`* is the fully qualified domain name. An example name looks like this:

```
d52-54-00-8e-ce-e3.example.com
```

Unless you have entered a custom *Public Name* for a client (see Section 4.2, “Node Installation” (page 53) for details), the public name is the same as the internal name prefixed by `public.`:

```
public.d52-54-00-8e-ce-e3.example.com
```

To look up the node names open the Crowbar Web interface and click on a node name in the *Node Dashboard*. The names are listed as *Full Name* and *Public Name*.

2.5 Hardware Requirements

Precise hardware requirements can only be listed for the Administration Server and the OpenStack Control Node. The requirements of the OpenStack Compute and Storage Nodes depends on the number of concurrent instances and their virtual hardware equipment.

The minimum number of machines required for a SUSE Cloud setup featuring all services is seven: one Administration Server, one Control Node, one Compute Node, and four Storage Nodes. In addition to that, a gateway providing access to the public network is required.

IMPORTANT: Physical Machines and Architecture

All SUSE Cloud nodes need to be physical machines. Although the Administration Server and the Control Node can be virtualized in test environments, this is not supported for production systems.

SUSE Cloud currently only runs on `x86_64` hardware.

2.5.1 Administration Server

- Architecture: x86_64
- RAM: at least 2 GB, 4 GB recommended
- Hard disk: at least 40 GB. It is recommended to put `/srv` on a separate partition with at least 30 GB space, unless you mount the update repositories from another server (see Section 2.2, “Product and Update Repositories” (page 17) for details).
- Number of network cards: 1 for single and dual mode, 2 or more for team mode. Additional networks such as the bastion network and/or a separate BMC network each need an additional network card. See Section 2.1, “Network” (page 7) for details.

2.5.2 Control Node

- Architecture: x86_64
- RAM: at least 1 GB, 2 GB recommended
- Number of network cards: 1 for single mode, 2 for dual mode, 2 or more for team mode. See Section 2.1, “Network” (page 7) for details.
- Hard disk: See Section 2.3.2.2, “Control Nodes” (page 22).

2.5.3 Compute Node

The Compute Nodes need to be equipped with a sufficient amount of RAM and CPUs, matching the numbers required by the maximum number of instances running concurrently. An instance started in SUSE Cloud cannot share resources from several physical nodes, but rather uses the resources of the node on which it was started. So if you offer a flavor (see Flavor (page 125) for a definition) with 8 CPUs and 12 GB RAM, at least one of your nodes should be able to provide these resources.

See Section 2.3.2.3, “Compute Nodes” (page 22) for storage requirements.

2.5.4 Storage Node

The Storage Nodes are sufficiently equipped with a single CPU and 1 or 2 GB of RAM. See Section 2.3.2.4, “Storage Nodes” (page 23) for storage requirements.

2.6 Software Requirements

The following software requirements need to be met in order to install SUSE Cloud:

- SUSE Linux Enterprise Server 11 SP3 installation media (ISO image, included in the SUSE Cloud Administration Server subscription)
- Access to the SUSE Linux Enterprise Server 11 SP3 Update repositories
- SUSE Cloud 2.0 installation media (ISO image).

- Access to the SUSE Cloud 2.0 Update repositories

- A SUSE/Novell account (for product registration and SMT setup). If you do not already have one, go to `http://www.suse.com/login` to create it.

- Optional: Subscription Management Tool 11 SP3 installation media. A free download is available on `http://www.novell.com/linux/smt/`. See Section 2.2, “Product and Update Repositories” (page 17).

2.7 Summary: Considerations and Requirements

As outlined above, there are some important considerations to be made before deploying SUSE Cloud. The following briefly summarizes what was discussed in detail in this chapter. Keep in mind that as of SUSE Cloud 2.0 it is not possible to change some aspects such as the network setup once SUSE Cloud is deployed!

Network

- If you do not want to stick with the default networks and addresses, define custom networks and addresses. You need four different networks, at least three of them VLANs (nova-fixed/floating, public and storage). If you need to separate the admin and the BMC network, a fifth network is required. See Section 2.1, “Network” (page 7) for details.

- The SUSE Cloud networks are completely isolated, therefore it is not required to use public IP addresses for them. A class C network as used in this documentation provides a sufficient number of addresses. However, you may alternatively choose addresses from a class B or A network.

- Determine how to allocate addresses from your network. Make sure not to allocate IP addresses twice. See Section 2.1.1, “Network Address Allocation” (page 10) for the default allocation scheme.

- Define which network mode to use. Keep in mind that all machines within the cloud (including the Administration Server) will be set up with the chosen mode and therefore need to meet the hardware requirements. See Section 2.1.2, “Network Modes” (page 13) for details.

- Define how to access the admin and BMC network(s): no access from the outside (no action is required), via an external gateway (gateway needs to be provided), or via bastion network. See Section 2.1.3, “Accessing the Administration Server via a Bastion Network” (page 16) for details.

- Provide a gateway to access the public network (public, nova-floating).

- Make sure the admin server's hostname is correctly configured (`hostname -f` needs to return a fully qualified hostname).

- Prepare a list of MAC addresses and the intended use of the corresponding host for all OpenStack nodes.

Update Repositories

- Depending on your network setup you have different options on how to provide up-to-date update repositories for SUSE Linux Enterprise Server and SUSE Cloud for SUSE Cloud deployment: using an existing SMT or SUSE Manager server, installing SMT on the Administration Server, syncing data with an existing repository, mounting remote repositories or using a “Sneakernet”. Choose the option that best matches your needs.

Storage

- Decide whether you want to deploy the object storage service Swift. If so, you need to deploy at least two nodes with sufficient disk space exclusively dedicated to Swift.

- Decide which backend to use with Cinder. If using the *raw* backend (local disks) it is strongly recommended to use a separate node equipped with several hard disks for deploying `cinder-volume`. If using Ceph, you need to deploy at least two nodes with sufficient disk space exclusively dedicated to it.

IMPORTANT: Ceph not Supported

Ceph is included in SUSE Cloud 2.0 as a technology preview. Customers can use this in test environments but it is not recommended for production. Supported block storage is provided by Cinder.

- Make sure all Compute Nodes are equipped with sufficient hard disk space.

SSL Encryption

- Decide whether to use different SSL certificates for the services and the API or whether to use a single certificate.
- Get one or more SSL certificates certified by a trusted third party source.

Hardware and Software Requirements

- Make sure the hardware requirements for the different node types are met.
- Make sure to have all required software at hand.

Installing and Configuring the Administration Server

3

Deploying and installing SUSE Cloud is a multi-step process, starting by deploying a basic SUSE Linux Enterprise Server installation and the SUSE Cloud add-on product to the Administration Server. Now the product and update repositories need to be set up and the SUSE Cloud network needs to be configured. Next the Administration Server setup will be finished. Once the Administration Server is ready, you can start deploying and configuring the OpenStack nodes. The complete node deployment is done automatically via Crowbar and Chef from the Administration Server. All you need to do is to PXE boot the nodes and to deploy the OpenStack services to them.

Procedure 3.1: *High Level Overview of the SUSE Cloud Installation*

1. Install SUSE Linux Enterprise Server 11 SP3 on the Administration Server with the add-on products Subscription Management Tool (optional) and SUSE Cloud. See below.

2. Once the Administration Server is set up, PXE boot all nodes onto which the OpenStack components should be deployed and allocate them in the Crowbar Web interface to start the automatic SUSE Linux Enterprise Server installation. See Chapter 4, *Installing the OpenStack Nodes* (page 51).

3. Configure and deploy the OpenStack services via the Crowbar Web interface or command line tools. See Chapter 5, *Deploying the OpenStack Services* (page 69).

4. When all OpenStack services are up and running, SUSE Cloud is ready. The cloud admin can now upload images to enable users to start deploying instances. See *User Guide for Administrators* (↑*User Guide for Administrators*).

In this chapter you will learn how to install and set up the Administration Server from bare metal. As a result, the Administration Server will be ready to deploy OpenStack nodes and services. It will run on SUSE Linux Enterprise Server 11 SP3 and will include the add-on products SUSE Cloud and SMT. Installing the Administration Server involves the following basic steps:

3.1 Operating System Installation

Start the installation by booting from the SUSE Linux Enterprise Server 11 SP3 installation medium.

NOTE: Differences from the Default Installation Process

For an overview of a default SUSE Linux Enterprise Server installation, refer to the SUSE Linux Enterprise Server *Installation Quick Start*. Detailed installation instructions are available in the SUSE Linux Enterprise Server *Deployment Guide*. Both documents are available at `http://www.suse.com/documentation/sles11/`.

The following sections will only cover the differences from the default installation process.

3.1.1 Add-On Product Selection

SUSE Cloud is an add-on product to SUSE Linux Enterprise Server. Installing it during the the SUSE Linux Enterprise Server installation is the easiest and recommended way to set up the Administration Server. If you are planning to set up an SMT server on the Administration Server as well, also install it in this step (SMT is an add-on product, too). Make sure to be able to access the installation media (DVD or ISO image) for all add-on products. Alternatively, install the add-on products after the SUSE Linux Enterprise Server installation.

NOTE: SMT Server Installation

When not using an existing SMT or SUSE Manager server, installing an SMT server on the Administration Server is the easiest way to ensure that SUSE Cloud is provided with up-to-date update repositories for SUSE Linux Enterprise

Server and SUSE Cloud. See Section 2.2, “Product and Update Repositories” (page 17) for alternatives and background information.

On the *Installation Mode* screen, click *Include Add-On products from Separate Media*. Proceed with *Next* to the Add-On product installation dialog. If you have direct access to the installation media (for example, via DVD or USB stick), skip the network installation dialog. Otherwise configure the network as described in Section 3.1.7, “Basic Network Configuration” (page 35). Add SUSE Cloud and SMT (optional) as add-on products and proceed with the installation. Consult the SUSE Linux Enterprise Server *Deployment Guide* at http://www.suse.com/documentation/sles11/book_sle_deployment/data/sec_i_yast2_inst_mode.html for detailed instructions.

3.1.2 Partitioning

Currently, Crowbar requires `/opt` to be writable. Apart from that, SUSE Cloud has no special requirements in regards of partitioning. However, it is recommended to create a separate partition or volume for `/srv`. /srv will host all update and product repositories for SUSE Linux Enterprise Server and SUSE Cloud. A size of at least 25 GB is required. Help on using the partitioning tool is available at http://www.suse.com/documentation/sles11/book_sle_deployment/data/sec_yast2_i_y2_part_expert.html.

3.1.3 Software Selection

Installing a minimal base system is sufficient to set up the Administration Server. The following patterns are the minimum requirement:

- *Base System*
- *Minimal System (Appliances)*
- *Subscription Management Tool* (optional)
- *SUSE Cloud Admin Node*
- *Web and LAMP Server* (only needed when installing SMT)

3.1.4 Product Registration

Although you can also register your products at any time after the installation, it is recommended to register SUSE Linux Enterprise Server and SUSE Cloud now, because it will give you immediate access to the update channels. If you have installed the SMT Add-On product, you *must* register your SUSE Linux Enterprise Server version at the Novell Customer Center *now*, otherwise you will not be able to configure the SMT server. You have received registration keys with the SUSE Cloud Administration Server subscription. See `http://www.suse.com/documentation/sles11/book_sle_deployment/data/sec_i_yast2_conf.html` for details on the Novell Customer Center registration.

NOTE: SUSE Login Required

In order to register a product, you need to have a SUSE/Novell login. If you do not have such a login, create it at `http://www.suse.com/login`.

3.1.5 Online Update

A successful product registration will add update repositories for SUSE Linux Enterprise Server and all add-on products. After having successfully registered you will be asked to perform an online update, which will update the system and the add-on products. It is strongly recommended to perform the update at this point in time. If you choose to skip the update now, you must perform it later, before running the Cloud installation script.

3.1.6 CA Setup

Skip this step if you have not installed SMT add-on product. In case you have installed SMT you need to provide a certification authority (CA). If you already have a CA certificate in your organization, import it. Otherwise generate all certificates in the Administration Server itself by accepting the YaST proposal. See `http://www.suse.com/documentation/sles11/book_security/data/cha_security_yast_ca.html` for more information.

If SMT is not installed, click on the *CA Management* link and choose to not set up a CA.

3.1.7 Basic Network Configuration

Only the first interface (`eth0`) on the Administration Server needs to be configured during the installation. Other interfaces will be automatically configured by the cloud installation script.

`eth0` needs to be given a fixed IP address from the admin network—when sticking with the default network addresses this would be `192.168.124.10`. The address you need to enter for the *Default Gateway* depends on whether you have provided an external gateway for the admin network (use the address of that gateway) or not (use *`xxx.xxx.xxx`*.1, e.g. `192.168.124.1`). Using a custom IP address or more than one network interfaces requires to adjust the Crowbar configuration in a later step as described in Section 3.1.9, “Crowbar Setup” (page 36).

If you allow to access the admin network from another network (via gateway or bastion network), you can also add one or more name servers. The Administration Server's name server will automatically be configured by the cloud installation script to forward requests for non-local records to those server(s).

You also need to assign a hostname and a fully qualified domain name (FQDN) such as *`admin.cloud.example.com`* to `eth0` (tab *Hostname/DNS* in the *Network settings Dialog*).

Last, the firewall needs to be disabled for all interfaces.

IMPORTANT: Administration Server Domain Name and Hostname

Setting up the SUSE Cloud will also install a DNS server for all nodes in the cloud. The domainname you specify for the Administration Server will be used for the DNS zone. It is recommended to use a sub-domain such as *`cloud.example.com`*.

The hostname and the FQDN need to be resolvable with `hostname -f`. Double-check whether `/etc/hosts` contains an appropriate entry for the Administration Server. It should look like the following:

```
192.168.124.10 admin.cloud.example.com admin
```

It is *not* possible to change the Administration Server hostname or the FQDN once the cloud installation script has been run.

3.1.8 SMT Configuration

Skip this step if you have not installed the SMT add-on product. In case you have installed it, you will be asked to configure it. Configuring the SMT server requires you to have your mirroring credentials (username and password) and your registration e-mail address at hand. To access them, log in to the Novell Customer Center at `http://www.novell.com/center/`. Get the mirror credentials by selecting *My Products* > *Mirror Credentials* in the left navigation. Obtain your registration e-mail address from *My Profile* > *Login Profile*.

Enter this data at the *SMT Configuration Wizard Step 1/2* into the fields *User*, *Password*, and *NCC E-mail Used for Registration*. Accept the pre-filled defaults for the other input fields. Make sure to *Test* the credentials.

In step two of the SMT configuration you need to enter a database password and specify an e-mail address for getting reports. Refer to `http://www.suse.com/documentation/smt11/` for the complete *SMT for SUSE Linux Enterprise 11 Guide*.

3.1.9 Crowbar Setup

This YaST module enables you to configure all networks within the cloud and set the network mode for all networks. Furthermore you can also change the username and password for the Crowbar Web interface with which you can manage the OpenStack nodes.

Start YaST and choose *Miscellaneous* > *Crowbar* to start the YaST Crowbar module. The *Administration Settings* tab lets you change the password for the Crowbar Web interface.

On the *Network Mode* tab you can choose between *single*, *dual*, and *team* mode. When choosing *team*, you also need to set the *Bonding Policy*. See Section 2.1.2, “Network Modes” (page 13) for details on SUSE Cloud and network modes. In-depth information about the *Bonding Policy* (also known as bonding modes) is available at `https://www.kernel.org/doc/Documentation/networking/bonding.txt` in section 2, *Bonding Driver Options*, under *mode*.

If you do not want to use the default IP addresses and the default address allocation, change these settings on the *Networks* tab. See Section 2.1, “Network” (page 7) for details on the cloud network. You can also change the Bridge and VLAN allocation on the *Networks* tab. Only change them if you really know what you require, sticking with the defaults is recommended.

If you want to separate the admin and the BMC network, you must change the settings for the networks *bmc* and *bmc_vlan*. The *bmc_vlan* is used to generate a VLAN tagged interface on the Administration Server that can access the *bmc* network. The *bmc_vlan* needs to be in the same ranges as *bmc*, and *bmc* has to have *VLAN* enabled.

Table 3.1: *Separate BMC Network Example Configuration*

	bmc	**bmc_vlan**
Subnet	`192.168.128.0`	
Netmask	`255.255.255.0`	
Router	`192.168.128.1`	
Broadcast	`192.168.128.255`	
Host Range	`192.168.128.10` - `192.168.128.100`	`192.168.128.101` - `192.168.128.101`
VLAN	yes	
VLAN ID	100	
Bridge	no	

IMPORTANT: No Network Changes after Having Run the Cloud Installation Script

As of SUSE Cloud 2.0 it is not possible to change the network setup after having run the cloud installation script. Allowing such changes is planned for future releases of SUSE Cloud.

NOTE: Setting up a Bastion Network

As of SUSE Cloud 2.0 it is not possible to set up a bastion network with YaST Crowbar. It needs to be configured manually—see Section 3.2.4.1, “Setting Up a Bastion Network” (page 47).

Other, more flexible network mode setups can be configured by manually editing the Crowbar network configuration files. See Appendix B, *The Network Barclamp Template File* (page 105) for more information. SUSE can assist you in creating a custom setup within the scope of a Level 3 support contract.

3.2 Post-Installation Configuration

After the installation has finished, you need to set up product and update repositories and, optionally, configure the bastion network. Once the Administration Server host is fully configured, start the cloud installation script.

3.2.1 Setting up the SMT Repositories

If you are using an SMT server (locally or from another network), you need to configure it to mirror the update repositories needed for SUSE Cloud. Skip this step if you are not using an SMT server (but make sure the repositories listed at Section 2.2, “Product and Update Repositories” (page 17) are available on the Administration Server).

The SMT server mirrors the update channels from the Novell Customer Center. In order to be able to access the SUSE Linux Enterprise Server and 2.0 repositories, make sure to have registered both products in Novell Customer Center. Run the following commands as user `root` on the SMT server:

```
for REPO in SLES11-SP3-{Pool,Updates} SUSE-Cloud-2.0-{Pool,Updates}; do
```

```
  smt-repos $REPO sle-11-x86_64 -e
done
smt-mirror -L /var/log/smt/smt-mirror.log
```

The `smt-repos` command will add the list of repositories to the SMT server. The `smt-mirror` command will mirror them and download approximately several GB of patches. This process may last up to several hours. A log file is written to `/var/log/smt/smt-mirror.log`. The following table lists all repositories and their file system location:

Table 3.2: *SMT Repositories for SUSE Cloud*

Repository	Directory
SLES11-SP3-Pool	`/srv/www/htdocs/repo/$RCE/SLES11-SP3-Pool/sle-11-x86_64`
SLES11-SP3-Updates	`/srv/www/htdocs/repo/$RCE/SLES11-SP3-Updates/sle-11-x86_64`
SUSE-Cloud-2.0-Pool	`/srv/www/htdocs/repo/$RCE/SUSE-Cloud-2.0-Pool/sle-11-x86_64`
SUSE-Cloud-2.0-Updates	`/srv/www/htdocs/repo/$RCE/SUSE-Cloud-2.0-Updates/sle-11-x86_64`

3.2.2 Setting Up Repositories for Node Deployment

In order to deploy the OpenStack nodes and to provide update repositories for them, product and update repositories for SUSE Linux Enterprise Server and SUSE Cloud must be configured. See Section 2.2, “Product and Update Repositories” (page 17) for background information.

3.2.2.1 Product Repositories

The files in the product repositories for SUSE Linux Enterprise Server and SUSE Cloud do not change, therefore they do not need to be synced with a remote source. It is sufficient (and recommended) to copy the data once, either from a remote host or directly from the installation media. Alternatively you may mount the product repository from a remote server via NFS. Please note that the data *must* be directly available from the local directories listed below. It is not possible to use links.

While the SUSE Linux Enterprise Server product repository must be made available locally, the SUSE Cloud repository may also be served via http from a remote host. However, copying the data (approximately 250 MB) to the Administration Server as described here, is recommended, since it does not require a custom network configuration for the Administration Server.

If copying, it is recommended to use rsync. If the installation data is located on a removable device, make sure to mount it first (for example, after inserting the DVD in the Administration Server and waiting for the device to become ready):

```
mkdir -p /srv/tftpboot/suse-11.3/install/
mount /dev/dvd /mnt
rsync -avP /mnt/ /srv/tftpboot/suse-11.3/install/
umount /mnt
```

If the SLES installation data is provided by a remote machine, log in to that machine and push the data to the Administration Server (which has the IP address 192.168.124.10 in the following example):

```
rsync -avPz /data/sles11sp3/ 192.168.124.10:/srv/tftpboot/suse-11.3/install/
```

Also make the contents of the SUSE Cloud product repository available at /srv/tftpboot/repos/Cloud/ the same way:

```
mkdir -p /srv/tftpboot/repos/Cloud/
mount /dev/dvd /mnt
rsync -avP /mnt/ /srv/tftpboot/repos/Cloud/
umount /mnt
```

If the Cloud installation data is provided by a remote machine, log in to that machine and push the data to the Administration Server (which has the IP address 192.168.124.10 in the following example):

```
rsync -avPz /data/Cloud/ 192.168.124.10:/srv/tftpboot/repos/Cloud/
```

The following product repositories are now available for SUSE Cloud deployment:

Table 3.3: *Local Product Repositories for SUSE Cloud*

Repository	Directory
SLES11 SP3 Product	`/srv/tftpboot/suse-11.3/install`
Cloud 2.0 Product	`/srv/tftpboot/repos/Cloud`

3.2.2.2 Update Repositories

Update repositories are already used when deploying the nodes that will build SUSE Cloud in order to ensure they are initially equipped with the latest software versions available. If you are using a remote SMT or SUSE Manager server, the update repositories from the remote server are used directly. In this case the repository URLs need to be added to the file `/etc/crowbar/provisioner.json`. If using repositories locally available, the Administration Server's itself acts as the repository provider for all nodes. This requires to make them available in `/srv/tftpboot/repos`.

SMT Server installed on a Remote Host

In order to use repositories from a remote SMT server to deploy SUSE Cloud you first need to make sure the products SUSE Linux Enterprise Server 11 SP3 and Cloud 2.0 are registered and the corresponding channels are mirrored in SMT. Now you need to add the repository URLs to `/etc/crowbar/provisioner.json` by running the following commands:

```
for REPO in SLES11-SP3-{Pool,Updates} SUSE-Cloud-2.0-{Pool,Updates}; do
  ATTR_REPO=$(echo $REPO | sed "s:\.:\\\.:g")
  /opt/dell/bin/json-edit \
    -a "attributes.provisioner.suse.autoyast.repos.$ATTR_REPO.url" \
    -v "http://smt.example.com/repo/\$RCE/$REPO/sle-11-x86_64/" \
    /etc/crowbar/provisioner.json
done
```

Note that you need to replace *`smt.example.com`* with the fully qualified hostname of your SMT server.

NOTE: Debugging / Adding Additional Repositories

In case you have specified a wrong path or URL to a repository, the failure to add the repository is silently ignored when deploying the nodes. This can cause a broken node deployment which is hard to debug. If adding repositories to `/etc/crowbar/provisioner.json` you can optionally also set the `ask_on_error` flag for each repository. If set to `true` (it defaults to `false`), node deployment stops with an error message if a wrong repository path is detected. Set it to `true` for a specific repository by running the following command:

```
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.repository.ask_on_error"
 \
  -v "true" \
  /etc/crowbar/provisioner.json
```

If you need to provide additional custom repositories for node deployment, you can also add them to `/etc/crowbar/provisioner.json`. Run the following command:

```
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.my_custom_repo.url" \
  -v "URL" \
  /etc/crowbar/provisioner.json
```

SUSE Manager Server

In order to use repositories from SUSE Manager to deploy SUSE Cloud you first need to make sure the products SUSE Linux Enterprise Server 11 SP3 and Cloud 2.0 are registered and the corresponding channels are mirrored in SUSE Manager. By default SUSE Manager does not expose repositories for direct access. In order to be able to access them via `https`, you need to create a *Distribution* for Autoinstallation for the SUSE Linux Enterprise Server 11 SP3 (x86_64) Product on SUSE Manager. Instructions can be found at `https://www.suse.com/documentation/suse_manager/book_susemanager_ref/data/book_susemanager_ref.html` under the heading *Creating Distribution for Autoinstallation*. During the distribution setup you need to provide a *Label* for the distribution. This label will be part of the URL under which the repositories are available. It is recommended to choose a name consisting of characters that do not need to be URL-encoded, for example `sles11-sp3-x86_64`.

Creating a distribution for SUSE Linux Enterprise Server 11 SP3 not only make the installation data and Update repositories for this product available, but also for all registered add-on products, including SUSE Cloud 2.0.

The repositories are available under the following URLs. `manager.example.com` needs to be replaced by the fully qualified host name of you SUSE Manager server and `sles11-sp3-x86_64` needs to be replaced by the distribution label you specified when setting up the distribution for autoinstallation. Note that the URLs are not browseable.

SLES11-SP3-Update
: http://manager.example.com/ks/dist/child/sles11-sp3-updates-x86_64/sles11-sp3-x86_64/

SUSE-Cloud-2.0-Pool
: http://manager.example.com/ks/dist/child/suse-cloud-2.0-pool-x86_64/sles11-sp3-x86_64/

SUSE-Cloud-2.0-Updates
: http://manager.example.com/ks/dist/child/suse-cloud-2.0-updates-x86_64/sles11-sp3-x86_64/

To make the repositories available for node deployment, you need to add the repository URLs to `/etc/crowbar/provisioner.json` by running the commands listed below. These repositories will be used during the node installation. Once the nodes have been installed you need to properly register the nodes with the SUSE Manager in a second step described at Section 4.3.2, “Configuring Node Updates with the SUSE Manager Barclamp” (page 61).

```
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.SLES11-SP3-Updates.url" \
  -v
"http://manager.example.com/ks/dist/child/sles11-sp3-updates-x86_64/sles11-sp3-x86_64/"
 \
  /etc/crowbar/provisioner.json
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.SUSE-Cloud-2\.0-Pool.url" \
  -v
"http://manager.example.com/ks/dist/child/suse-cloud-2.0-pool-x86_64/sles11-sp3-x86_64/"
 \
  /etc/crowbar/provisioner.json
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.SUSE-Cloud-2\.0-Updates.url" \
  -v
"http://manager.example.com/ks/dist/child/suse-cloud-2.0-updates-x86_64/sles11-sp3-x86_64/"
 \
  /etc/crowbar/provisioner.json
```

Note that you need to replace *`manager.example.com`* with the fully qualified hostname of your SUSE Manager server. `sles11-sp3-x86_64` needs to be replaced by the distribution label you specified when setting up the distribution for autoinstallation.

NOTE: Debugging / Adding Additional Repositories

In case you have specified a wrong path or URL to a repository, the failure to add the repository is silently ignored when deploying the nodes. This can cause a broken node deployment which is hard to debug. If adding repositories to `/etc/crowbar/provisioner.json` you can optionally also set the `ask_on_error` flag for each repository. If set to `true` (it defaults to `false`), node deployment stops with an error message if a wrong repository path is detected. Set it to `true` for a specific repository by running the following command:

```
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.repository.ask_on_error"
 \
  -v "true" \
  /etc/crowbar/provisioner.json
```

If you need to provide additional custom repositories for node deployment, you can also add them to `/etc/crowbar/provisioner.json`. Run the following command:

```
/opt/dell/bin/json-edit \
  -a "attributes.provisioner.suse.autoyast.repos.my_custom_repo.url" \
  -v "URL" \
  /etc/crowbar/provisioner.json
```

SMT Server installed on the Administration Server

Link the repositories mirrored by SMT to `/srv/tftpboot`:

```
for REPO in SLES11-SP3-{Pool,Updates} SUSE-Cloud-2.0-{Pool,Updates}; do
  ln -s /srv/www/htdocs/repo/\$RCE/$REPO/sle-11-x86_64 /srv/tftpboot/repos/$REPO
done
```

Update Repositories Hosted a Remote Host

If the update repositories are hosted on a remote host that can be accessed from the Administration Server you can either mount them, for example via `NFS`, or regularly `rsync` them.

To `NFS`-mount the repositories from a remote host, either use the YaST *NFS Client* module or edit `/etc/fstab`. The local mount point should be `/srv/tftpboot/repos/`*<REPOSITORY_NAME>*.

To `rsync` the repositories from a remote host, create a daily cron job running the following command on the Administration Server. This command will *pull* the files from a host named host.example.com:

```
for REPO in SLES11-SP3-{Pool,Updates} SUSE-Cloud-2.0-{Pool,Updates}; do
  rsync -avPz host.example.com:/srv/www/htdocs/repo/\\\$RCE/$REPO/sle-11-x86_64/ \
  /srv/tftpboot/repos/$REPO/
done
```

Alternatively you may set up the cron job on the remote host and *push* the file to the Administration Server (which has the IP address `192.168.124.10` in the following example):

```
for REPO in SLES11-SP3-{Pool,Updates} SUSE-Cloud-2.0-{Pool,Updates}; do
  rsync -avPz /srv/www/htdocs/repo/\\\$RCE/$REPO/sle-11-x86_64/ \
  192.168.124.10:/srv/tftpboot/repos/$REPO/
done
```

NOTE: Mind the Trailing Slash

The `rsync` command must be used with trailing slashes in the directory names as shown above. Otherwise rsync would copy the repositories into the wrong directory.

Sneakernet

If your admin network is isolated from other networks, you need to manually sync the update repositories from removable media. To do so you can either use `rsync` (see above for an example) or `cp -axu`. If copying from a SMT server, see Section 3.2.1, “Setting up the SMT Repositories” (page 38) for a list of directories to copy.

The following update repositories are now available for SUSE Cloud deployment:

Table 3.4: *Local Update Repositories for SUSE Cloud*

Repository	Directory
SLES11-SP3-Pool	`/srv/tftpboot/repos/SLES11-SP3-Pool`
SLES11-SP3-Updates	`/srv/tftpboot/repos/SLES11-SP3-Updates`

Repository	Directory
SUSE-Cloud-2.0-Pool	`/srv/tftpboot/repos/SUSE-Cloud-2.0-Pool`
SUSE-Cloud-2.0-Updates	`/srv/tftpboot/repos/SUSE-Cloud-2.0-Updates`

3.2.3 Software Repository Sources on the Administration Server

Update repositories are not only required to deploy SUSE Cloud. The Administration Server itself also needs to be kept up-to-date and therefore needs to have a proper repository setup. In case you have registered SUSE Linux Enterprise Server and SUSE Cloud during the installation process, the Administration Server already has all required update repositories.

These repositories are served directly from Novell Customer Center. In order to avoid downloading the same patches twice or in case you would like to cut off the Administration Server from the internet, it makes sense to change this setup in a way that the repositories set up for SUSE Cloud deployment are also used on the Administration Server. To do so, you need to disable or delete all services. In a second step all SUSE Linux Enterprise Server and SUSE Cloud repositories need to be edited in order to point to the alternative sources. Editing the repository setup can either be done with Zypper or YaST. Note that changing the repository setup on the Administration Server is optional.

3.2.4 Custom Network Configuration

In case you need to adjust the pre-defined network setup of SUSE Cloud beyond the scope of changing IP address assignments (as described in Section 3.1.9, “Crowbar Setup” (page 36)), you need to manually modify the network Barclamp template. Refer to Appendix B, *The Network Barclamp Template File* (page 105) for details.

3.2.4.1 Setting Up a Bastion Network

As outlined in Section 2.1, “Network” (page 7), one way to access the Administration Server from a defined external network is via a Bastion network and a second network card (as opposed to providing an external gateway).

To set up the Bastion network, you need to have a static IP address for the Administration Server from the external network. You need to adjust the network template file `/etc/crowbar/network.json`. The example configuration used below assumes that the external network from which to access the admin network has the following addresses. You need to adjust them according to your needs.

Table 3.5: *Example Addresses for a Bastion Network*

Subnet	`10.10.1.0`
Netmask	`255.255.255.0`
Broadcast	`10.10.1.255`
Gateway	`10.10.1.1`
Static Administration Server address	`10.10.1.125`

To add a bastion network, you need to manually adjust `/etc/crowbar/network.json`. Once the bastion network configuration has been added to `network.json`, it can be adjusted using the YaST Crowbar module.

1 Run the following series of commands as user `root` in a shell to add the bastion network definition:

```
BASE="attributes.network.networks.bastion"
FILE="/etc/crowbar/network.json"
/opt/dell/bin/json-edit -a "${BASE}.add_bridge" -v "false" $FILE
/opt/dell/bin/json-edit -a "${BASE}.vlan" -v "50" $FILE
/opt/dell/bin/json-edit -a "${BASE}.router" -v "10.10.1.1" $FILE
/opt/dell/bin/json-edit -a "${BASE}.broadcast" -v "10.10.1.255" $FILE
/opt/dell/bin/json-edit -a "${BASE}.netmask" -v "255.255.255.0" $FILE
/opt/dell/bin/json-edit -a "${BASE}.use_vlan" -v "false" $FILE
/opt/dell/bin/json-edit -a "${BASE}.conduit" -v "bastion1" $FILE
/opt/dell/bin/json-edit -a "${BASE}.subnet" -v "10.10.1.0" $FILE
/opt/dell/bin/json-edit -a "${BASE}.router_pref" -v "5" $FILE
/opt/dell/bin/json-edit -a "${BASE}.ranges.admin.start" -v "10.10.1.125"
```

```
$FILE
/opt/dell/bin/json-edit -a "${BASE}.ranges.admin.end" -v "10.10.1.125" $FILE
```

2 Next you need to add the bastion network interface to the conduit_map section of the config file. That section has subsections for the three network modes (single, dual, team, see Section 2.1.2, "Network Modes" (page 13) for details). The interface needs to be added to the network mode you plan to use:

2a Open `/etc/crowbar/network.json` in an editor of your choice and search for the string that matches your network mode:

```
single/.*
dual/.*
team/.*
```

The search will match the following context (having search for `single/.*`):

```
"pattern" : "single/.*/.*",
"conduit_list" : {
```

2b Add the following lines directly after the ones from above:

```
"bastion1" : {
   "if_list" : [
      "1g2"
   ]
},
```

`1g2` is the Crowbar identifier for "the second 1 Gigabit network interface". If you plan to use a different interface, or use bonding in team mode, please refer to Section B.4, "Network Conduits" (page 109) for more details.

3 Save the file.

3.2.5 Running the Cloud Installation Script

Before running the cloud installation script to finish the configuration of the Administration Server make sure to double-check the following items.

Final Check Points

- Make sure the network configuration is correct. Run *YaST* > *Crowbar* to review/change the config. See Section 3.1.9, “Crowbar Setup” (page 36) for further instructions.

- Make sure `hostname -f` returns a fully qualified hostname. See Section 3.1.7, “Basic Network Configuration” (page 35) for further instructions.

- Make sure all update and product repositories are available locally. See Section 3.2.2, “Setting Up Repositories for Node Deployment” (page 39) for further instructions.

- Make sure the operating system and SUSE Cloud are up-to-date and have the latest patches installed. Run `zypper patch` to install them.

Now everything is in place to finally configure the Administration Server. This is done by running the script `install-suse-cloud`. This command will install and configure Chef, and use it to complete the installation of Crowbar and all required Barclamps. It will take several minutes to complete. If you are *not* using SUSE Manager to provide update repositories, run the following command:

```
screen install-suse-cloud
```

In case you are using SUSE Manager (as described in “SUSE Manager Server” (page 42), you need to run the following command:

```
screen env REPOS_SKIP_CHECKS+=" SLES11-SP3-Pool" install-suse-cloud
```

IMPORTANT: Use a Terminal Multiplexer to run the Cloud Installation Script

Run the installation script `install-suse-cloud` inside of a terminal multiplexer like GNU Screen (provided by the `screen` package).

During the run of this script the network will be reconfigured. This may result in interrupting the script when being run from a network connection (like SSH). Using `screen` will continue running the script in a session to which you can reconnect via `screen -r` if you lose the connection.

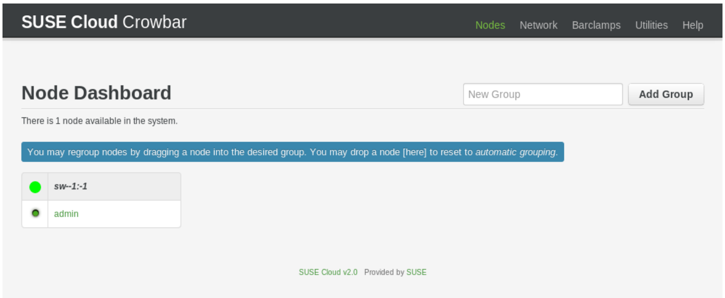
`install-suse-cloud` will produce a lot of output that gets written to a log file located at `/var/log/crowbar/install.log`. Check this log file in case something goes wrong. You can run `install-suse-cloud` multiple times as long as you have not started to deploy the OpenStack services. It is also possible to run `install-suse-cloud` in verbose mode with the `-v` switch. It will show the same output that goes to the log file on STDOUT, too.

If the script has successfully finished, you will see a message telling you how to log in to the Crowbar Web interface.

WARNING: No Network Changes After Having Run the Cloud Installation Script

Once you have run the cloud installation script, you cannot change the network setup anymore. If you did, you would have to completely set up the Administration Server again.

Figure 3.1: *Crowbar Web Interface: Initial State*



Node Dashboard

There is 1 node available in the system.

You may regroup nodes by dragging a node into the desired group. You may drop a node [here] to reset to *automatic grouping*.

● *sw--1:-1*

● admin

SUSE Cloud v2.0 Provided by SUSE

Installing the OpenStack Nodes

4

The OpenStack nodes represent the actual cloud infrastructure. Node installation and service deployment is done automatically from the Administration Server. Before deploying the OpenStack services, you need to install SUSE Linux Enterprise Server on every node. In order to do so, each node needs to be PXE booted using the `tftp` server from the Administration Server. Afterwards you can allocate the nodes and trigger the operating system installation. There are three different types of nodes:

Control Node(s): One or more central management node(s) interacting with all other nodes.
Compute Nodes: The nodes on which the instances are started.
Storage Nodes: Nodes providing object or block storage.

4.1 Preparations

Meaningful Node names

Make a note of the MAC address and the purpose of each node (for example, controller, storage Ceph, storage Swift, compute). This will make deploying the OpenStack services a lot easier and less error-prone, since it enables you to assign meaningful names (aliases) to the nodes, which are otherwise listed with the MAC address by default.

BIOS Boot Settings

Make sure PXE-booting (booting from the network) is enabled and configured as the *primary* boot-option for each node. The nodes will boot twice from the network during the allocation and installation phase.

Custom Node Configuration

All nodes are installed using AutoYaST with the same configuration located at /opt/dell/chef/cookbooks/provisioner/templates/default/autoyast.xml.erb. If this configuration does not match your needs (for example if you need special third party drivers) you need to make adjustments to this file. An AutoYaST manual can be found at http://www.suse.com/documentation/sles11/book_autoyast/data/book_autoyast.html. Having changed the AutoYaST config file, you need to re-upload it to Chef, using the following command:

```
knife cookbook upload -o /opt/dell/chef/cookbooks/ provisioner
```

Direct root Login

By default, the root account on the nodes has no password assigned, so a direct root login is not possible. Logging in on the nodes as root is only possible via SSH public keys (for example, from the Administration Server).

If you want to allow direct root login, you can set a password via the Crowbar Provisioner Barclamp before deploying the nodes. That password will be used for the root account on all OpenStack nodes. Using this method after the nodes are deployed is not possible. In that case you would have to log in to each node via ssh from the Administration Server and change the password manually with passwd.

Setting a root Password for the OpenStack Nodes

1. Create an md5-hashed root-password, for example by using openssl passwd -1.

2. Open a browser and point it to the Crowbar Web interface available at port 3000 of the Administration Server, for example http://192.168.124.10:3000/. Log in as user crowbar. The password defaults to crowbar, if you have not changed it during the installation.

3. Open the Barclamp menu by clicking *Barclamps* > *Crowbar*. Click the *Provisioner* Barclamp entry and *Edit* the *Default* proposal.

4. Click *Raw* in the *Attributes* section to edit the configuration file.

5. Add the following line to the end of the file before the last closing curly bracket:

   ```
   , "root_password_hash": "HASHED_PASSWORD"
   ```

replacing "*HASHED_PASSWORD*" with the password you generated in the first step.

6. Click *Apply*.

Preparing a Windows Netboot Environment

In case you plan to deploy Compute Nodes running either Microsoft Hyper-V Server or Windows Server 2012, you need to prepare a Windows Netboot Environment. Refer to Appendix C, *Setting up a Netboot Environment for Microsoft* Windows* (page 117) for details.

4.2 Node Installation

To install a node, you need to PXE boot it first. It will be booted with an image that enables the Administration Server to discover the node and make it available for installation. Once you have allocated the node, it will PXE boot again and the automatic installation will start.

1 PXE-boot all nodes you want to deploy. The nodes will boot into the “SLEShammer” image, which performs the initial hardware discovery.

IMPORTANT: Limit the Number of Concurrent PXE boots

PXE Booting a large number nodes a the the same time will cause heavy load on the TFPT server, because all nodes will request the boot image at the same time. It's recommended to boot the nodes time-delayed.

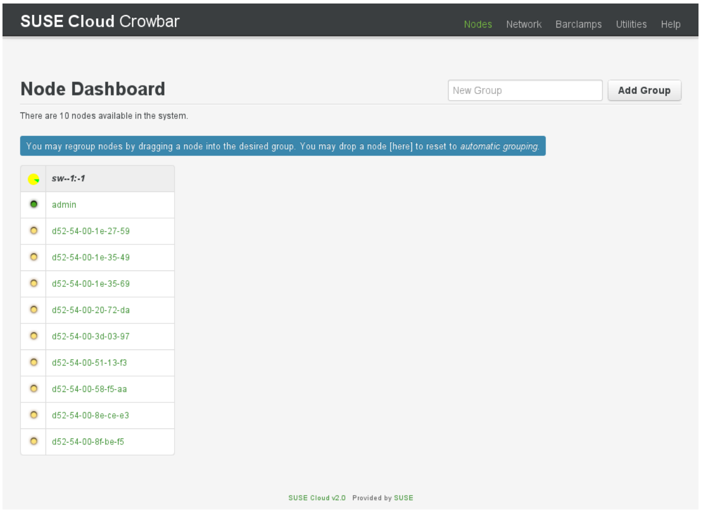
2 Open a browser and point it to the Crowbar Web interface available at port `3000` of the Administration Server, for example `http://192.168.124.10:3000/`. Log in as user `crowbar`. The password defaults to `crowbar`, if you have not changed it.

Click *Nodes* > *Dashboard* to open the *Node Dashboard*.

3 Each node that has successfully booted will be listed as being in state `Discovered`, indicated by a yellow bullet. The nodes will be listed with their MAC address as a name. Wait until all nodes are listed as being `Discovered` before proceeding. In

case a node does not report as being `Discovered`, it may need to be rebooted manually.

Figure 4.1: *Discovered Nodes*



Node Dashboard

There are 10 nodes available in the system.

You may regroup nodes by dragging a node into the desired group. You may drop a node [here] to reset to *automatic grouping*.

	sw--1:-1
●	admin
●	d52-54-00-1e-27-59
●	d52-54-00-1e-35-49
●	d52-54-00-1e-35-69
●	d52-54-00-20-72-da
●	d52-54-00-3d-03-97
●	d52-54-00-51-13-f3
●	d52-54-00-58-f5-aa
●	d52-54-00-8e-ce-e3
●	d52-54-00-8f-be-f5

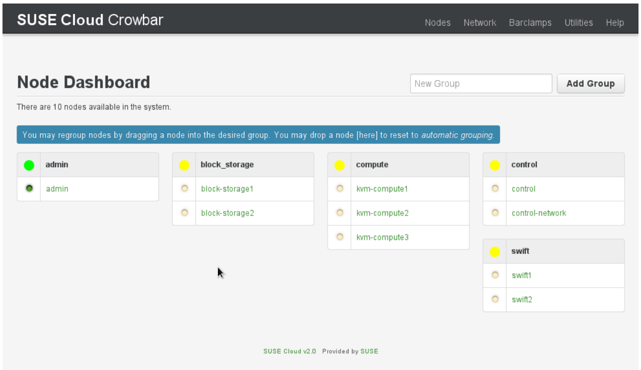
SUSE Cloud v2.0 Provided by SUSE

4 Although this step is optional, it is recommended to properly group your nodes at this stage, since it lets you clearly arrange all nodes. Grouping the nodes by role would be one option, for example control, compute, object storage (Swift), and block storage (Ceph).

4a Enter the name of a new group into the *New Group* input field and click *Add Group*.

4b Drag and drop a node onto the title of the newly created group. Repeat this step for each node you would like to put into the group.

Figure 4.2: *Grouping Nodes*



Node Dashboard

There are 10 nodes available in the system.

You may regroup nodes by dragging a node into the desired group. You may drop a node [here] to reset to *automatic grouping*.

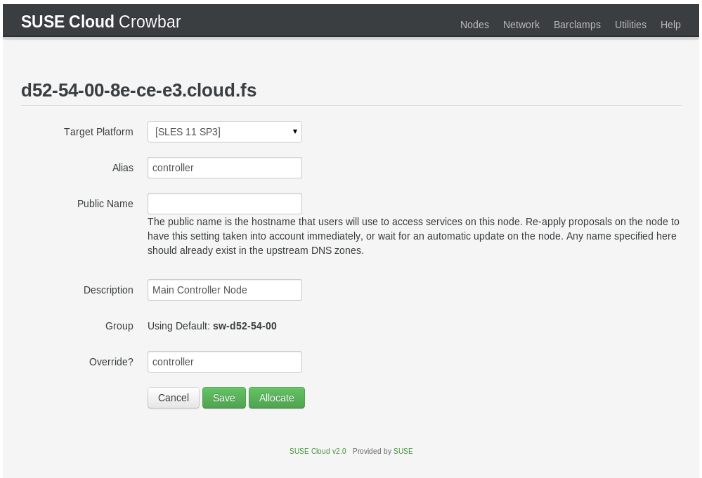
● admin	● block_storage	● compute	● control
● admin	○ block-storage1	○ kvm-compute1	○ control
	○ block-storage2	○ kvm-compute2	○ control-network
		○ kvm-compute3	

● swift
○ swift1
○ swift2

SUSE Cloud v2.0 Provided by SUSE

5 To allocate the nodes click on *Nodes* > *Bulk Edit*. If you prefer to allocate the nodes one-by-one, click a node's name followed by a click on *Edit* instead.

Figure 4.3: *Editing a Single Node*



d52-54-00-8e-ce-e3.cloud.fs

Target Platform	[SLES 11 SP3]
Alias	controller
Public Name	

The public name is the hostname that users will use to access services on this node. Re-apply proposals on the node to have this setting taken into account immediately, or wait for an automatic update on the node. Any name specified here should already exist in the upstream DNS zones.

Description	Main Controller Node
Group	Using Default: **sw-d52-54-00**
Override?	controller

SUSE Cloud v2.0 Provided by SUSE

IMPORTANT: Limit the Number of Concurrent Node Deployments

Deploying a large number nodes in bulk mode will cause heavy load on the Administration Server server. The subsequent concurrent Chef client runs triggered by the nodes will require a lot of RAM on the Administration Server.

Therefore it is recommended to limit the number of concurrent “Allocations” in bulk mode. The maximum number depends on the amount of RAM on the Administration Server—limiting concurrent deployments to five up to ten is recommended.

6 Provide a meaningful *Alias*, *Public Name* and a *Description* for each node and check the *Allocate* box. The entries for *BIOS* and *RAID* are currently not used. Normally *Target Platform* needs to be set to *suse-11.3*. If you plan to support Hyper-V in your cloud, you need to prepare the Windows netboot environment as described in Appendix C, *Setting up a Netboot Environment for Microsoft* Windows* (page 117). Once that is done, set the *Target Platform* of the Compute Nodes that should run Windows to either *Windows Server* or *HyperV Server*. When specifying *Windows Server* you also need to add a valid *License Key*.

TIP: Alias Names

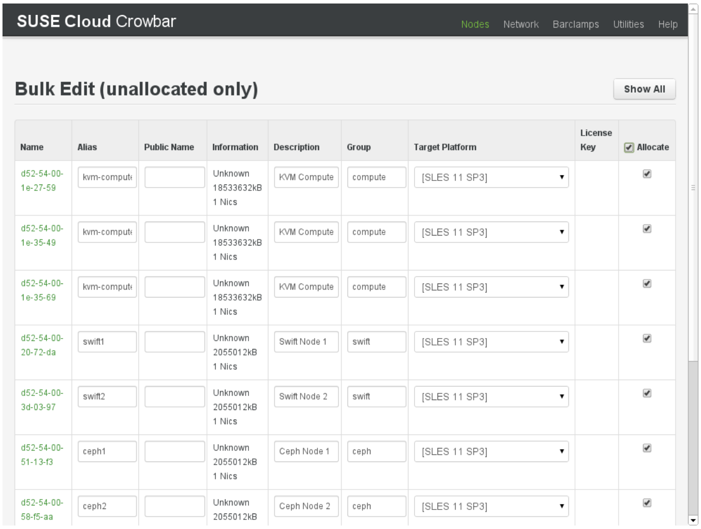
Providing an alias name will change the default node names (MAC address) to the name you provided, making it easier to identify the node. Furthermore, this alias will also be used as a DNS `CNAME` for the node in the admin network. As a result, you will be able to access the node via this alias when, for example, logging in via SSH.

TIP: Public Names

A node's *Alias Name* is resolved by the DNS server installed on the Administration Server and therefore only available within the cloud network. The Nova Dashboard or some APIs (`keystone-server`, `glance-server`, `cinder-controller`, `quantum-server`, `nova-multi-controller`, and `swift-proxy`) can be accessed from outside the SUSE Cloud network. In order to be able to access them by name, these names need to be resolved by a name server placed outside of the SUSE Cloud network. If you have created DNS entries for nodes, specify the name in the *Public Name* field.

The *Public Name* is never used within the SUSE Cloud network. However, if you create an SSL certificate for a node that has a public name, this name must be added as an `AlternativeName` to the certificate (see Section 2.4, “SSL Encryption” (page 24) for more information)..

Figure 4.4: *Bulk Editing Nodes*



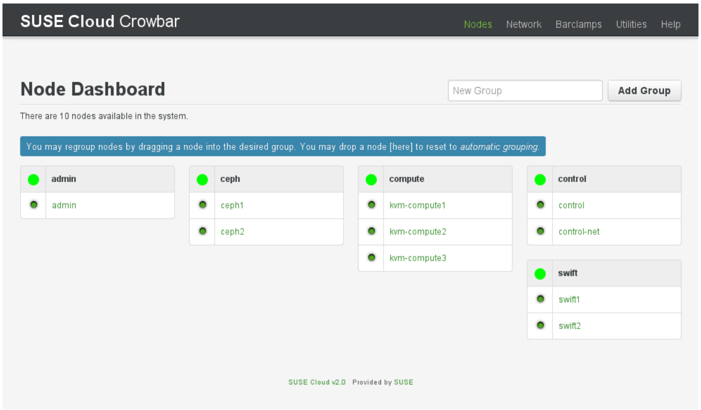
Bulk Edit (unallocated only)

Name	Alias	Public Name	Information	Description	Group	Target Platform	License Key	☑ Allocate
d52-54-00-1e-27-59	kvm-comput		Unknown 18533632kB 1 Nics	KVM Compute	compute	[SLES 11 SP3] ▾		☑
d52-54-00-1e-35-49	kvm-comput		Unknown 18533632kB 1 Nics	KVM Compute	compute	[SLES 11 SP3] ▾		☑
d52-54-00-1e-35-69	kvm-comput		Unknown 18533632kB 1 Nics	KVM Compute	compute	[SLES 11 SP3] ▾		☑
d52-54-00-20-72-da	swift1		Unknown 2055012kB 1 Nics	Swift Node 1	swift	[SLES 11 SP3] ▾		☑
d52-54-00-3d-03-97	swift2		Unknown 2055012kB 1 Nics	Swift Node 2	swift	[SLES 11 SP3] ▾		☑
d52-54-00-51-13-f3	ceph1		Unknown 2055012kB 1 Nics	Ceph Node 1	ceph	[SLES 11 SP3] ▾		☑
d52-54-00-58-f5-aa	ceph2		Unknown 2055012kB	Ceph Node 2	ceph	[SLES 11 SP3] ▾		☑

7 Once you have filled in the data for all nodes, click *Save*. The nodes will reboot and commence the AutoYaST-based SUSE Linux Enterprise Server installation via a second PXE boot. Click *Nodes* > *Dashboard* to return to the *Node Dashboard*.

8 Nodes that are being installed are listed with the status `Installing` (yellow/green bullet). Once the installation of a node has finished, it is listed as being `Ready`, indicated by a green bullet. Wait until all nodes are listed as being `Ready` before proceeding.

Figure 4.5: All Nodes Have Been Installed



Node Dashboard

There are 10 nodes available in the system.

You may regroup nodes by dragging a node into the desired group. You may drop a node [here] to reset to *automatic grouping*.

- ● admin
  - ● admin
- ● ceph
  - ● ceph1
  - ● ceph2
- ● compute
  - ● kvm-compute1
  - ● kvm-compute2
  - ● kvm-compute3
- ● control
  - ● control
  - ● control-net
- ● swift
  - ● swift1
  - ● swift2

SUSE Cloud v2.0 Provided by SUSE

4.3 Post-Installation Configuration

The following lists some *optional* configuration steps like configuring node updates, monitoring, access and SSL-enablement. You may entirely skip the following steps or perform them any of them at a later stage.

4.3.1 Configuring Node Updates with the Updater Barclamp

In order to keep the operating system and the SUSE Cloud software itself up-to-date on the nodes, you can either deploy the Updater Barclamp or the SUSE Manager Barclamp. While the latter requires access to a SUSE Manager server, the Updater Barclamp uses zypper to install updates and patches from repositories made available on the Administration Server.

The easiest way to provide the required repositories on the Administration Server is to set up an SMT server as described in Section 3.1.8, “SMT Configuration” (page 36) and Section 3.2.1, “Setting up the SMT Repositories” (page 38). Alternatives to setting up an SMT server are described in Section 2.2, “Product and Update Repositories” (page 17).

To deploy the Updater Barclamp, proceed as follows. For general instructions on how to edit Barclamp proposals refer to Section 5.1, “Barclamps” (page 70).

1. Open a browser and point it to the Crowbar Web interface available at port `3000` of the Administration Server, for example `http://192.168.124.10:3000/`. Log in as user `crowbar`. The password defaults to `crowbar`, if you have not changed it during the installation.

2. Open the Barclamp menu by clicking *Barclamps* > *Crowbar*. Click the *Updater* Barclamp entry and *Create* to open the proposal.

3. Configure the Barclamp by the following attributes. This configuration always applies to all nodes on which the Barclamp is deployed. Creating individual configurations for certain nodes is not supported.

   Use zypper

   Define which zypper subcommand to use for updating. *patch* will install all patches applying to the system from the configured update repositories that are available. *update* will update packages from all configured repositories (not just the update repositories) that have a higher version number as the installed packages. *dist-upgrade* replaces each package installed with the version from the repository and deletes packages not available in the repositories.

   Using *patch* is recommended.

   Enable GPG Checks

   If set to true (recommended), checks if packages are correctly signed.

   Automatically Agree With Licenses

   If set to true (recommended), zypper automatically accepts third party licenses.

   Include Patches that need Reboots (Kernel)

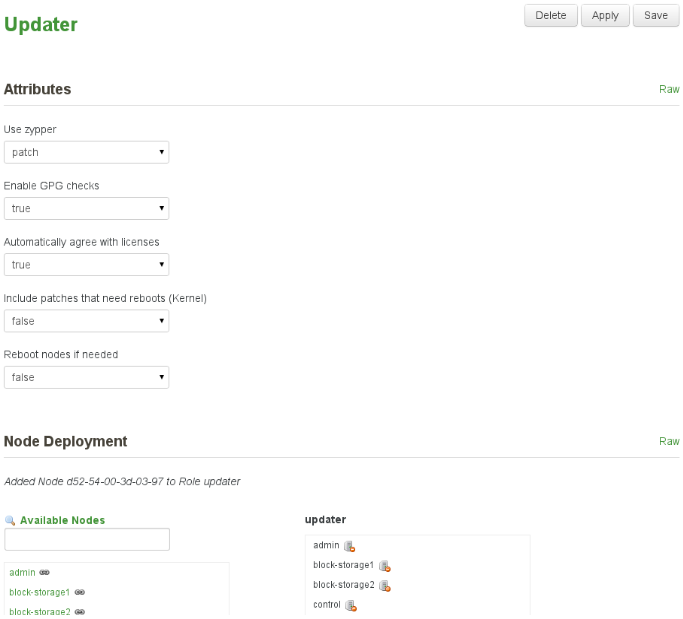
   Installs patches that require a reboot (for example Kernel or glibc updates). It's strongly recommended to set this option to *false* and to install these updates manually when you have a chance to safely reboot the node. Installing a new Kernel and not rebooting may result in an unstable system.

   Reboot Nodes if Needed

   Automatically reboots the system in case a patch requiring a reboot has been installed. It's strongly recommended to set this option to *false* and to install updates requiring a reboot manually when you have a chance to safely reboot the node. Automatically rebooting for example a Compute Node will immediately terminate all instances. Unsaved data on these guests will be lost and running processes will be aborted.

4 Choose the nodes on which the Updater Barclamp should be deployed in the *Node Deployment* section by dragging them to the *Updater* column. It's recommended to deploy it on all nodes in the SUSE Cloud.

Figure 4.6: *SUSE Updater Barclamp*



Use zypper: patch

Enable GPG checks: true

Automatically agree with licenses: true

Include patches that need reboots (Kernel): false

Reboot nodes if needed: false

Added Node d52-54-00-3d-03-97 to Role updater

`zypper` keeps track of the packages and patches it installs in `/var/log/zypp/history`. Review that log file on a node to find out which updates have been installed. A second log file recording debug information on the `zypper` runs can be found at `/var/log/zypper.log` on each node.

4.3.2 Configuring Node Updates with the SUSE Manager Barclamp

In order to keep the operating system and the SUSE Cloud software itself up-to-date on the nodes, you can either deploy SUSE Manager Barclamp or the Updater Barclamp. While the latter uses zypper to install updates and patches from repositories made available on the Administration Server.

To enable the SUSE Manager server to manage the SUSE Cloud nodes, the respective SUSE Cloud 2.0 channels (SUSE-Cloud-2.0-Pool, SUSE-Cloud-2.0-Updates) need to be available on the server. It also requires to generate an `Activation Key` for SUSE Cloud.

The SUSE Manager Barclamp requires access to the SUSE Manager server from every node it is deployed to.

To deploy the SUSE Manager Barclamp, proceed as follows. For general instructions on how to edit Barclamp proposals refer to Section 5.1, “Barclamps” (page 70).

1 Generate an `Activation Key` for SUSE Cloud on the SUSE Manager server. See the section *Activation Keys* at `http://www.suse.com/documentation/suse_manager/book_susemanager_ref/data/s1-sm-systems.html` for instructions).

2 Download the package `rhn-org-trusted-ssl-cert-`*`VERSION-RELEASE`*`.noarch.rpm` from https://*`susemanager.example.com`*/pub/ (*`VERSION`* and *`RELEASE`* may vary, *`susemanager.example.com`* has to be replaced by the address of your SUSE Manager server. Copy the file you just downloaded to `/opt/dell/chef/cookbooks/suse-manager-client/files/default/ssl-cert.rpm` on the Administration Server. The package contains the SUSE Manager's CA SSL Public Certificate. The certificate installation has not been automated on purpose, because downloading the certificate manually enables you to check it before copying it.

3 Re-install the Barclamp by running the following command:

```
/opt/dell/bin/barclamp_install.rb --rpm suse-manager-client
```

4. Open a browser and point it to the Crowbar Web interface available at port `3000` of the Administration Server, for example `http://192.168.124.10:3000/`. Log in as user `crowbar`. The password defaults to `crowbar`, if you have not changed it during the installation.

5. Open the Barclamp menu by clicking *Barclamps* > *Crowbar*. Click the *SUSE Manager Client* Barclamp entry and *Create* to open the proposal.

6. Configure the Barclamp by the following attributes. This configuration always applies to all nodes on which the Barclamp is deployed. Creating individual configurations for certain nodes is not supported.

   Activation Key

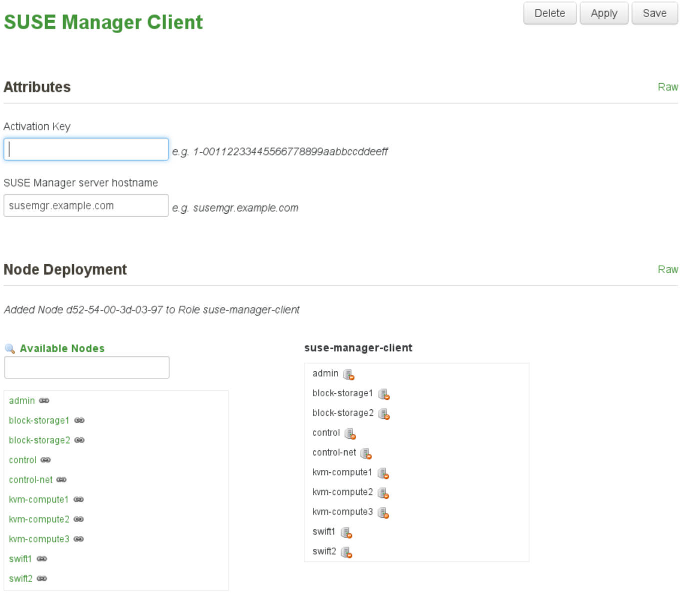
   Enter the SUSE Manager activation key for SUSE Cloud here. This key must have been generated on the SUSE Manager server.

   SUSE Manager Server Hostname

   Fully qualified hostname of the SUSE Manager server. This name must be resolvable via the DNS server on the Administration Server.

7. Choose the nodes on which the SUSE Manager Barclamp should be deployed in the *Node Deployment* section by dragging them to the *suse-manager-client* column. It's recommended to deploy it on all nodes in the SUSE Cloud.

Figure 4.7: *SUSE Manager Barclamp*



SUSE Manager Client

Attributes

Activation Key

e.g. 1-00112233445566778899aabbccddeeff

SUSE Manager server hostname

susemgr.example.com

e.g. susemgr.example.com

Node Deployment

Added Node d52-54-00-3d-03-97 to Role suse-manager-client

Available Nodes

admin
block-storage1
block-storage2
control
control-net
kvm-compute1
kvm-compute2
kvm-compute3
swift1
swift2

suse-manager-client

admin
block-storage1
block-storage2
control
control-net
kvm-compute1
kvm-compute2
kvm-compute3
swift1
swift2

4.3.3 Mounting NFS Shares on a Node

The NFS Barclamp allows you to mount NFS share from a remote host on nodes in the cloud. This feature can, for example, be used to provide an image repository for Glance. Note that all nodes which are to mount an NFS share must be able to reach the NFS server. This requires to manually adjust the network configuration.

To deploy the NFS Barclamp, proceed as follows. For general instructions on how to edit Barclamp proposals refer to Section 5.1, “Barclamps” (page 70).

1. Open a browser and point it to the Crowbar Web interface available at port `3000` of the Administration Server, for example `http://192.168.124.10:3000/`. Log in as user `crowbar`. The password defaults to `crowbar`, if you have not changed it during the installation.

2. Open the Barclamp menu by clicking *Barclamps* > *Crowbar*. Click the *NFS Client* Barclamp entry and *Create* to open the proposal.

3 Configure the Barclamp by the following attributes. Each set of attributes is used to mount a single NFS share.

Name
: Unique name for the current configuration. This name is used in the Web interface only to distinguish between different shares.

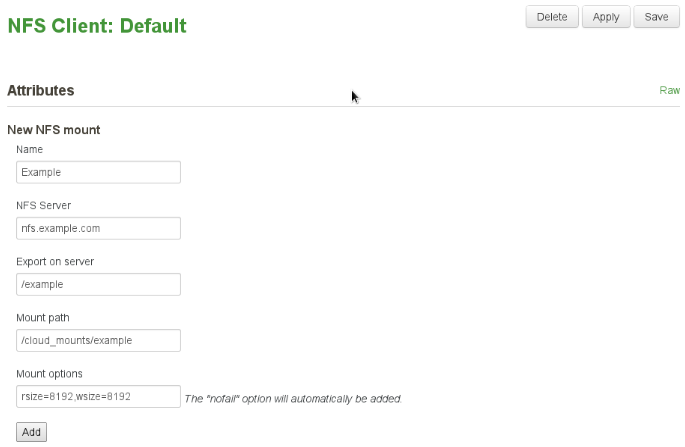
NFS Server
: Fully qualified hostname or IP address of the NFS server.

Export on Server
: Export name for the share on the NFS server.

Mount Options
: Mount options that will be used on the node. See `Man 8 mount` for general mount options and `man 5 nfs` for a list of NFS-specific options. Note that the general option `nofail` (do not report errors if device does not exist) is automatically set.

4 Click *Add* after having filled in all attributes. If you want to mount more than one share, fill in the data for another NFS mount, otherwise click *Save* to save the data, or *Apply* to deploy the proposal. Note that you must always click *Add* before saving or applying the Barclamp, otherwise the data that was entered will be lost.

Figure 4.8: *NFS Barclamp*



NFS Client: Default

Attributes

New NFS mount

Name: Example

NFS Server: nfs.example.com

Export on server: /example

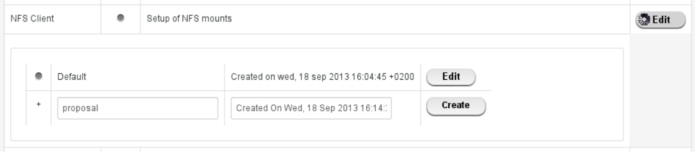
Mount path: /cloud_mounts/example

Mount options: rsize=8192,wsize=8192 *The "nofail" option will automatically be added.*

5 Go to the *Node Deployment* section and drag and drop all nodes, on which the NFS shares defined above should be mounted, to the *nfs-client* column. Click *Apply* to deploy the proposal.

The NFS Barclamp is the only Barclamp that lets you create different proposals, enabling you to be able to mount different NFS shares on different nodes. Once you have created an NFS proposal, a special *Edit* is shown in the Barclamp overview screen of the Crowbar Web interface. Click it to either *Edit* an existing proposal or *Create* a new one. Creating a new proposal requires to give it a unique name.

Figure 4.9: *Editing an NFS Barclamp Proposal*



NFS Client ● Setup of NFS mounts

●	Default	Created on wed, 18 sep 2013 16:04:45 +0200
+	proposal	Created On Wed, 18 Sep 2013 16:14::

4.3.4 Accessing the Nodes

The nodes can only be accessed via SSH from the Administration Server—it is not possible to connect to them from any other host in the network.

The `root` account *on the nodes* has no password assigned, therefore logging in to a node as `root@`*`node`* is only possible via SSH and key authentication. By default, only the key of the `root` of the Administration Server (root@*`admin`*) is enabled to log in via SSH.

In case you have added additional users to the Administration Server and want to give them permission, to log in to the nodes as well, you need to add these user's public SSH keys to `root`'s `authorized_keys` file on all nodes. Proceed as follows:

Procedure 4.1: *Copying SSH Keys to all Nodes*

1 If not already existing, generate an SSH key pair for the user that should be able to log in to the nodes with `ssh-keygen`. Alternatively copy an existing public key with `ssh-copy-id`. Refer to the respective man pages for more information.

2. Log in to the Crowbar Web interface available at port 3000 of the Administration Server, for example http://192.168.124.10:3000/ (username and default password: crowbar).

3. Open the Barclamp menu by clicking *Barclamps* > *All Barclamps*. Click the *Provisioner* Barclamp entry and *Edit* the *Default* proposal.

4. Copy and paste the *public* SSH key of the user into the *Additional SSH Keys* input field. If adding keys for multiple users, note that each key needs to be placed on a new line.

5. Click *Apply* to deploy the keys and save your changes to the proposal.

4.3.5 Enabling SSL (optional)

In order to enable SSL to encrypt communication within the cloud (see Section 2.4, "SSL Encryption" (page 24) for details), the respective certificates need to be available on the nodes running the encrypted services. An SSL certificate is at least required on the Control Node.

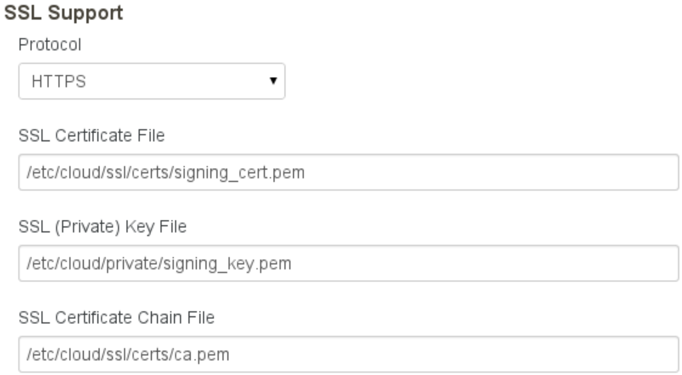
To make them available, copy them to the node. Each certificate consists of a pair of files the certificate file (e.g. signing_cert.pem) and the key file (e.g. signing_key.pem). If you use your own certificate authority (CA) for signing, you will also need a certificate file for the CA (e.g. ca.pem). It is recommended to copy the files to the /etc directory using the directory structure outlined below. If you use a dedicated certificate for each service, create directories named after the services (e.g. /etc/keystone). If sharing the certificates, use a directory such as /etc/cloud.

SSL Certificate File
: /etc/cloud/ssl/certs/signing_cert.pem

SSL Key File
: /etc/cloud/private/signing_key.pem

CA Certificates File
: /etc/cloud/ssl/certs/ca.pem

Figure 4.10: *The SSL Dialogue*



SSL Support

Protocol

HTTPS

SSL Certificate File

/etc/cloud/ssl/certs/signing_cert.pem

SSL (Private) Key File

/etc/cloud/private/signing_key.pem

SSL Certificate Chain File

/etc/cloud/ssl/certs/ca.pem

4.4 Editing Allocated Nodes

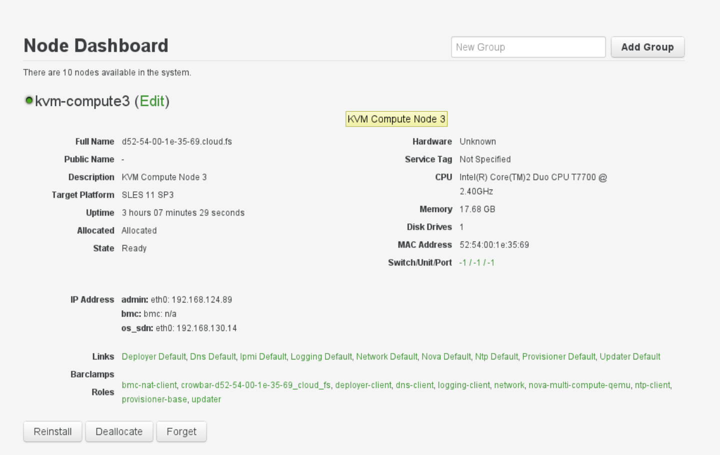
All nodes that have been allocated can be decommissioned or re-installed. Click a node's name in the *Node Dashboard* to open a screen with the node details. The following options are available:

Reinstall
: Triggers a reinstallation. The machine stays allocated.

Deallocate
: Temporarily removes the node from the pool of nodes. Once you reallocate the node it will take its former role. Useful for adding additional machines in times of high load or for decommissioning machines in times of low load.

Forget
: Deletes a node from the pool. If you want to re-use this node again, it needs to be reallocated and re-installed from scratch.

Figure 4.11: *Node Information*



Node Dashboard

There are 10 nodes available in the system.

● kvm-compute3 (Edit)

KVM Compute Node 3

Full Name d52-54-00-1e-35-69.cloud.fs
Public Name -
Description KVM Compute Node 3
Target Platform SLES 11 SP3
Uptime 3 hours 07 minutes 29 seconds
Allocated Allocated
State Ready

Hardware Unknown
Service Tag Not Specified
CPU Intel(R) Core(TM)2 Duo CPU T7700 @ 2.40GHz
Memory 17.68 GB
Disk Drives 1
MAC Address 52:54:00:1e:35:69
Switch/Unit/Port -1 / -1 / -1

IP Address **admin:** eth0: 192.168.124.89
bmc: bmc: n/a
os_sdn: eth0: 192.168.130.14

Links Deployer Default, Dns Default, Ipmi Default, Logging Default, Network Default, Nova Default, Ntp Default, Provisioner Default, Updater Default
Barclamps
Roles bmc-nat-client, crowbar-d52-54-00-1e-35-69_cloud_fs, deployer-client, dns-client, logging-client, network, nova-multi-compute-qemu, ntp-client, provisioner-base, updater

WARNING: Editing Nodes in a Production System

When deallocating nodes that provide essential services, the complete cloud will become unusable. While it is uncritical to disable single storage nodes (provided you have not disabled redundancy) or single compute nodes, disabling Control Node(s) will cause major problems. It will either “kill” certain services (for example Swift) or, at worst (when deallocating the Control Node hosting Neutron) the the complete cloud. You should also not disable nodes providing Ceph monitoring services or the nodes providing swift ring and proxy services.

Deploying the OpenStack Services 5

Once the nodes are installed and configured you can start deploying the OpenStack services in order to finalize the installation. The services need to be deployed in a given order, because they depend on one another. Deployment is done from the Crowbar Web interface through recipes, so-called “Barclamps”.

The services controlling the cloud (including storage management and control services) need to be installed on the Control Node(s). However, you may *not* use your Control Node(s) as a compute node or storage host for Swift or Ceph. Here is a list with services that may *not* be installed on the Control Node(s): *swift-storage*, *Ceph-store*, *Nova-multi-compute*. These services need to be installed on dedicated nodes.

The OpenStack services need to be deployed in the following order. For general instructions on how to edit and deploy Barclamp, refer to Section 5.1, “Barclamps” (page 70). Deploying Swift and Ceph is optional; all other services must be deployed.

1. Deploying the Database

2. Deploying Keystone

3. Deploying RabbitMQ

4. Deploying Swift (optional)

5. Deploying Ceph (optional, unsupported)

> **IMPORTANT: Ceph not Supported**
>
> Ceph is included in SUSE Cloud 2.0 as a technology preview. Customers can use this in test environments but it is not recommended for production. Supported block storage is provided by Cinder.

6. Deploying Glance

7. Deploying Cinder

8. Deploying Neutron

9. Deploying Nova

10 Deploying Horizon (Nova Dashboard)

5.1 Barclamps

The OpenStack services are automatically installed on the nodes by using so-called Barclamps—a set of recipes, templates, and installation instructions. All existing Barclamps can be accessed from the Crowbar Web interface by clicking on *Barclamps*. To edit a Barclamp, proceed as follows:

1. Open a browser and point it to the Crowbar Web interface available at port `3000` of the Administration Server, for example `http://192.168.124.10:3000/`. Log in as user `crowbar`. The password defaults to `crowbar`, if you have not changed it.

   Click *Barclamps* to open the *All Barclamps* menu. Alternatively you may filter the list to *Crowbar* or *OpenStack* Barclamps by choosing the respective option from *Barclamps*. The *Crowbar* Barclamps contain general recipes for setting up and configuring all nodes, while the *OpenStack* are dedicated to OpenStack service deployment and configuration.

2. You can either *Create* a proposal or *Edit* an existing one.

   Most OpenStack Barclamps consist of two sections: the *Attributes* section lets you change the configuration, and the *Node Deployment* section lets you choose onto which nodes to deploy the Barclamp.

3 To edit the *Attributes* section, change the values via the Web form. Alternatively you can directly edit the configuration file by clicking *Raw*.

WARNING: Raw Mode

If you switch between *Raw* mode and Web form (*Custom* mode), make sure to *Save* your changes before switching, otherwise they will be lost.

In the *Node Deployment* section of the OpenStack Barclamp you can drag and drop nodes from the *Available Nodes* column to the desired role. You need to drop the node onto the role name. Do *not* drop a node onto the input field—this is rather used to filter the list of *Available Nodes*!

One or more nodes are usually automatically pre-selected for available roles. If this pre-selection does not meet your requirements, remove it *before* dragging new nodes to the role. To remove a node from a role, click the respective *Remove* icon.

4 To save and deploy your edits, click *Apply*. To just save your changes without deploying them, click *Save*. To remove the complete proposal, click *Delete*. A proposal that already has been deployed can only be deleted manually, see Section 5.1.1, “Delete a Proposal that Already has been Deployed” (page 72) for details.

If you deploy a proposal onto a node where a previous one is still active, the new proposal will overwrite the old one.

NOTE: Wait Until a Proposal has been Deployed

Deploying a proposal might take some time (up to several minutes). It is strongly recommended to always wait until you see the note “Successfully applied the proposal” before proceeding on to the next proposal.

WARNING: Barclamp Deployment Failure

In case the deployment of a Barclamp fails, make sure to fix the reason that has caused the failure and deploy the Barclamp again. Refer to the respective troubleshooting section at Q & A 6.1.2, “OpenStack Node Deployment” (page 97) for help. A deployment failure may leave your node in an inconsistent state.

5.1.1 Delete a Proposal that Already has been Deployed

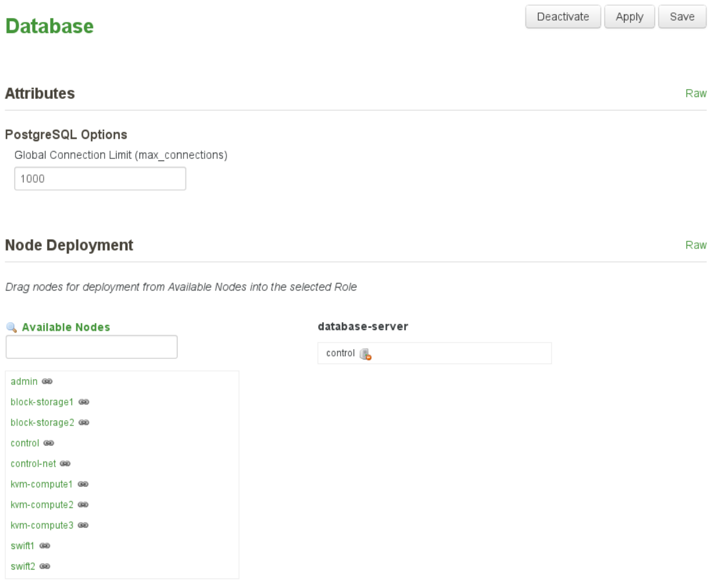
To delete a proposal that already has been deployed, you first need to *Deactivate* it in the Crowbar Web interface. Deactivating a proposal will remove software and services having been deployed by this proposal from the affected nodes. After a proposal has been deactivated, you can *Delete* it in the Crowbar Web interface and *Create* a new proposal from the Barclamp overview.

5.2 Deploying the Database

The very first service that needs to be deployed is the *Database*. The database service using PostgreSQL is used by all other services. It must be installed on a Control Node.

The only attribute you may change is the maximum number of database connections (*Global Connection Limit*). The default value should work in most cases—only change it for large deployments in case the log files show database connection failures.

Figure 5.1: *The Database Barclamp*



Database

Attributes

PostgreSQL Options

Global Connection Limit (max_connections): 1000

Node Deployment

Drag nodes for deployment from Available Nodes into the selected Role

Available Nodes

- admin
- block-storage1
- block-storage2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3
- swift1
- swift2

database-server

- control

5.3 Deploying Keystone

Keystone is another core component that is used by all other OpenStack services. It provides authentication and authorization services. *Keystone* needs to be installed on a Control Node. You can configure the following parameters of this Barclamp:

Algorithm for Token Generation
: Set the algorithm used by Keystone to generate the tokens. It's strongly recommended to use `PKI`, since it will reduce network traffic.

Default Tenant
: Tenant for the users. Do not change the default value of `openstack`.

Regular User/Administrator Username/Password
: Username and password for the regular user and the administrator. Both accounts can be used to log in to the SUSE Cloud Dashboard to manage Keystone users and access.

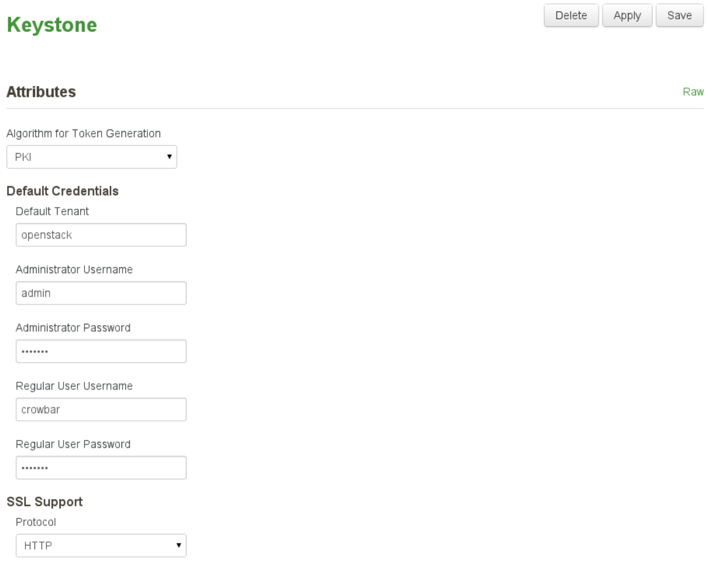
Protocol
: When sticking with the default value *HTTP*, public communication will not be encrypted . Choose *HTTPS* to use SSL for encryption. See Section 2.4, “SSL Encryption” (page 24) for background information and Section 4.3.5, “Enabling SSL (optional)” (page 66) for installation instructions. The following additional configuration options will become available when choosing *HTTPS*:

  SSL Certificate File / SSL (Private) Key File
  : Location of the certificate key pair files.

  SSL Certificate is insecure
  : Check this option if using self-signed certificates in order to disable certificate checks. You should only self-signed certificates for non-production deployments. Always use properly signed certificates in production environments. Never check this option in a production deployment!

  Require Client Certificate / SSL CA Certificates File
  : If your certificates are signed by a trusted third party organization, *Require Client Certificate* should be set to *false*, since the “official” certification authorities (CA) are already known by the system. If the certificates are signed by a CA within your organization or by any other CA not known by the system, a CA certificate is needed.

Figure 5.2: The Keystone Barclamp



Attributes

Algorithm for Token Generation: PKI

Default Credentials

Default Tenant: openstack

Administrator Username: admin

Administrator Password: •••••••

Regular User Username: crowbar

Regular User Password: •••••••

SSL Support

Protocol: HTTP

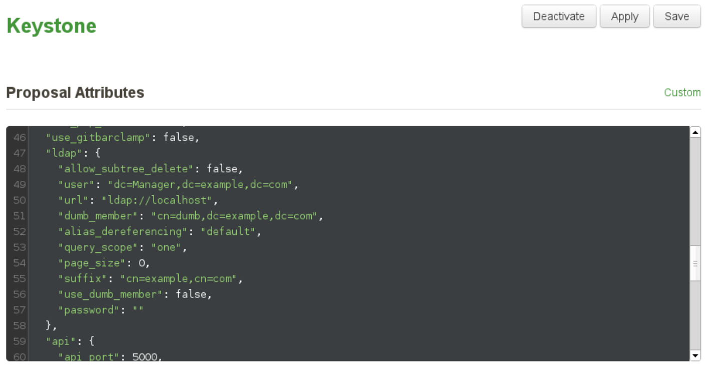
5.3.1 LDAP Authentication with Keystone

By default Keystone uses an SQL database backend store for authentication. Alternatively, LDAP can be used. Using LDAP requires the Control Node on which Keystone is installed to be able to contact the LDAP server. See Appendix B, *The Network Barclamp Template File* (page 105) for instructions on how to adjust the network setup.

To configure LDAP integration, you need to open the Keystone Barclamp *Attribute* configuration in *Raw* mode. Search for the *ldap* section which, by default, looks like the following:

```
"ldap": {
  "password": "",
  "allow_subtree_delete": false,
  "alias_dereferencing": "default",
  "use_dumb_member": false,
  "user": "dc=Manager,dc=example,dc=com",
  "suffix": "cn=example,cn=com",
  "url": "ldap://localhost",
  "page_size": 0,
  "query_scope": "one",
  "dumb_member": "cn=dumb,dc=example,dc=com"
},
```

Figure 5.3: The Keystone Barclamp: Raw Mode



```
  "use_gitbarclamp": false,
  "ldap": {
    "allow_subtree_delete": false,
    "user": "dc=Manager,dc=example,dc=com",
    "url": "ldap://localhost",
    "dumb_member": "cn=dumb,dc=example,dc=com",
    "alias_dereferencing": "default",
    "query_scope": "one",
    "page_size": 0,
    "suffix": "cn=example,cn=com",
    "use_dumb_member": false,
    "password": ""
  },
  "api": {
    "api_port": 5000,
```

Adjust the settings according to your LDAP setup. The default configuration does not include all attributes that can be set—a complete list of options is available in the file `/opt/dell/chef/data_bags/crowbar/bc-template-keystone.schema` on the Administration Server (search for `ldap`). There are three types of attribute values: strings (e.g. the value for `url:"ldap://localhost"`), bool (e.g. the value for `use_dumb_member: false`) and integer (e.g. the value for `page_size: 0`). Attribute names and string values always need to be quoted with double quotes, bool and integer values must not be quoted.

IMPORTANT: Using LDAP over SSL (ldaps) is recommended

In a production environment, it is recommended to use LDAP over SSL (ldaps), otherwise passwords will be transferred as plain text.

Apart from the LDAP configuration, you need to replace Keystone's the default SQL identity driver with the hybrid driver. Search for the `identity` section and replace the value for *driver*, resulting in the following code:

```
"identity": {
  "driver": "keystone.identity.backends.hybrid.Identity"
},
```

5.4 Deploying RabbitMQ

The RabbitMQ messaging system enables services to communicate with the other nodes via Advanced Message Queue Protocol (AMQP). Deploying it is mandatory. RabbitMQ needs to be installed on a Control Node. It is recommended not to change the default values of the proposal's attributes.

Virtual Host

Name of the default virtual host to be created and used by the RabbitMQ server (`default_vhost` config option in `rabbitmq.config`).

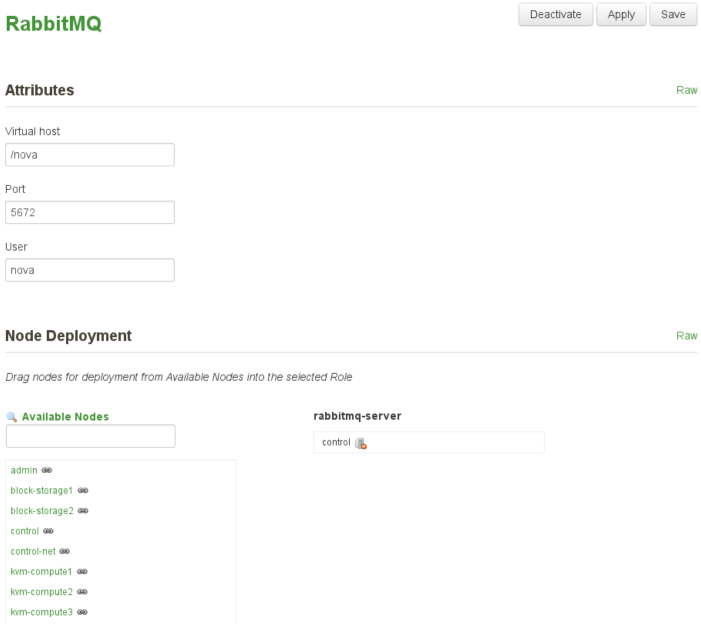
Port

Port the RabbitMQ server listens on (`tcp_listeners` config option in `rabbitmq.config`).

User

RabbitMQ default user (`default_user` config option in `rabbitmq.config`).

Figure 5.4: *The RabbitMQ Barclamp*



RabbitMQ

Attributes

Virtual host

/nova

Port

5672

User

nova

Node Deployment

Drag nodes for deployment from Available Nodes into the selected Role

Available Nodes

- admin
- block-storage1
- block-storage2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3

rabbitmq-server

- control

5.5 Deploying Swift (optional)

Swift adds an object storage service to SUSE Cloud that lets you store single files such as images or snapshots. It offers high data security by storing the data redundantly on a pool of Storage Nodes—therefore Swift needs to be installed on at least two dedicated nodes.

In order to be able to properly configure Swift it's important to understand how it places the data. Data is always stored redundantly within the hierarchy. The Swift hierarchy in SUSE Cloud is formed out of zones, nodes, hard drives and logical partitions. Zones are physically separated clusters, for example different server rooms each with it's own power supply and network segment. A failure of one zone must not affect another zone. The next level in the hierarchy are the individual Swift storage nodes (on which *swift-storage* has been deployed) followed by the hard drives. Logical partitions come last.

Swift automatically places three copies of each object on the highest hierarchy level possible. If three zones are available, the each copy of the object will be placed in a different zone. In a one zone setup with more than two nodes, the object copies will each be stored on a different node. In a one zone setup with two nodes, the copies will be distributed on different hard disks. If no other hierarchy element fits, logical partitions are used.

The following attributes can be set to configure Swift:

Allow Public Containers
: Allows to enable public access to containers id set to `true`.

Zones
: Number of zones (see above). If you do not have different independent installations of storage nodes, set the number of zones to `1`.

Create 2^X Logical Partitions
: Partition power. The number entered here is used to compute the number of logical partitions to be created in the cluster by using it as a power of 2 (2^X).

  It's recommended to use a minimum of 100 partitions per disk. To measure the partition power for your setup, do the following: Multiply the number of disks from all Swift nodes with 100 and then round up to the nearest power of two. Keep in mind that the first disk of each node is not used by Swift, but rather for the operating system.

Example: 10 Swift nodes with 5 HDD each Four hard disks on each node are used for Swift, so there is a total of forty disks. Multiplied with 100 gives 4000. The nearest power of two, 4096, equals 2^12. So the partition power that needs to be entered is `12`.

> **IMPORTANT: Value Cannot be Changed Once the Proposal Has Been Deployed**
>
> Changing the number of logical partition after Swift has been deployed is not supported. Therefore the value for the partition power should be calculated from the maximum number of partitions this cloud installation is likely going to need at any point in time.

Minimum Hours before Partition is reassigned

This option sets the number of hours before a logical partition is considered for relocation. `24` is the recommended value.

Replicas

The number of copies generated for each object. Set this value to `3`, the tested and recommended value.

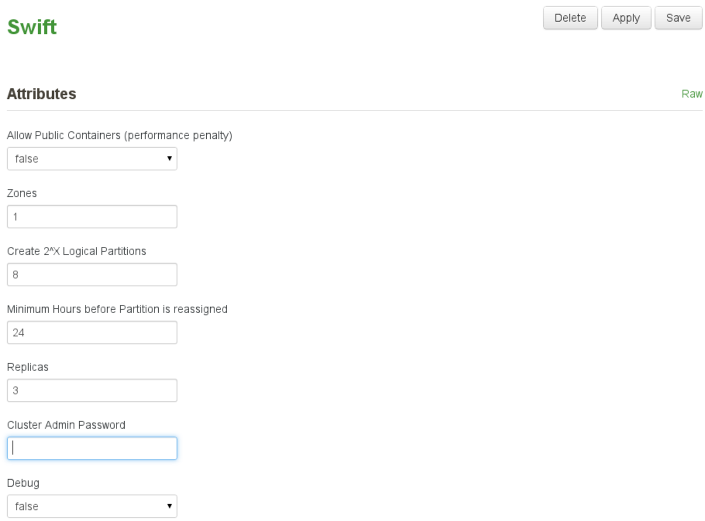
Cluster Admin Password

The Swift administrator password.

Debug

Shows debugging output in the log files when set to `true`.

Figure 5.5: *The Swift Barclamp*



Swift

Attributes

Allow Public Containers (performance penalty)
false

Zones
1

Create 2^X Logical Partitions
8

Minimum Hours before Partition is reassigned
24

Replicas
3

Cluster Admin Password

Debug
false

Apart from the general configuration described above, the Swift Barclamp lets you also activate and configure *Additional Middlewares*. The features these middlewares provide can be used via the Swift command line client only. The Ratelimit and S3 middlewares certainly provide for the most interesting features, whereas it is recommended to only enable further middlewares for specific use-cases.

S3

Provides an S3 compatible API on top of Swift.

StaticWeb

Enables to serve container data as a static web site with an index file and optional file listings. See `http://docs.openstack.org/developer/swift/misc.html#module-swift.common.middleware.staticweb` for details.

This middleware requires to set *Allow Public Containers* to `true`.

TempURL

Enables to create URLs to provide time limited access to objects. See `http://docs.openstack.org/developer/swift/misc.html#module-swift.common.middleware.tempurl` for details.

FormPOST

Enables to upload files to a container via web form. See `http://docs.openstack.org/developer/swift/misc.html#module-swift.common.middleware.formpost` for details.

Domain Remap

Translates container and and account parts of a domain to path parameters that the Swift proxy server understands. Can be used to create short URLs that are easy to remember, for example by rewriting `home.tux.example.com/$ROOT/exampleuser;/home/myfile` to `home.tux.example.com/myfile`. See `http://docs.openstack.org/developer/swift/misc.html#module-swift.common.middleware.domain_remap` for details.

CNAME Lookup

CNAME Lookup translates an unknown domain in the host header to something that ends with the configured *Storage Domain* by looking up the given domain's CNAME record in DNS. See `http://docs.openstack.org/developer/swift/misc.html#module-swift.common.middleware.cname_lookup` for details.

Ratelimit

Ratelimit enables you to throttle resources such as requests per minute to provide denial of service protection. See `http://docs.openstack.org/developer/swift/ratelimit.html` for details.

The Swift service consists of four different roles. Deploying *swift-dispersion* is optional:

swift-ring-compute

The ring maintains the information about the location of objects, replicas, and devices. It can be compared to an index, that is used by various OpenStack services to look up the physical location of objects. *swift-ring-compute* must only be installed on a single node; it is recommended to use a Control Node.

swift-proxy

The Swift proxy server takes care of routing requests to Swift. Installing a single instance of *swift-proxy* on a Control Node is recommended.

swift-dispersion

Deploying *swift-dispersion* is optional. The Swift dispersion tools can be used to test the health of the cluster. It creates a heap of dummy objects (using 1% of the total space available). The state of these objects can be queried using the swift-dispersion-report query. *swift-dispersion* needs to be installed on the same node as *swift-proxy*.

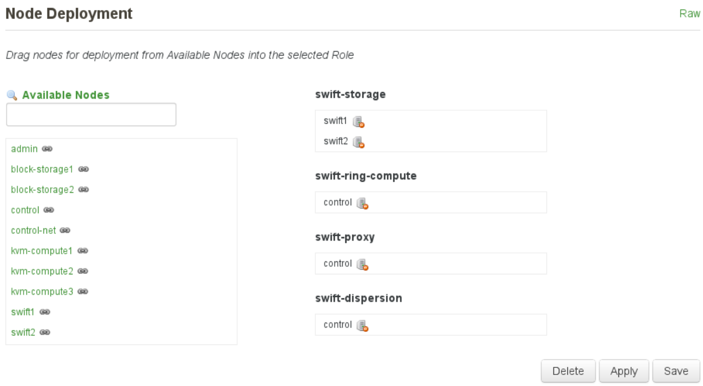
swift-storage

The virtual object storage service. Install this role on all dedicated Swift Storage Nodes (at least two), but not on any other node.

WARNING: swift-storage Needs Dedicated Machines

Never install the swift-storage service on a node that runs other OpenStack services.

Figure 5.6: *The Swift Barclamp: Node Deployment Example*



Node Deployment

Drag nodes for deployment from Available Nodes into the selected Role

Available Nodes

- admin
- block-storage1
- block-storage2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3
- swift1
- swift2

swift-storage

- swift1
- swift2

swift-ring-compute

- control

swift-proxy

- control

swift-dispersion

- control

5.6 Deploying Ceph (optional, unsupported)

Ceph adds a redundant block storage service to SUSE Cloud. It lets you store persistent devices that can be mounted from instances. It offers high data security by storing the data redundantly on a pool of Storage Nodes—therefore Ceph needs to be installed on at least two dedicated nodes.

For more information on the Ceph project, vist http://ceph.com/.

IMPORTANT: Ceph not Supported

Ceph is included in SUSE Cloud 2.0 as a technology preview. Customers can use this in test environments but it is not recommended for production. Supported block storage is provided by Cinder.

The Ceph Barclamp only has one configuration option: telling Ceph which devices to use on the nodes. Enter a space-separated list of devices that should be used by Ceph. For example:

```
dev/sdb /dev/sdc /dev/sdd
```

IMPORTANT: Devices

Not all of the devices used for Ceph need to exist on all nodes. All devices from a node matching the list will be used. They must *not* be mounted prior to deploying Ceph. Any data stored on these devices will be lost.

The Ceph service consists of three different roles:

Ceph-mon-master
: Master cluster monitor daemon for the Ceph distributed file system. *Ceph-mon-master* must only be installed on a single node; it is recommended to use a Control Node.

Ceph-mon
: Cluster monitor daemon for the Ceph distributed file system. *Ceph-mon* needs to be installed on two or four Storage Nodes.

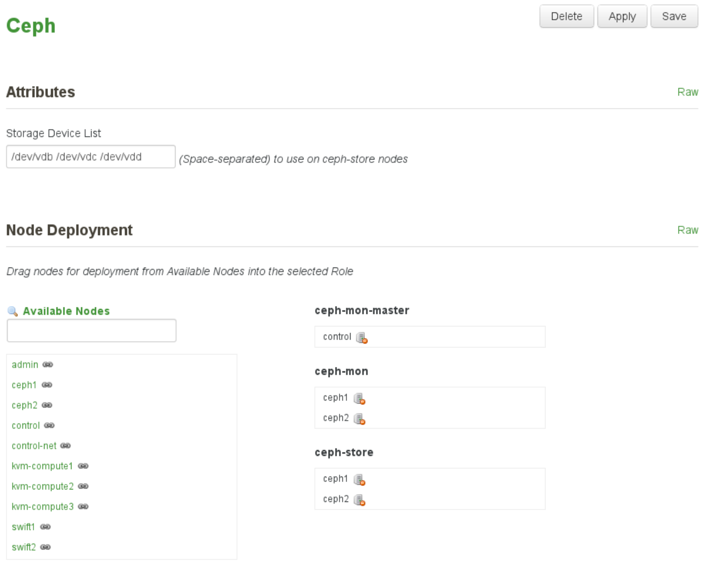
> **IMPORTANT: Number of Ceph Monitor Nodes**
>
> In addition to the node running the Ceph-mon-master service an additional two or four nodes also need to run the *Ceph-mon* service. The sum of the *Ceph-mon-master* and the *Ceph-mon* nodes must always be an odd number (either three or five).
>
> Nodes running *Ceph-mon* cannot be deleted or temporarily be disabled.

Ceph-store

The virtual block storage service. Install this role on all dedicated Ceph Storage Nodes (at least two), but not on any other node.

> **WARNING: *Ceph*-store Needs Dedicated Machines**
>
> Never deploy *Ceph-store* on a node that runs other non-Ceph OpenStack services. The only service that may be deployed together with it is *Ceph-mon*.

Figure 5.7: *The Ceph Barclamp*



Ceph

Attributes — Raw

Storage Device List

/dev/vdb /dev/vdc /dev/vdd *(Space-separated) to use on ceph-store nodes*

Node Deployment — Raw

Drag nodes for deployment from Available Nodes into the selected Role

Available Nodes

- admin
- ceph1
- ceph2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3
- swift1
- swift2

ceph-mon-master

- control

ceph-mon

- ceph1
- ceph2

ceph-store

- ceph1
- ceph2

Deploying Ceph requires to perform the steps in a given order:

1. Edit the Barclamp proposal to specify the devices to be used by Ceph as described above.
2. Drag and drop a node (for example, a Control Node) to the *Ceph-mon-master* role.
3. Drag and drop two or four nodes to the *Ceph-mon* role. Note that the maximum number of *Ceph-mon* nodes cannot exceed four and that the sum of *Ceph-mon-master* and *Ceph-mon* nodes must be odd.

4. Drag and drop all dedicated Ceph Storage Nodes to the *Ceph-store* (at least two). You may also use the nodes with the *Ceph-mon* roles, but not the *Ceph-mon-master* node (you can add that one later).

5. Click *Apply* to deploy your proposal. This can take some time.

6. If you also want to use the *Ceph-mon-master* as a Storage Node, drag and drop it to the *Ceph-store* role and click *Apply* again. Note that it is not recommended to use a Control Node for non-management purposes such as storage or compute.

5.7 Deploying Glance

Glance provides discovery, registration, and delivery services for virtual disk images. An image is needed to start an instance—it is its pre-installed root-partition. All images you want to use in your cloud to boot instances from, are provided by Glance.

Glance must be deployed onto a Control Node. There are a lot of options to configure Glance. The most important ones are explained below—for a complete reference refer to `http://github.com/crowbar/crowbar/wiki/Glance--barclamp`.

Notification Strategy
: Glance notifications can be used for auditing and troubleshooting. By default (*Noop*) are disabled. When choosing *RabbitMQ*, notifications are send to the RabbitMQ service.

Backing Type
: Choose whether to use Swift to store the images or to store them in an image file on the Control Node. If you have deployed Swift, it is recommended to use it for Glance as well.

Image Store Directory
: This option is only available when having chosen *Backing Type* > *File*. Specify the directory to host the image file. The directory specified here can also be an NFS share. See Section 4.3.3, “Mounting NFS Shares on a Node” (page 63) for more information.

Glance Swift Container
: This option is only available when having chosen *Backing Type* > *Swift*. Sets the name of the container to use for the images in Swift.

API: Bind to All Addresses

Set this option to *true* to enable users to upload images to Glance. If unset, only the operator will be able to upload images.

Verbose

Shows debugging output in the log files when set to *true*.

Caching

Enable and configure image caching in this section. By default, image caching is disabled. Learn more about Glance's caching feature at `http://docs.openstack.org/developer/glance/cache.html`.

Database: SQL Idle Timeout

Time after which idle database connections will be dropped.

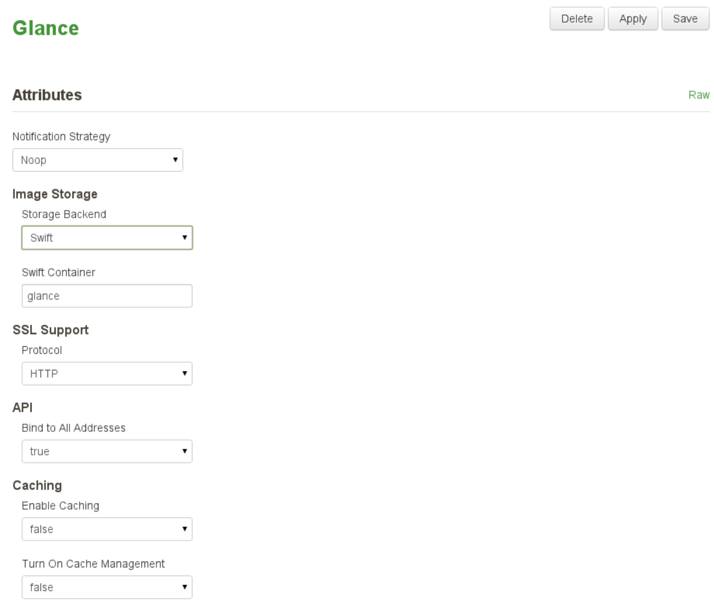
Use Syslog

Use syslog logging if set to *true*.

Protocol

Enable encrypted communication for Glance by choosing *HTTPS*. Refer to “*Protocol* ” (page 73) for configuration details.

Figure 5.8: *The Glance Barclamp*



Glance

Attributes

Notification Strategy
Noop

Image Storage
Storage Backend
Swift

Swift Container
glance

SSL Support
Protocol
HTTP

API
Bind to All Addresses
true

Caching
Enable Caching
false

Turn On Cache Management
false

5.8 Deploying Cinder

Cinder, the successor of Nova Volume, provides volume block storage. It adds persistent storage to a instance that will persist until deleted (contrary to ephemeral volumes that will only persist until the instance is running).

Cinder can provide volume storage by using a local file, one or more local disks, or the Dell EqualLogic SAN (EQLX). The latter is only available on specific Dell hardware. Using a local file is not recommended for production systems for performance reasons.

The following attributes can be set to configure Cinder:

Type of Volume
Choose the volume type for Cinder:

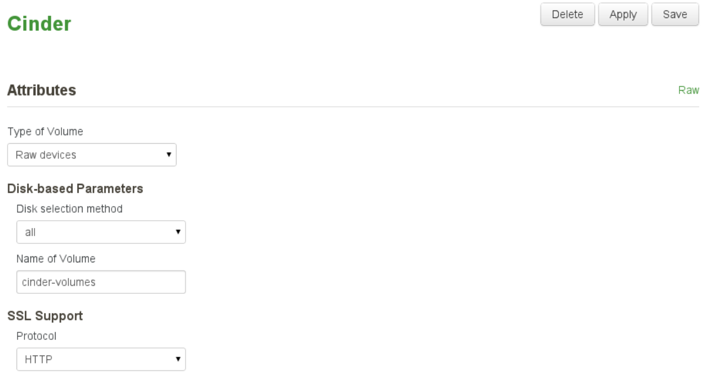
- *Raw devices*: local disk(s)
- *Local file*: local file

- *NetApp*
- *EMC*
- *Equalogic* (Dell EqualLogic SAN)
- Other driver

Disk-based Parameters
: This option is only available when having chosen *Type of Volume* > *raw*. Choose whether to only use the *first* available disk or *all* available disks.

File-based Parameters
: This option is only available when having chosen *Type of Volume* > *local* Choose the name and path for the file containing the volumes and set the *Maximum File Size*. Make sure not to overcommit the size, since it will result in data loss.

Figure 5.9: *The Cinder Barclamp*


Cinder

Attributes

Type of Volume: Raw devices

Disk-based Parameters

Disk selection method: all

Name of Volume: cinder-volumes

SSL Support

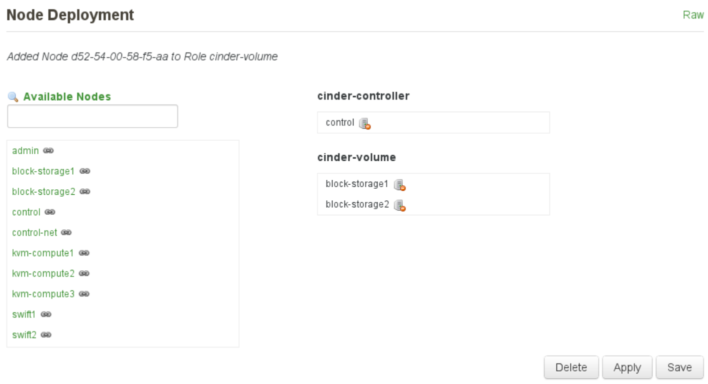
Protocol: HTTP

The Cinder service consists of two different roles:

cinder-controller
: The Cinder controller provides the scheduler and the API. Installing *cinder-controller* on a Control Node is recommended.

cinder-volume
: The virtual block storage service. It can be installed on a Control Node, but it's recommended to deploy it on one or more dedicated nodes supplied with sufficient networking capacity, since it will generate a lot of network traffic.

Figure 5.10: The Cinder Barclamp: Node Deployment Example



Node Deployment

Added Node d52-54-00-58-f5-aa to Role cinder-volume

Available Nodes

admin
block-storage1
block-storage2
control
control-net
kvm-compute1
kvm-compute2
kvm-compute3
swift1
swift2

cinder-controller

control

cinder-volume

block-storage1
block-storage2

5.9 Deploying Neutron

Neutron provides network connectivity between interface devices managed by other OpenStack services (most likely Nova). The service works by enabling users to create their own networks and then attach interfaces to them.

Neutron must be deployed on a Control Node. The following attributes can be set to configure Neutron:

Plugin
: Choose the plugin to be used with Neutron. The *linuxbridge* plugin only supports VLANs in SUSE Cloud, whereas the *openvswitch* plugin supports GRE and flat networks. If you plan to enable the VMware ESXi support, you must choose *linuxbridge*.

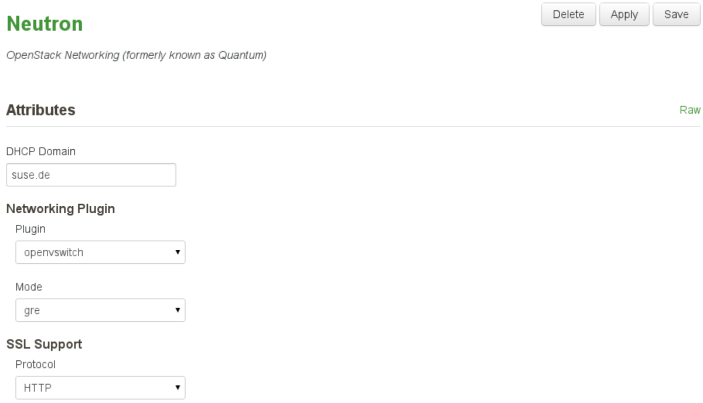
Mode
: This option is only available when having chosen *Plugin* > *openvswitch*. Set the network type to be set up by the plugin: *gre* (Generic Routing Encapsulation) or *flat*.

DHCP Domain
: Domain to use for building the host names.

Protocol

Enable encrypted communication for Neutron by choosing *HTTPS*. Refer to "*Protocol* " (page 73) for configuration details.

Figure 5.11: *The Neutron Barclamp*



Neutron

OpenStack Networking (formerly known as Quantum)

Attributes

DHCP Domain: suse.de

Networking Plugin

Plugin: openvswitch

Mode: gre

SSL Support

Protocol: HTTP

5.10 Deploying Nova

Nova provides key services for managing the SUSE Cloud, sets up the Compute Nodes. SUSE Cloud currently supports KVM, Xen and Microsoft Hyper V. Support for VMware ESX is included as a technology preview. The unsupported QEMU option is included to enable test setups with virtualized nodes. The following attributes can be configured for Nova:

Verbose

Shows debugging output in the log files when set to *true*.

Setup Shared Storage

Sets up a directory `/var/lib/nova/instances` on the Control Node on which *nova-multi-controller* and exports it via NFS to all compute nodes. This setup is required for live migration of Xen instances (but not for KVM) and can be used to provide central handling of instance data. Enabling this option is only recommended if Xen live migration is required—otherwise it should be disabled.

Enable Libvirt Migration

Allows to move KVM and Xen instances to a different Compute Node running the same hypervisor (cross hypervisor migrations are not supported). Useful when a Compute Node needs to be shut down or rebooted for maintenance or when the load of the Compute Node is very high. instances can be moved while running (Live Migration).

> **WARNING: Libvirt Migration and Security**
>
> Enabling the libvirt migration option will open a TCP port on the Compute Nodes that allows to access to all instances from all machines in the admin network. Please ensure that only authorized machines have access to the admin network when enabling this option.

Virtual RAM to Physical RAM allocation ratio

Set the "overcommit ratio" for RAM for instances on the Compute Nodes. A ratio of `1.0` means no overcommitment. Changing this value is not recommended.

Virtual CPU to Physical CPU allocation ratio

Set the "overcommit ratio" for CPUs for instances on the Compute Nodes. A ratio of `1.0` means no overcommitment.

KVM Options: Enable Kernel Samepage Merging

Kernel SamePage Merging (KSM) is a Linux Kernel feature which merges identical memory pages from multiple running processes into one memory region. Enabling it optimizes memory usage on the Compute Nodes when using the KVM hypervisor at the cost of slightly increasing CPU usage.

VMware Support

Setting up VMware support is described in a separate section. See Appendix D, *VMware ESX Installation Instructions* (page 121).

SSL Support: Protocol

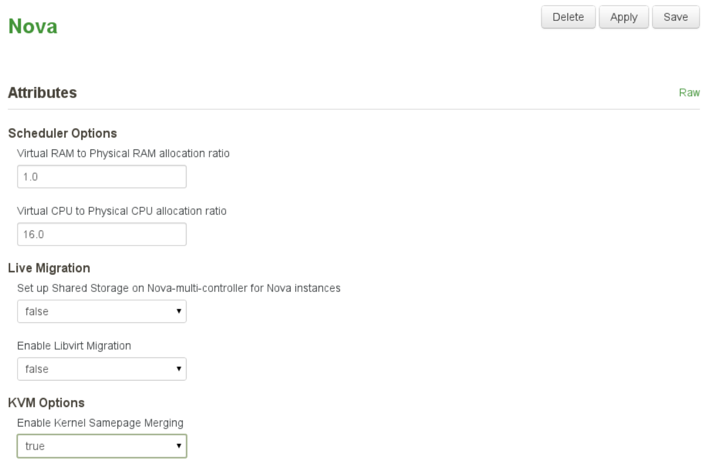
Enable encrypted communication for Nova by choosing *HTTPS*. Refer to "*Protocol* " (page 73) for configuration details.

SSL Support for noVNC: Protocol

After having started an instance you can display its VNC console in the Nova Dashboard (Horizon) via the browser using the noVNC implementation. By default this connection is not encrypted and can potentially be eavesdropped.

Enable encrypted communication for noVNC by choosing *HTTPS*. Refer to "*Protocol* " (page 73) for configuration details.

Figure 5.12: *The Nova Barclamp*



Nova

Attributes

Scheduler Options

Virtual RAM to Physical RAM allocation ratio: 1.0

Virtual CPU to Physical CPU allocation ratio: 16.0

Live Migration

Set up Shared Storage on Nova-multi-controller for Nova instances: false

Enable Libvirt Migration: false

KVM Options

Enable Kernel Samepage Merging: true

The Nova service consists of four different roles:

nova-multi-controller

Distributing and scheduling the instances is managed by the *Nova-multi-controller*. It also provides networking and messaging services. *Nova-multi-controller* needs to be installed on a Control Node.

nova-multi-compute-esxi / *nova-multi-compute-hyperv* / *nova-multi-compute-kvm* / *nova-multi-compute-qemu* / *nova-multi-compute-xen*

Provides the hypervisors (Hyper-V, KVM, QEMU, VMware ESX and Xen) and tools needed to manage the instances. Only one hypervisor can be deployed on a single compute node but you can use different hypervisors in your cloud by deploying different hypervisors to different Compute Nodes. A `Nova-multi-compute` role needs to be installed on every Compute Node. However, not all hypervisors need to be deployed.

Each image that will be made available in SUSE Cloud to start a instance is bound to a hypervisor. Each hypervisor can be deployed on multiple Compute Nodes (except for the VMware ESX role, see below). In a multi-hypervisor deployment you should make sure to deploy the `Nova-multi-compute` roles in a way, that enough compute power is available for each hypervisor.

NOTE: Re-assigning Hypervisors

Existing `Nova-multi-compute` can be changed in a productive SUSE Cloud without service interruption. You need to “evacuate” the node, re-assign a new `Nova-multi-compute` role via the Nova Barclamp and *Apply* the change. *Nova-multi-compute-esxi* can only be deployed on a single node.

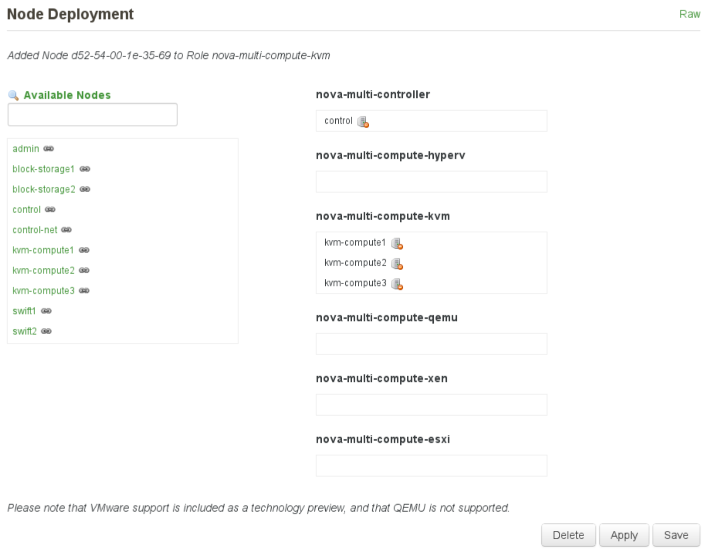
IMPORTANT: Deploying Hyper-V

nova-multi-compute-hyperv can only be deployed to Compute Nodes running either Microsoft Hyper-V Server or Windows Server 2012. Being able to set up such Compute Nodes requires to set up a netboot environment for Windows. Refer to Appendix C, *Setting up a Netboot Environment for Microsoft* Windows* (page 117) for details.

IMPORTANT: Deploying VMware ESX (esxi)

VMware ESX is not supported “natively” by SUSE Cloud—it rather delegates requests to an existing vCenter. It requires preparations at the vCenter and post install adjustments of the Compute Node. See Appendix D, *VMware ESX Installation Instructions* (page 121) for instructions. *Nova-multi-compute-esxi* can only be deployed on a single Compute Node.

Figure 5.13: *The Nova Barclamp: Node Deployment Example with Three KVM Nodes*



Node Deployment

Added Node d52-54-00-1e-35-69 to Role nova-multi-compute-kvm

Available Nodes

- admin
- block-storage1
- block-storage2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3
- swift1
- swift2

nova-multi-controller
- control

nova-multi-compute-hyperv

nova-multi-compute-kvm
- kvm-compute1
- kvm-compute2
- kvm-compute3

nova-multi-compute-qemu

nova-multi-compute-xen

nova-multi-compute-esxi

Please note that VMware support is included as a technology preview, and that QEMU is not supported.

5.11 Deploying Horizon (Nova Dashboard)

The last service that needs to be deployed is Horizon, the Nova Dashboard. It provides a Web interface for users to start and stop instances and for administrators to manage users, groups, roles, etc. Horizon should be installed on a Control Node.

The following attributes can be configured:

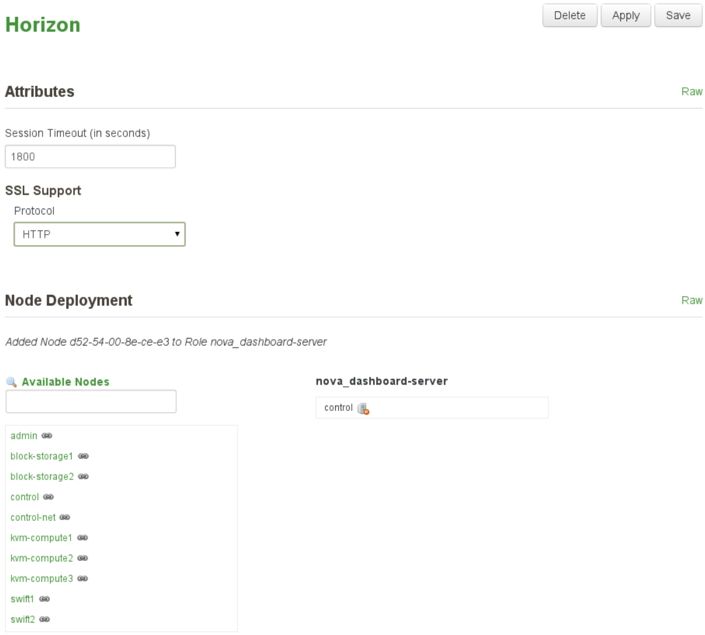
Session Timeout

> Timeout (in seconds) after which a user is been logged out automatically. The default value is set to 30 minutes (1800 seconds).

SSL Support: Protocol

> Enable encrypted communication for Nova Dashboard by choosing *HTTPS*. Refer to "*Protocol* " (page 73) for configuration details.

Figure 5.14: *The Horizon Barclamp*



Horizon

Attributes

Session Timeout (in seconds)

1800

SSL Support

Protocol

HTTP

Node Deployment

Added Node d52-54-00-8e-ce-e3 to Role nova_dashboard-server

Available Nodes

- admin
- block-storage1
- block-storage2
- control
- control-net
- kvm-compute1
- kvm-compute2
- kvm-compute3
- swift1
- swift2

nova_dashboard-server

- control

5.12 How to Proceed

With a successful deployment of the Nova Dashboard, the SUSE Cloud installation is finished. In order to be able to test your setup by starting an instance one last step remains to be done—uploading an image to the Glance service. Refer to Section "Managing Images" (Chapter 2, *Using OpenStack Command Line Clients*, ↑*User Guide for Administrators*) for instructions. Images for SUSE Cloud can be built in SUSE Studio—see this blog post for details: `http://blog.susestudio.com/2012/10/kvm-build-format-suse-cloud-support.html`.

Now you can hand over to the cloud administrator to set up users, roles, flavors, etc.—refer to the *User Guide for Administrators* (↑*User Guide for Administrators*) for details. The default credentials for the Nova Dashboard are username `admin` and password `crowbar`.

Troubleshooting and Support 6

6.1 FAQ

Find solutions for the most common pitfalls here. If your problem is not mentioned here, checking the log files on either the Administration Server or the OpenStack nodes may help. A list of log files is available at Appendix A, *Log Files* (page 101).

6.1.1. Admin Node Deployment

Question: What to do when `install-suse-cloud` fails?

A: Please check the script's log file at `/var/log/crowbar/install.log` for error messages.

Question: What to do when `install-suse-cloud` fails on deploying the IPMI/BMC network?

A: As of SUSE Cloud 2.0 it is assumed that each machine can be accessed directly via IPMI/BMC. However, this is not the case on certain blade hardware, where several nodes are accessed via a common adapter. Such a hardware setup causes an error on deploying the IPMI/BMC network. You need to disable IPMI deployment running the following command:

```
/opt/dell/bin/json-edit -a attributes.ipmi.bmc_enable \
-v false /opt/dell/chef/data_bags/crowbar/bc-template-ipmi.json
```

Re-run `install-suse-cloud` after having disabled the IPMI deployment.

Question: Why am I not able to reach the Administration Server from outside the admin network via the bastion network? `route -n` shows no gateway for the bastion network.

A: Make sure the value for the bastion network's `"router_pref":` entry in `/etc/crowbar/network.json` is set to a *lower* value than the `"router_pref":` entry for the admin network.

Question: Can I change the hostname of the Administration Server?

A: No, once you have run `install-suse-cloud` you cannot change the hostname anymore. Services like Crowbar, Chef, and the RabbitMQ will fail when having changed the hostname.

Question: What to do when browsing the Chef Web UI gives a `Tampered with cookie` error?

A: You probably have an old cookie in your browser from a previous Chef installation on the same IP. Remove the cookie named `_chef_server_session_id` and try again.

6.1.2. OpenStack Node Deployment

Question: How can I log in to a node as `root`?

A: By default you cannot directly log in to a node as `root`, because the nodes were set up without a `root` password. You can only log in via SSH from the Administration Server. You should be able to log in to a node with `ssh root@`*`NAME`* where *`NAME`* is the name (alias) of the node.

If name resolution does not work, go to the Crowbar Web interface and open the *Node Dashboard*. Click on the name of the node and look for its *admin (eth0) IP Address*. Log in to that IP address via SSH as user `root`.

Question: What to do if a node refuses to boot or boots into a previous installation?

A: Make sure to change the boot order in the BIOS of the node, so that the first boot option is to boot from the network/PXE boot.

Question: What to do if a node hangs during hardware discovery after the very first PXE boot into the “SLEShammer” image?

A: The `root` login is enabled at a very early state in discovery mode, so chances are high that you can login for debugging purposes as described in Question: *How can I log in to a node as root?* (page 97). If logging in as `root` does not work, you need to set the `root` password manually:

1. Create a directory named `/updates/discovering-pre`

   ```
   mkdir /updates/discovering-pre
   ```

2. Create a hook script `setpw.hook` in the directory created in the previous step:

   ```
   cat > /updates/discovering-pre/setpw.hook <<EOF
   #!/bin/sh
   echo "linux" | passwd --stdin root
   EOF
   ```

3. Make the script executable:

   ```
   chmod a+x  /updates/discovering-pre/setpw.hook
   ```

Question: What to do when a deployed node fails to PXE boot with the following error message: `Could not find kernel image: ../suse-11.3/install/boot/x86_64/loader/linux`?

A: The installation repository at `/srv/tftpboot/suse-11.3/install` on the Administration Server has not been set up correctly to contain the SUSE Linux Enterprise Server 11 SP3 installation media. Please review the instructions at Section 3.2.2, “Setting Up Repositories for Node Deployment” (page 39).

Question: Why does a deployed node hangs at `Unpacking initramfs` during PXE boot?

A: The node probably does not have enough RAM. You need at least 2 GB RAM.

Question: What to do if a node hangs at `Executing AutoYast script: /var/adm/autoinstall/init.d/crowbar_join` after the installation has been finished?

A: Be patient—the AutoYaST script may take a while to finish. If it really hangs, log in to the node as `root` (see Question: *How can I log in to a node as root?* (page 97) for details). Check the log files at `/var/log/crowbar/crowbar_join/*` for errors.

If the node is in a state where login in from the Administration Server is not possible, you need to create a `root` password for it as described in "Direct `root` Login" (page 52). Now re-install the node by going to the node on the Crowbar Web interface and clicking *Reinstall*. After having been re-installed, the node will hang again, but now you will be able to log in and check the log files to find the cause.

Question: Where to find more information when applying a Barclamp proposal fails?

A: Check the Chef client logs on the Administration Server located at `/var/log/crowbar/chef-client/d*.log`. Further information is available from the Chef client logs located on the node(s) affected by the proposal (`/var/log/chef/client.log`), and also from the logs of the service that failed to be deployed. Additional information may be gained from the Crowbar Web UI logs on the Administration Server. For a list of log file locations refer to Appendix A, *Log Files* (page 101).

Question: I have installed a new hard disk on a node that was already deployed. Why is it ignored by Crowbar?

A: When adding a new hard disk to a node that has already been deployed, it can take up to 15 minutes before the new disk is detected.

6.2 Support

Whenever you contact support to help you with a problem on SUSE Cloud, it is strongly recommended that you gather as much information about your system and the problem as possible. For this purpose, SUSE Cloud ships with a tool called supportconfig. It gathers system information such as the current kernel version being used, the hardware, RPM database, partitions, and other items. supportconfig also collects the most important log files, making it easier for the supporters to identify and solve your problem.

It is recommended to always run supportconfig on the Administration Server as well as on the Control Node. If a Compute Node or a Storage Node is part of the problem, run supportconfig on the affected node as well. For details on how to run supportconfig, please refer to http://www.suse.com/documentation/sles11/book_sle_admin/data/cha_adm_support.html.

A Log Files

Find a list of log files below, sorted according to the nodes where they can be found.

A.1 On the Administration Server

- Crowbar Web Interface: `/var/log/crowbar/production.log`
- Chef server: `/var/log/chef/server.log`
- Chef expander: `/var/log/chef/expander.log`
- Chef client (for the Administration Server only): `/var/log/chef/client.log`
- Apache SOLR (Chef's search server): `/var/log/chef/solr.log`
- HTTP (AutoYaST) installation server for provisioner Barclamp: `/var/log/apache2/provisioner-{access,error}_log`
- Default SUSE log files: `/var/log/messages`, `/var/log/zypper.log` etc.
- Syslogs for all nodes: `/var/log/nodes/*.log` (these are collected via remote syslogging)
- Log file from mirroring SMT repositories (optional): `/var/log/smt/smt-mirror.log`

- Other client node log files saved on the Administration Server:
  - /var/log/crowbar/sledgehammer/d*.log: Initial Chef client run on PXE-booted nodes prior to discovery by Crowbar.
  - /var/log/crowbar/chef-client/d*.log: Output from Chef client when proposals are applied to nodes. This is the first place to look if a Barclamp proposal fails to apply.

A.2 On All Other Crowbar Nodes

Logs for when the node registers with the Administration Server:

- /var/log/crowbar/crowbar_join/errlog
- /var/log/crowbar/crowbar_join/$TOPIC.{log,err}: STDOUT/STDERR from running commands associated with $TOPIC when the node joins the Crowbar cluster. $TOPIC can be:
  - zypper: package management activity
  - ifup: network configuration activity
  - Chef: Chef client activity
  - time: starting of ntp client
- Chef client log: /var/log/chef/client.log
- Default SUSE log files: /var/log/messages, /var/log/zypper.log etc.

A.3 On the Control Node

- /var/log/keystone/keystone.log: OpenStack authentication, etc.
- /var/log/rabbitmq/*: logs for RabbitMQ, used by OpenStack for handling message queues

- `/var/log/nova/`: various logs relating to Nova services:
    - `api.log`
    - `consoleauth.log`
    - `network.log`
    - `nova-manage.log`
    - `scheduler.log`
    - `volume.log`
- `/var/log/apache2/openstack-dashboard-*`: Logs for the Nova Dashboard

A.4 On Compute Nodes

`/var/log/nova/`: various logs relating to Nova services:

- `compute.log`
- `nova-manage.log`

A.5 On Nodes with Ceph Barclamp

`/var/log/ceph/*.log`

The Network Barclamp Template File

B

The Crowbar network Barclamp provides two functions for the system. The first is a common role to instantiate network interfaces on the crowbar managed systems. The other function is address pool management. While the addresses can be managed with the YaST Crowbar module, complex network setups require to manually edit the network Barclamp template file `/etc/crowbar/network.json`. This section explains the file in detail. Settings in this file are applied to all nodes in SUSE Cloud.

WARNING: No Network Changes After Having Run the Cloud Installation Script

Once you have run the SUSE Cloud installation script, you cannot change the network setup anymore. If doing so, you would have to completely set up the Administration Server again.

The only exception from this rule is the interface map. This section can be changed at a later stage as well. See Section B.3, “Interface Map” (page 107) for details.

B.1 Editing network.json

The `network.json` is located in `/etc/crowbar/`. To edit it, open it in an editor of your choice. The template has the following general structure:

```
{
   "attributes" : {
      "mode" : "value",
      "start_up_delay" : value,
      "teaming" : { "mode": value },❶
      "network" : {
         "interface_map"❷ : [
            ...
         ],
         "conduit_map"❸ : [
            ...
         ],
         "networks"❹ : {
            ...
         },
      }
   }
}
```

❶ General attributes. Refer to Section B.2, “Global Attributes” (page 106) for details.

❷ Interface map section. Defines the order in which the physical network interfaces are to be used. Refer to Section B.3, “Interface Map” (page 107) for details.

❸ Network conduit section defining the network modes and the network interface usage. Refer to Section B.4, “Network Conduits” (page 109) for details.

❹ Network definition section. Refer to Section B.5, “Network Definitions” (page 114) for details.

NOTE: Order of Elements

The order in which the entries in the `network.json` file appear, may differ from the one listed above. Use your editor's search function to find certain entries.

B.2 Global Attributes

The most important options to define in the global attributes section are the default values for the network and bonding modes. The following global attributes exist:

```
"start_up_delay" : 30,❶
         "mode": "single",❷
         "teaming" : { "mode" : 5 },❸
```

❶ Time (in seconds) the Chef-client waits for the network interfaces to become online before running into a time-out.

❷ Network mode. Defines the configuration name (or name space) to be used from the conduit_map (see Section B.4, “Network Conduits” (page 109)). This allows to define multiple configurations (single, dual, and team are preconfigured) and switch them by just changing this parameter.

❸ Default bonding mode. See https://www.kernel.org/doc/Documentation/networking/bonding.txt for a list of available modes.

WARNING: Bonding Mode 6 (balance-alb) not supported

Adaptive load balancing (balance-alb or 6) is not supported because of problems with bridges and openvswitch.

B.3 Interface Map

By default physical network interfaces are used by the order they appear under /sys/class/net/. In case you would like to apply a different order, you need to create an interface map where you can specify a custom order of the bus IDs. Interface maps are created for specific hardware configurations and are applied to all machines matching this configuration.

```
{
   "pattern" : "PowerEdge R610"❶,
   "serial_number" : "0x02159F8E"❷,
   "bus_order" : [❸
      "0000:00/0000:00:01",
      "0000:00/0000:00:03"
   ]
}
```

❶ Hardware specific identifier. This identifier can be obtained by running the command dmidecode -s system-product-name on the machine you want to identify. You can login to a nodes during the hardware discovery phase (when booting the SLEShammer image) via the Administration Server.

❷ Additional hardware specific identifier. This identifier can be used in case two machines have the same value for *pattern*, but different interface maps are needed. Specifying this parameter is optional (it's not included in the default network.json file). The serial number of a machine can be obtained by running the command dmidecode -s system-serial-number on the machine you want to identify.

❸ Bus IDs of the interfaces. The order in which they are listed here defines the order in which Chef addresses the interfaces. The IDs can be obtained by listing the contents of /sys/class/net/.

IMPORTANT: PXE Boot Interface must be Listed First

The physical interface used to PXE boot the node must always be listed first!

NOTE: Interface Map Changes Allowed after Having Run the SUSE Cloud Installation Script

Contrary to all other sections in `network.json`, you can change interface maps after having executed the SUSE Cloud installation script. However, nodes that are already deployed and affected by these changes, need to be deployed again. Therefore it is not recommended to make changes to the interface map that affects active nodes.

If you change the interface mappings after having run the SUSE Cloud installation script, you *must not* make your changes by editing `network.json`. You must rather use the Crowbar Web interface and open *Barclamps* > *Crowbar* > *Network* > *Edit*. Activate your changes by clicking *Apply*.

B.3.1 Interface Map Example

Example B.1: *Changing the Network Interface Order on a Machine with four NICs*

1. Get the machine identifier by running the following command on the machine to which the map should be applied:

   ```
   ~ # dmidecode -s system-product-name
   AS 2003R
   ```

   The resulting string needs to be entered on the *pattern* line of the map. It is interpreted as a Ruby regular expression (see `http://www.ruby-doc.org/core-2.0/Regexp.html` for a reference). Unless the pattern starts with `^` and ends with `$` a substring match is performed against the name return from the above commands.

2. List the interface devices in `/sys/class/net` to get the current order and the bus ID of each interface:

   ```
   ~ # ls -lgG /sys/class/net/ | grep eth
   lrwxrwxrwx 1 0 Jun 19 08:43 eth0 -> ../../devices/pci0000:00/0000:00:1c.0/0000:09:00.0/net/eth0
   lrwxrwxrwx 1 0 Jun 19 08:43 eth1 -> ../../devices/pci0000:00/0000:00:1c.0/0000:09:00.1/net/eth1
   lrwxrwxrwx 1 0 Jun 19 08:43 eth2 -> ../../devices/pci0000:00/0000:00:1c.0/0000:09:00.2/net/eth2
   lrwxrwxrwx 1 0 Jun 19 08:43 eth3 -> ../../devices/pci0000:00/0000:00:1c.0/0000:09:00.3/net/eth3
   ```

The bus id is included in the path of the link target—it's the following string: `../../devices/pciBUS ID/net/eth0`

3. Create an interface map with the bus id listed in the order the interfaces should be used. Keep in mind that the interface from which the node is PXE booted must be listed first. In the following example the default interface order has been changed to `eth0`, `eth2`, `eth1` and `eth3`.

```
{
   "pattern" : "AS 2003R",
   "bus_order" : [
      "0000:00/0000:00:1c.0/0000:09:00.0",
      "0000:00/0000:00:1c.0/0000:09:00.2",
      "0000:00/0000:00:1c.0/0000:09:00.1",
      "0000:00/0000:00:1c.0/0000:09:00.3"
   ]
}
```

B.4 Network Conduits

Network conduits define mappings for logical interfaces—one or more physical interfaces bonded together. Each conduit can be identified by a unique name, the *pattern*. This pattern is also referred to as “Network Mode” in this document.

Several network modes are already pre-defined. The most important ones are:

single: Only use the first interface for all networks. VLANs will be added on top of this single interface.
dual: Use the first interface as the admin interface and the second one for all other networks. VLANs will be added on top of the second interface.
team: Bond first two interfaces. VLANs will be added on top of the bond.

See Section 2.1.2, “Network Modes” (page 13) for detailed descriptions. Apart from these modes a fallback mode `".*/.*/.*"` is also pre-defined—it is applied in case no other mode matches the one specified in the global attributes section. These modes can be adjusted according to your needs. It is also possible to define a custom mode.

The mode name that is specified with `mode` in the global attributes section is deployed on all nodes in SUSE Cloud. It is not possible to use a different mode for a certain node. However, you can define “sub” modes with the same name that only match machines with a certain number of physical network interfaces or machines with certain roles (all Compute Nodes for example).

```
"conduit_map" : [
...
   {
      "pattern" : "team/.*/.*"❶,
      "conduit_list" : {
         "intf2"❷ : {
            "if_list" : ["1g1","1g2"]❸,
            "team_mode" : 5❹
         },
         "intf1" : {
            "if_list" : ["1g1","1g2"],
            "team_mode" : 5
         },
         "intf0" : {
            "if_list" : ["1g1","1g2"],
            "team_mode" : 5
         }
      }
   },
...
],
```

❶ This line contains the pattern definition for a mode. The value for pattern must have the following form:

mode_name/number_of_nics/node_role

It is interpreted as a Ruby regular expression (see http://www.ruby-doc.org/core-2.0/Regexp.html for a reference).

mode_name

Name of the network mode. This string is used to reference the mode from the general attributes section.

number_of_nics

Normally it is not possible to apply different network modes to different roles—you can only specify one mode in the global attributes section. However, it does not make sense to apply a network mode that bonds three interfaces on a machine with only two physical network interfaces. This option enables you to create modes for nodes with a given number of interfaces.

`node_role`

This part of the pattern let's you create matches for a certain node role. This enables you to create network modes for certain roles, for example the Compute Nodes (role: *nova-multi-compute*) or the Swift nodes (role: *swift-storage*). See Example B.3, “Network Modes for Certain Roles” (page 113) for the full list of roles.

❷ The logical network interface definition. Each conduit list must contain at least one such definition. This lines defines the name of the logical interface. This identifier must be unique and will also be referenced in the network definition section. It is recommended to stick with the pre-defined naming scheme with `intf0` for “Interface 0”, `intf1` for “Interface 1”, etc.. If you change the name (not recommended), you also need to change all references in the network definition section.

❸ This lines maps one or more *physical* interfaces to the logical interface. Each entry represents a physical interface. If more than one entry exist, the interfaces are bonded—either with the mode defined in the *team_mode* attribute of this conduit section or, if that is not present by the globally defined *teaming* attribute.

The physical interfaces definition needs to fit the following pattern:

```
[Quantifier][Speed][Interface Number]
```

Valid examples are `+1g2`, `10g1` or `?1g2`.

Quantifier

Specifying the quantifier is optional. The following values may be entered:

`+`: at least the speed specified afterwards (specified value or higher)
`-`: at most the speed specified afterwards (specified value or lower)
`?`: any speed (speed specified afterwards is ignored)

If no quantifier is specified, the exact speed specified is used.

Speed

Specifying the interface speed is mandatory (even if using the `?` quantifier). The following values may be entered:

`10m`: 10 Mbit
`100m`: 100 Mbit
`1g`: 1 Gbit

10g: 10 Gbit

Order

Position in the interface order. Specifying this value is mandatory. The interface order is defined by the order in which the interfaces appear in /sys/class/net (default) or, if existing, by an interface map. The order is also linked to the speed in this context, so 1g1 means “The first 1Gbit interface”, +1g1 means “The first 1Gbit or 10Gbit interface”. ?1g1 would match the very first interface, regardless of its speed.

NOTE: Ordering Numbers

Ordering numbers start with 1 rather than with 0.

❹ The bonding mode to be used for this logical interface. Overwrites the default set in the global attributes section *for this interface*. See https://www.kernel.org/doc/Documentation/networking/bonding.txt for a list of available modes. Specifying this option is optional—if not specified here, the global setting applies.

B.4.1 Network Conduit Examples

Example B.2: *Network Modes for Different NIC Numbers*

The following example defines a network mode named my_mode for nodes with 6, 3 and an arbitrary number of network interfaces. Since the first mode that matches is applied, it is important that the specific modes (for 6 and 3 NICs) are listed before the general one:

```
{
   "pattern" : "my_mode/6/.*",
   "conduit_list" : { ... }
},
{
   "pattern" : "my_mode/3/.*",
   "conduit_list" : { ... }
},
{
   "pattern" : "my_mode/.*/.*",
   "conduit_list" : { ... }
},
```

Example B.3: *Network Modes for Certain Roles*

The following example defines network modes for Compute Nodes with four physical interfaces, the Administration Server (role `crowbar`), the Control Node, and a general mode applying to all other nodes.

```
{
   "pattern" : "my_mode/4/nova-multi-compute",
   "conduit_list" : { ... }
},
{
   "pattern" : "my_mode/.*/crowbar",
   "conduit_list" : { ... }
},
{
   "pattern" : "my_mode/.*/nova-multi-controller",
   "conduit_list" : { ... }
},
{
   "pattern" : "my_mode/.*/.*",
   "conduit_list" : { ... }
},
```

The following values for `node_role` can be used:

```
ceph-mon-master
ceph-mon
ceph-store
cinder-controller
cinder-volume
crowbar
database-server
glance-server
keystone-server
nova-multi-compute
nova-multi-controller
nova_dashboard-server
quantum-server
rabbitmq-server
swift-dispersion
swift-proxy
swift-ring-compute
swift-storage
```

The role `crowbar` refers to the Administration Server.

Example B.4: *Network Modes for Certain Machines*

Apart from the roles listed under Example B.3, “Network Modes for Certain Roles” (page 113) each individual node in SUSE Cloud has a unique role, which lets you create modes matching exactly one node. The role is named after the scheme `crowbar-d`*`FULLY QUALIFIED HOSTNAME`*`. The` *`FULLY QUALIFIED HOSTNAME`* in turn is composed of the MAC address of the network interface used to PXE boot the node and the domain name configured on the Administration Server. Colons and periods are replaced with underscores. An example role name would be: `crowbar-d1a-12-05-1e-35-49_my_cloud`.

Network mode definitions for certain machines must be listed first in the conduit map to make sure no other, general rules which would also map, are applied.

```
{
   "pattern" : "my_mode/.*/crowbar-d1a-12-05-1e-35-49_my_cloud",
   "conduit_list" : { ... }
},
```

B.5 Network Definitions

The network definitions contain IP address assignments, the bridge and VLAN setup and settings for the router preference. Each network is also assigned to a logical interface defined in the network conduit section. In the following the network definition is explained using the example of the admin network definition:

```
"admin" : {
   "conduit" : "intf0"❶,
   "add_bridge" : false❷,
   "use_vlan" : false❸,
   "vlan" : 100❹,
   "router_pref" : 10❺,
   "subnet" : "192.168.124.0"❻,
   "netmask" : "255.255.255.0",
   "router" : "192.168.124.1",
   "broadcast" : "192.168.124.255",
   "ranges" : {
      "admin" : { "start" : "192.168.124.10",
                  "end" : "192.168.124.11" },
      "switch" : { "start" : "192.168.124.241",
                    "end" : "192.168.124.250" },
      "dhcp" : { "start" : "192.168.124.21",
                 "end" : "192.168.124.80" },
      "host" : { "start" : "192.168.124.81",
                 "end" : "192.168.124.160" }
   }
},
```

❶ Logical interface assignment. The interface must be defined in the network conduit section and must be part of the active network mode.

❷ Bridge setup. Do not touch. Should be `false` for all networks.

❸ Create a VLAN for this network. Changing this setting is not recommended.

❹ ID of the VLAN. Change this to the VLAN ID you intend to use for the specific network if required. This setting can also be changed using the YaST Crowbar interface.

❺ Router preference, used to set the default route. On nodes hosting multiple networks the router with the lowest `router_pref` becomes the default gateway. Changing this setting is not recommended.

❻ Network address assignments. These values can also be changed by using the YaST Crowbar interface.

Setting up a Netboot Environment for Microsoft* Windows

C

Setting up Compute Nodes running Microsoft Hyper-V Server or Windows Server 2012 requires to configure the Administration Server to be able to provide the netboot environment for node installation. The environment is generated from a machine running Microsoft Hyper-V Server or Microsoft Windows Server 2012.

C.1 Requirements

The following requirements must be met in order to successfully deploy Hyper-V:

1. Provide a separate machine running Microsoft Windows Server 2012 or Microsoft Hyper-V Server. The machine must be able to access the Administration Server and the Internet.

2. Install Samba (package `samba` and the Microsoft Hyper-V tools (package `hyper-v`) on the Administration Server.

3. and the Microsoft Hyper-V tools on the Administration Server.

C.2 Providing a Hyper-V Netboot Environment

In order to provide a Hyper-V netboot environment on the Administration Server, a samba share, that can be mounted on the Windows machine, is created on the Administration Server. Among others, this share contains the Hyper-V tools which provide Windows scripts to generate the environment.

Procedure C.1: *Preparing the Administration Server*

1 Ensure that the requirements listed at Section C.1, "Requirements" (page 117) are met.

2 Make sure the Samba daemons `smb` and `nmb` are automatically started during boot and are currently running by executing the following commands:

```
insserv smb nmb
rcsmb start
rcnmb start
```

3 Edit the Samba configuration file `/etc/samba/smb.conf` and add the following share:

```
[reminst]
        comment = MS Windows remote install
        guest ok = Yes
        inherit acls = Yes
        path = /srv/tftpboot
        read only = No
        force user = root
```

By default, the workgroup name is set to `WORKGROUP` in the `[global]` section of `/etc/samba/smb.conf`. Adjust it, if needed.

4 Reload the smb service:

```
rcsmb reload
```

Once Samba is properly configured on the Administration Server, log in to the machine running Microsoft Windows Server 2012 or Microsoft Hyper-V Server and generate the environment:

Procedure C.2: *Netboot Environment Generation*

1. Log in to the Microsoft Windows Server 2012 or Microsoft Hyper-V Server machine. Connect the device name `X:` to the Samba share `\\crowbar\reminst` configured on the Administration Server (which is always named `crowbar`) in the previous step. Use the login credentials of the Administration Server's `root` for the connection. This can either be done from the Explorer or on the command line with the `net use` command.

2. Device `X:` contains a directory `X:\adk-tools` with image creation scripts for either Hyper-V (`build_winpe_hyperv-6.2.ps1`) or Windows Server (`build_winpe_windows-6.2.ps1`). Build the image by running the following commands on the command line:

   Hyper-V Server

   ```
   powershell Set-ExecutionPolicy -ExecutionPolicy Bypass
   powershell x:\adk-tools\build_winpe_hyperv-6.2.ps1
   ```

   Windows Server 2012

   ```
   powershell Set-ExecutionPolicy -ExecutionPolicy Bypass
   powershell x:\adk-tools\build_winpe_windows-6.2.ps1
   ```

   Executing the script requires Internet access, because additional software needs to be downloaded.

Once the netboot environment is set up, you can choose either *Windows Server 2012* or *Hyper-V Server 2012* as the *Target Platform* for newly discovered nodes from the *Node Dashboard*.

VMware ESX Installation Instructions

D

VMware ESX is not supported “natively” by SUSE Cloud—it rather delegates requests to an existing vCenter. It requires preparations at the vCenter and post install adjustments of the Compute Node.

D.1 Requirements

The following requirements must be met in order to successfully deploy esxi:

1. VMware vSphere vCenter 5.1

2. VMware vSphere ESXi nodes 5.1

3. Does not work with Neutron OpenVSwitch with *gre* tunnels.

D.2 Preparing the VMware vCenter Server

SUSE Cloud requires the VMware vCenter server to run version 5.1 or better. You need to create a datacenter for SUSE Cloud:

1 Log into the vCenter Server using the vSphere Web Client

2 Choose *Hosts and Clusters* and create a *Datacenter*

3. Set up a *New Cluster* which has *DRS* enabled.

4. Set *Automation Level* to `Fully Automated` and *Migration Threshold* to `Aggressive`.

5. Add ESXi hosts to the cluster.

D.3 Finishing the esxi Compute Node Installation

Deploy Nova as described in Section 5.10, “Deploying Nova” (page 89) on a single Compute Node. Set *VMware Support* to *true* and fill in the following attributes:

vCenter Host Address
: IP address of the vCenter server.

vCenter Username / *vCenter Password*
: vCenter login credentials.

Cluster Name
: Name of the cluster you have added ob the vCenter server.

Datastore Name
: Regular Expression matching the name of the datastore.

Terminology

Administration Server

Also called Crowbar Administration Node. Manages all other nodes. It assigns IP addresses to them, PXE boots them, configures them, and provides them the necessary software for their roles. To provide these services, the Administration Server runs Crowbar, Chef, DHCP, TFTP, NTP, and other services.

AMI (Amazon Machine Image)

A virtual machine that can be created and customized by a user. AMIs can be identified by an ID prefixed with `ami-`.

Availability Zone

An OpenStack method of partitioning clouds. It enables you to arrange OpenStack Compute hosts into logical groups, which typically have physical isolation and redundancy from other availability zones, for example, by using separate power supply or network equipment for each zone. When users provision resources, they can specify from which availability zone their instance should be created. This allows cloud consumers to ensure that their application resources are spread across disparate machines to achieve high availability in the event of hardware failure. Since the Grizzly release, availability zones are implemented via host aggregates.

AWS (Amazon Web Services)

A collection of remote computing services (including Amazon EC2, Amazon S3, and others) that together make up Amazon's cloud computing platform.

Barclamp

A set of Chef cookbooks, templates, and other logic. Used to apply a particular role to individual nodes or a set of nodes.

Ceilometer

Code name for OpenStack Metering (page 129).

Cell

Cells provide a new way to scale Compute deployments, including the ability to have compute clusters (cells) in different geographic locations all under the same Compute API. This allows for a single API server being used to control access to multiple cloud installations. Cells provide logical partitioning of Compute resources in a child/parent relationship.

Chef

An automated configuration management platform for deployment of your entire cloud infrastructure. The Chef server manages many of the software packages and allows the easy changing of nodes.

Ceph

A massively scalable, open source, distributed storage system. It consists of an object store, a block store, and a POSIX-compliant distributed file system.

Cinder

Code name for OpenStack Block Storage (page 128).

Container

A container is a storage compartment for data. It can be thought of as a directory, only that it cannot be nested.

cloud-init

A package commonly installed in virtual machine images. It uses the SSH public key to initialize an instance after boot.

Compute Node

Node within a SUSE Cloud. A physical server running a Hypervisor. A Compute Node is a host for guest virtual machines that are deployed in the cloud. It starts virtual machines on demand using `nova-compute`. To split virtual machine load across more than one server, a cloud should contain multiple Compute Nodes.

Control Node

Node within a SUSE Cloud. The Control Node is configured through the Administration Server and registers with the Administration Server for all required software. Hosts the OpenStack API endpoints and the OpenStack scheduler and runs the `nova` services—except for `nova-compute`, which is run on the Compute Nodes. The Control Node coordinates everything about cloud virtual machines: like a central communication center it receives all requests (for example, if a user wants to start or stop a virtual machine) and communicates with the Compute Nodes to coordinate fulfillment of the request. A cloud can contain multiple Control Nodes.

Cookbook

A collection of Chef recipes which deploy a software stack or functionality. The unit of distribution for Chef.

Crowbar

Bare-metal installer and an extension of Chef server. The primary function of Crowbar is to get new hardware into a state where it can be managed by Chef. That means: Setting up BIOS and RAID, network, installing a basic operating system, and setting up services like DNS, NTP, and DHCP. The Crowbar server manages all nodes, supplying configuration of hardware and software.

EBS (Amazon Elastic Block Store)

Block-level storage volumes for use with Amazon EC2 instances. Similar to OpenStack Cinder.

EC2 (Amazon Elastic Compute Cloud)

A public cloud run by Amazon. It provides similar functionality to OpenStack Compute.

Ephemeral Disk

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

Flavor

The compute, memory, and storage capacity of `nova` computing instances (in terms of virtual CPUs, RAM, etc.). Flavors can be thought of as “templates” for the amount of cloud resources that are assigned to an instance.

Floating IP Address

An IP address that a Compute project can associate with a virtual machine. A pool of floating IPs is available in OpenStack Compute, as configured by the cloud operator. After a floating IP address has been assigned to an instance, the instance can be reached from outside the cloud by this public IP address. Floating IP addresses can be dynamically disassociated and associated with other instances.

Fixed IP Address

When an instance is launched, it is automatically assigned a fixed (private) IP address, which stays the same until the instance is explicitly terminated. Private IP addresses are used for communication between instances.

Glance

Code name for OpenStack Image (page 129).

Guest Operating System

An instance of an operating system installed on a virtual machine.

Heat

Code name for OpenStack Orchestration (page 130).

Horizon

Code name for OpenStack Dashboard (page 129).

Host

A physical computer.

Host Aggregate

An OpenStack method of grouping hosts via a common set of metadata. It enables you to tag groups of hosts with certain capabilities or characteristics. A characteristic could be related to physical location, allowing creation or further partitioning of availability zones, but could also be related to performance (for example, indicating the availability of SSD storage) or anything else which the cloud administrators deem appropriate. A host can be in more than one host aggregate.

Hybrid Cloud

One of several deployment models for a cloud infrastructure. A composition of both public and private clouds that remain unique entities, but are bound together by standardized technology for enabling data and application portability. Integrating SUSE Studio and SUSE Manager with SUSE Cloud delivers a platform and tools with which to enable enterprise hybrid clouds.

Hypervisor

A piece of computer software, firmware or hardware that creates and runs virtual machines. It arbitrates and controls access of the virtual machines to the underlying hardware.

IaaS (Infrastructure-as-a-Service)

A service model of cloud computing where processing, storage, networks, and other fundamental computing resources are rented over the Internet. It allows the customer to deploy and run arbitrary software, including operating systems and applications. The customer has control over operating systems, storage, and deployed applications but does not control the underlying cloud infrastructure. Housing and maintaining it is in the responsibility of the service provider.

Image

A file that contains a complete Linux virtual machine.

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.

Instance

A virtual machine that runs inside the cloud.

Instance Snapshot

A point-in-time copy of an instance. It preserves the disk state of a running instance and can be used to launch a new instance or to create a new image based upon the snapshot.

Keypair

OpenStack Compute injects SSH keypair credentials that are injected into images when they are launched.

Keystone

Code name for OpenStack Identity (page 129).

libvirt

Virtualization API library. Used by OpenStack to interact with many of its supported hypervisors.

Linux Bridge

A software allowing multiple virtual machines to share a single physical NIC within OpenStack Compute. It behaves like a hub: You can connect multiple (physical or virtual) network interface devices to it. Any ethernet frames that come in from one interface attached to the bridge is transmitted to all other devices.

Logical Volume (LV)

Acts as a virtual disk partition. After creating a Volume Group (VG) (page 134), logical volumes can be created in that volume group. Logical volumes can be used as raw block devices, swap devices, or for creating a (mountable) file system just like disk partitions.

Migration

The process of moving a virtual machine instance from one Compute Node to another. This process can only be executed by cloud administrators.

Network

In the OpenStack Networking API: An isolated L2 network segment (similar to a VLAN). It forms the basis for describing the L2 network topology in a given OpenStack Networking deployment.

Neutron

Code name for OpenStack Networking (page 129).

Node

A (physical) server that is managed by Crowbar.

Nova

Code name for OpenStack Compute (page 129).

Object

Basic storage entity in OpenStack Object Storage, representing a file that your store there. When you upload data to OpenStack Object Storage, the data is neither compressed nor encrypted, it is stored as-is.

Open vBridge

A virtual networking device. It behaves like a virtual switch: network interface devices connect to its ports. The ports can be configured similar to a physical switch's port, including VLAN configurations.

OpenStack

A collection of open source software to build and manage public and private clouds. Its components are designed to work together to provide Infrastructure as a Service and massively scalable cloud computing software.

At the same time, OpenStack is also a community and a project.

OpenStack Block Storage

One of the core OpenStack components and services (code name: `Cinder`). It provides persistent block level storage devices for use OpenStack compute instances. The block storage system manages the creation, attaching and detaching of the block devices to servers. Prior to the OpenStack Grizzly release, the service was part of `nova-volume` (block service).

OpenStack Compute

One of the core OpenStack components and services (code name: `Nova`). It is a cloud computing fabric controller and as such, the main part of an IaaS system. It provides virtual machines on demand.

OpenStack Dashboard

One of the core OpenStack components or services (code name: `Horizon`). It provides a modular Web interface for OpenStack services and allows end users and administrators to interact with each OpenStack service through the service's API.

OpenStack Identity

One of the core OpenStack components or services (code name: `Keystone`). It provides authentication and authorization for all OpenStack services.

OpenStack Image

One of the core OpenStack components or services (code name: `Glance`). It provides discovery, registration, and delivery services for virtual disk images.

OpenStack Metering

An OpenStack project (code name: `Ceilometer`), which is the cloud metering component for OpenStack-based clouds. The project aims to provide a unique point of contact across all OpenStack core components for acquiring metrics which can then be consumed by other components such as customer billing. It was incubated during the OpenStack Grizzly cycle and will become integrated for the OpenStack Havana release. In SUSE Cloud 2.0, it is included as a technology preview.

OpenStack Networking

One of the core OpenStack components or services (code name: `Neutron`). It provides “network connectivity as a service” between interface devices (for example, vNICs) managed by other OpenStack services (for example, Compute). Allows users to create their own networks and attach interfaces to them.

OpenStack Object Storage

One of the core OpenStack components or services (code name: `Swift`). Allows to store and retrieve files while providing built-in redundancy and fail-over. Can be used for backing up and archiving data, streaming data to a user's Web browser, or developing new applications with data storage integration.

OpenStack Orchestration
: An OpenStack incubation project (code name: `Heat`). OpenStack Orchestration is a service to orchestrate multiple composite cloud applications using file-based or Web-based templates. It contains both a user interface and an API and describes your cloud deployment in a declarative language. In SUSE Cloud 2.0 it is included as a technology preview.

OpenStack Service
: A collection of Linux services (or daemons) that work together to provide core functionality within the OpenStack project, like storing objects, providing virtual servers, or authentication and authorization. All services have code names, which are also used in configuration files and command line programs that belong to the service.

PaaS (Platform-as-a-Service)
: A service model of cloud computing where a computing platform and cloud-based application development tools are rented over the Internet. The customer controls software deployment and configuration settings, but not the underlying cloud infrastructure including network, servers, operating systems, or storage.

Port
: In the OpenStack Networking API: An attachment port to a L2 OpenStack Networking network.

Project
: A concept in OpenStack Identity. Used to identify a group, an organization, or a project (or more generically, an individual customer environment in the cloud). Also called `tenant`. The term `tenant` is primarily used in the OpenStack command line tools.

Proposal
: Special configuration for a Barclamp. It includes Barclamp-specific settings, and a list of nodes to which the proposal should be applied.

Private Cloud
: One of several deployment models for a cloud infrastructure. The infrastructure is operated exclusively for a single organization and may exist on or off premises. The cloud is owned and managed by the organization itself, by a third party or a combination of both.

Private IP Address

See Fixed IP Address (page 125).

Public Cloud

One of several deployment models for a cloud infrastructure. The cloud infrastructure is designed for use by the general public and exists on the premises of the cloud provider. Services like applications, storage, and other resources are made available to the general public for free or are offered on a pay-per-use model. The infrastructure is owned and managed by a business, academic or government organization, or some combination of these.

Public IP Address

See Floating IP Address (page 125).

qcow (QEMU Copy on Write)

A disk image format supported by the QEMU virtual machine manager. A `qcow2` image helps to optimize disk space as it consumes disk space only when contents are written on it and grows as data is added.

`qcow2` is a more recent version of the `qcow` format where a read-only base image is used, and all writes are stored to the `qcow2` image.

Quota

Restriction of resources to prevent overconsumption within a cloud. In OpenStack, quotas are defined per project and contain multiple parameters, such as amount of RAM, number of instances, or number of floating IP addresses.

RC File (`openrc.sh`)

Environment file needed for the OpenStack command line tools. The RC file is project-specific and contains the credentials used by OpenStack Compute, Image, and Identity services.

Recipe

A group of Chef scripts and templates. Recipes are used by Chef to deploy a unit of functionality.

Region

An OpenStack method of aggregating clouds. Regions are a robust way to share some infrastructure between OpenStack compute installations, while allowing for a high degree of failure tolerance. Regions have a separate API endpoint per installation.

Role

In the Crowbar/Chef context: an instance of a Proposal (page 130) that is active on a node.

In the OpenStack Identity (page 129) context: concept of controlling the actions that a user is allowed to perform.

S3 (Amazon Simple Storage Service)

An object storage by Amazon that can be used to store and retrieve data on the Web. Similar in function to OpenStack Object Storage. It can act as a back-end store for Glance images.

SaaS (Software-as-a-Service)

A service model of cloud computing where applications are hosted by a service provider and made available to customers remotely as a Web-based service.

Security Group

Concept in OpenStack Networking. A security group is a container for security group rules. Security group rules allow to specify the type of traffic and direction (ingress/egress) that is allowed to pass through a port.

Snapshot

See Volume Snapshot (page 134) or Instance Snapshot (page 127).

Storage Node

Node within a SUSE Cloud. Acts as the controller for cloud-based storage. A cloud can contain multiple Storage Nodes.

Subnet

In the OpenStack Networking API: A block of IP addresses and other network configuration (for example, a default gateway, DNS servers) that can be associated with an OpenStack Networking network. Each subnet represents an IPv4 or IPv6 address block. Multiple subnets can be associated with a network, if necessary.

SUSE Cloud Administrator

User role in SUSE Cloud. Manages projects, users, images, flavors, and quotas within SUSE Cloud.

SUSE Cloud Dashboard

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

SUSE Cloud Operator

User role in SUSE Cloud. Installs and deploys SUSE Cloud.

SUSE Cloud User

User role in SUSE Cloud. End-user who launches and manages instances, can create snapshots, and use volumes for persistent storage within SUSE Cloud.

Swift

Code name for OpenStack Object Storage (page 129).

TAP Device

A virtual networking device. A TAP device, such as `vnet0` is how hypervisors such as KVM and Xen implement a virtual network interface card (vNIC). An ethernet frame sent to a TAP device is received by the guest operating system. The tap option connects the network stack of the guest operating system to a TAP network device on the host.

Tenant

See Project (page 130).

Veth Pair

A virtual networking device. The acronym veth stands for virtual ethernet interface. A veth is a pair of virtual network interfaces correctly directly together. An Ethernet frame sent to one end of a veth pair is received by the other end of a veth pair. OpenStack Networking makes use of veth pairs as virtual patch cables to make connections between virtual bridges.

VLAN

A physical method for network virtualization. VLANs allow to create virtual networks across a distributed network so that disparate hosts (on independent networks) appear as if they were part of the same broadcast domain.

VM (Virtual Machine)

An operating system instance that runs on top of a hypervisor. Multiple virtual machines can run on the same physical host at the same time.

vNIC

Virtual network interface card.

Volume

Detachable block storage device. Unlike a SAN, it can only be attached to one instance at a time.

Volume Group (VG)

A virtual disk consisting of aggregated physical volumes. Volume groups can be logically partitioned into logical volumes.

Volume Snapshot

A point-in-time copy of an OpenStack storage volume. Used to back up volumes.

vSwitch (Virtual Switch)

A software that runs on a host or node and provides the features and functions of a hardware-based network switch.

Zone

A logical grouping of Compute services and virtual machine hosts.