CloudStack Installation Guide

For CloudStack Version 2.2.14

Revised May 21, 2012 12:22 PM

Contents

About CloudStack

What Is CloudStack?

CloudStack™ is a software platform that pools computing resources to build public, private, and hybrid Infrastructure as a Service (IaaS) clouds. It enables deployment, management, and configuration of cloud computing environments, regardless of where they are deployed. CloudStack manages the network, storage, and compute nodes that make up a cloud infrastructure.

Extending beyond individual virtual machine images running on commodity hardware, CloudStack provides a turnkey cloud infrastructure software stack for delivering virtual datacenters as a service – delivering all of the essential components used to build, deploy, and manage multi-tier and multi-tenant cloud applications. Both open-source and Premium versions are available, with the open-source version offering nearly identical features.

Who Should Read This

If you are new to CloudStack or you want to learn more about concepts before installing and running CloudStack, read this overview.

If you just want to get started, you can skip to Overview of Installation Steps on page 17.

Typical users are service providers and enterprises. With CloudStack, you can:

- Set up an on-demand, elastic cloud computing service. Service providers can sell self-service virtual machine instances, storage volumes, and networking configurations over the Internet.
- Set up an on-premise private cloud for use by employees. Rather than managing virtual machines in the same way as physical machines, with CloudStack an enterprise can offer self-service virtual machines to users without involving IT departments.

What Can CloudStack Do?

Multiple Hypervisor Support

CloudStack works with a variety of hypervisors. A single cloud deployment can contain multiple hypervisor implementations. You have the complete freedom to choose the right hypervisor for your workload.

CloudStack is designed to work with open source Xen and KVM hypervisors as well as enterprise-grade hypervisors such as VMware vSphere, and Citrix XenServer.

Massively Scalable Infrastructure Management

CloudStack can manage tens of thousands of servers installed in multiple geographically distributed datacenters. The centralized management server scales linearly, eliminating the need for intermediate cluster-level management servers. No single component failure can cause cloud-wide outage. Periodic maintenance of the management server can be performed without affecting the functioning of virtual machines running in the cloud.

Automatic Configuration Management

CloudStack automatically configures each guest virtual machine’s networking and storage settings.

CloudStack internally manages a pool of virtual appliances to support the cloud itself. These appliances offer services such as firewalling, routing, DHCP, VPN access, console proxy, storage access, and storage replication. The extensive use of virtual appliances greatly simplifies the installation, configuration, and on-going management of a cloud deployment.

Graphical User Interface

CloudStack offers an administrator's Web interface, used for provisioning and managing the cloud, as well as an end-user's Web interface, used for running VMs and managing VM templates. The UI can be customized to reflect the desired service provider or enterprise look and feel.

API and Extensibility

CloudStack provides an API that gives programmatic access to all the management features available in the UI. The API is maintained and documented. This API enables the creation of command line tools and new user interfaces to suit particular needs. See the Developer's Guide and API Reference, both available at http://docs.cloud.com/CloudStack_Documentation.

The CloudStack platform pluggable allocation architecture allows the creation of new types of allocators for the selection of storage and Hosts. See the Allocator Implementation Guide (http://docs.cloud.com/CloudStack_Documentation/Allocator_Implementation_Guide).

High Availability

The CloudStack platform has a number of features to increase the availability of the system. The Management Server itself may be deployed in a multi-node installation where the servers are load balanced. MySQL may be configured to use replication to provide for a manual failover in the event of database loss. For the Hosts, the CloudStack platform supports NIC bonding and the use of separate networks for storage as well as iSCSI Multipath.

More Information

- Design Considerations on page 97
- Additional Installation Options on page 118

Deployment Architecture Overview

A CloudStack installation consists of two parts: the Management Server and the cloud infrastructure that it manages. The parts are connected by a network. CloudStack offers several types of networking to choose from.

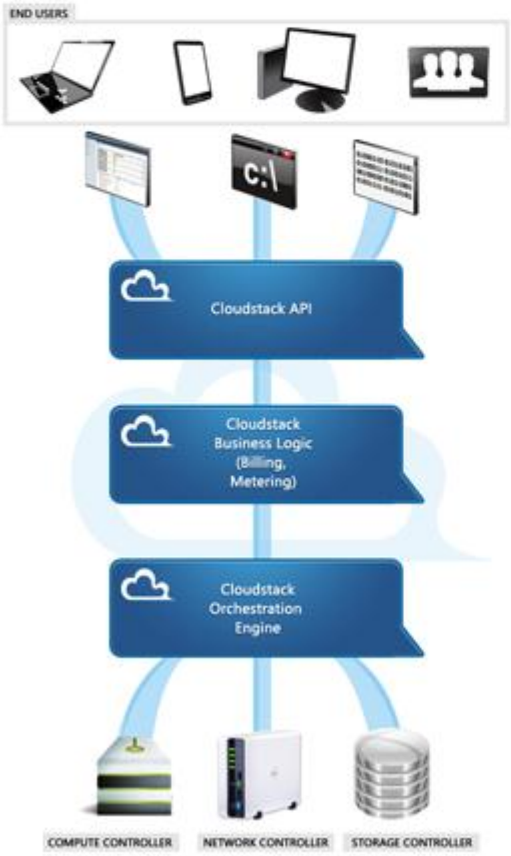
The CloudStack platform includes the Management Server and extensions to industry-standard hypervisor software installed on hosts running in the cloud. When you set up and manage a CloudStack cloud, you provision resources such as hosts, storage devices, and IP addresses into the Management Server, and the Management Server manages those resources. The Management Server provides a Web interface for the cloud administrator to manage the cloud, and another Web interface for users to manage their virtual machines (VMs) running in the cloud.

The minimum installation consists of one Management Server and an infrastructure of one host connected by a basic networking setup. A more full-featured installation consists of a highly-available multi-node Management Server installation and up to thousands of hosts using any of several advanced networking setups.

Management Server

The Management Server:

- Provides the web user interfaces for the administrator and end users.
- Provides the APIs for the CloudStack platform.
- Manages the assignment of guest VMs to particular Hosts.
- Manages the assignment of public and private IP addresses to particular accounts.
- Manages the allocation of storage to guests as virtual disks.
- Manages snapshots, templates, and ISO images, possibly replicating them across data centers.
- Provides a single point of configuration for the cloud.



Cloud Infrastructure

As its name implies, the Management Server is there to manage something -- one or more Zones (typically, datacenters) containing host computers where guest virtual machines will run. The cloud infrastructure is organized as follows:

- **Zone**: Typically, a Zone is equivalent to a single datacenter. A Zone consists of one or more Pods and Secondary Storage.
- **Pod**: A Pod is usually one rack of hardware. It includes a layer-2 switch and one or more clusters.
- **Cluster**: A Cluster consists of one or more Hosts and Secondary Storage.
- **Host**: A single compute node within a cluster.

CloudStack can manage hosts running VMs under a variety of hypervisors:

- Citrix XenServer® 5.6, 5.6 FP1, 5.6 SP2, or 5.6 SP2 + CSP
- VMware vSphere™ 4.1
- KVM (to find an operating system that provides the KVM hypervisor and is supported for use with CloudStack, see KVM Installation and Configuration on page 80)
- Oracle VM (OVM) 2.3
- Bare metal (no hypervisor)

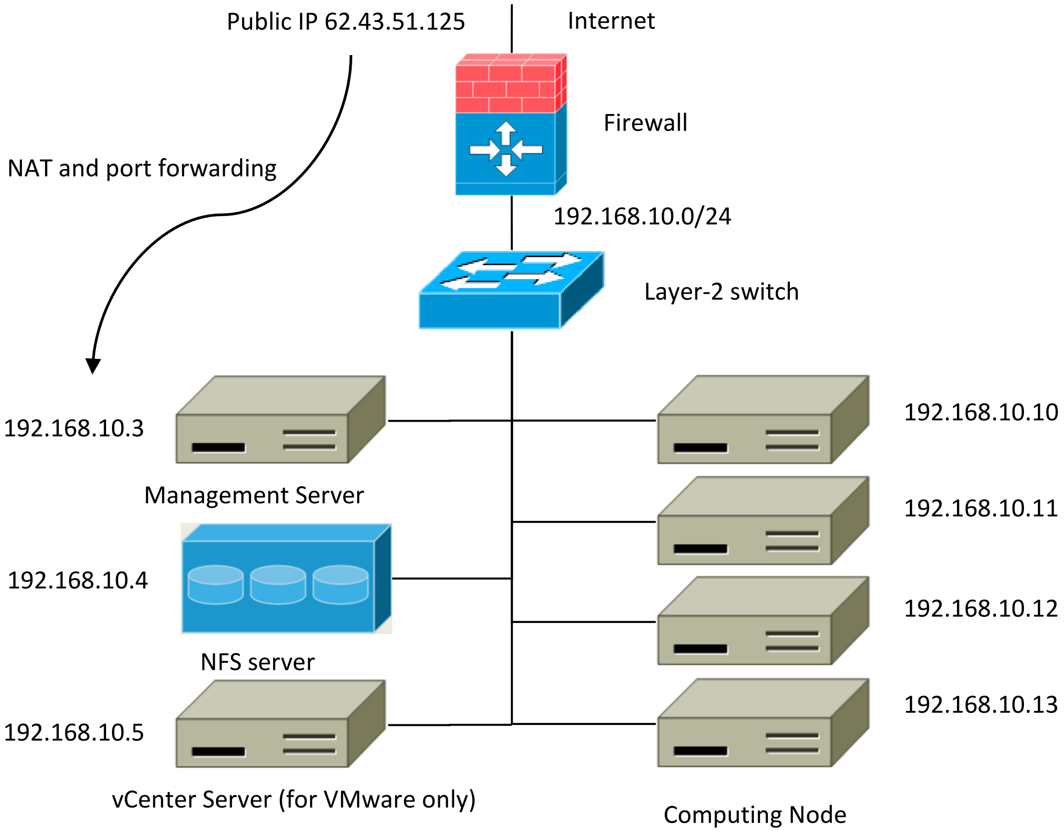
Network

CloudStack offers two types of networking scenario: basic and advanced. The various possible networking choices are described more fully in Network Setup on page 101 and in the Administration Guide.

Example Small-Scale Installation

The minimum installation consists of one Management Server and one or more Hosts. Additional Hosts can be added later. It is also possible to set up multiple Management Servers for high availability.

A Small-Scale CloudStack Installation



Minimum System Requirements

Management Server

This machine will run the Management Server and MySQL database node.

- Operating system: RHEL/CentOS 5.4+ 64-bit or RHEL6 64-bit
- 64-bit x86 CPU (more cores results in better performance)
- 2 GB of memory
- 80 GB of local disk
- At least 1 NIC
- Statically allocated IP address
- Fully qualified domain name as returned by the hostname command

Who Should Read This

If you want to install CloudStack and try it out, or you intend a very small-scale deployment, this section tells enough to make sure you have the hardware you need to get started.

If you're ready for a full-powered CloudStack cloud, you will find additional requirements listed with each component's installation instructions (for example, system requirements for XenServer hosts listed under Installing Citrix XenServer for CloudStack on page 55).

VM Host

As its name implies, the Management Server is there to manage something -- one or more host computers where guest virtual machines will run. For a small-scale setup, you only need one host with the following:

- One of the supported hypervisors (see Cloud Infrastructure on page 14)
- 64-bit x86 CPU (more cores results in better performance)
- Hardware virtualization support required
- 4 GB of memory
- 30 GB of local disk
- At least 1 NIC
- Statically allocated IP Address

Storage

CloudStack needs two types of storage. Primary storage is used for storing the guest VM root disks as well as additional data disk volumes. Secondary storage is used for templates and snapshots,

If desired, you can use localdisk for both primary and secondary storage (as long as your chosen hypervisor supports it).

- Secondary storage needs 100GB minimum capacity.

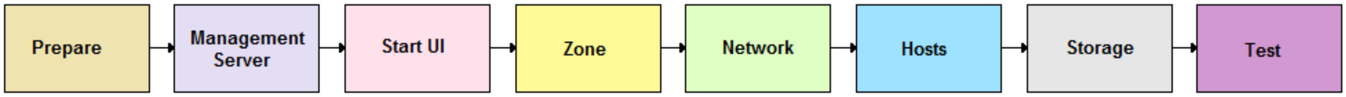
Overview of Installation Steps

These steps give a summary of how to install CloudStack:

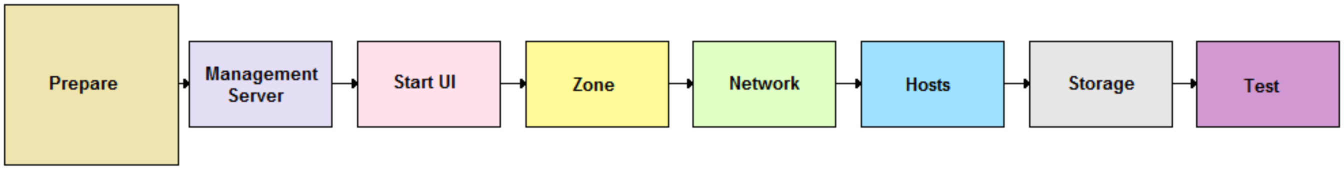
1. Gather information about your physical infrastructure (p. 18)
2. Install the CloudStack Management Server (single-node, p. 21, or multi-node, p. 26)
3. Log in to the CloudStack UI (p. 32)
4. Add a Zone (p. 33)
5. Add physical network to the Zone (p. 37)
6. Add a cluster of hosts (p. 39, p. 43)
7. Set up primary and secondary storage (p. 47, p. 51)
8. Test (p. 53)

For anything more than a simple trial installation, you will first go through a design and preparation phase. For example, there are several networking models that may be used with CloudStack. For more details to help you make those decisions, see:

- Design Considerations, page 97
- Additional Installation Options, page 118



Preparation Checklists



Before you install your chosen hypervisor and the CloudStack software, gather the information in the following checklists. This will make installation go more smoothly.

Management Server Checklist

You can either install a single instance of the CloudStack Management server or multiple Management Servers with a load balancer. You will need the following information for each Management Server.

Installation Requirement	Value	Notes
IP Address		No IPV6 addresses
Netmask		
Gateway		
FQDN		DNS should resolve the FQDN of the Management Server.
Root user		Login id of the root user.
Root password		Password for the root user.
OS	Choose: RHEL 5.4 (or later) or CentOS 5.4 (or later)	Choose one of the supported OS platforms.
ISO Available		CloudStack requires the ISO used for installing the OS in order to install dependent RPMS.

Database Checklist

For a single-node database setup, you will need the following information.

Installation Requirement	Value	Notes
IP Address		Do not use IPV6 addresses.
Netmask		
Gateway		
FQDN		DNS should resolve the FQDN of the Database Server.
Root user		Login id of the root user.
Root password		Password for the root user.
OS	Choose: RHEL 5.4 (or later) or CentOS 5.4 (or later)	Choose one of the supported OS platforms.
ISO Available		CloudStack requires the ISO used for installing the OS in order to install dependent RPMS.
Username for Cloud User in MySQL		Default is cloud.
Password for Cloud user in MySQL		Default is password.

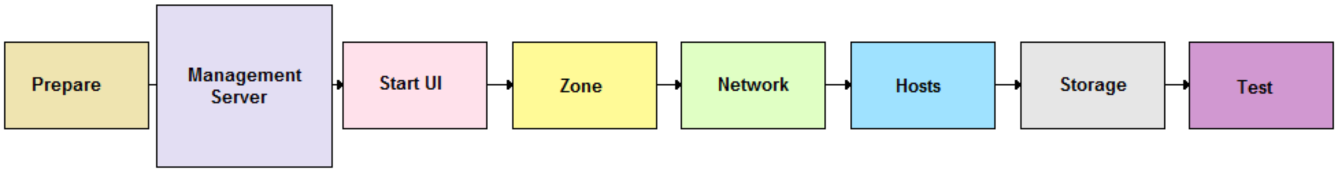
Storage Checklist

CloudStack requires two types of storage: Primary (NFS, iSCSI, local and FC) and Secondary Storage (NFS only). The volumes used for Primary and Secondary storage should be accessible from Management Server and the hypervisors. These volumes should allow root users to read/write data. These volumes must be for the exclusive use of CloudStack and should not contain any data.

You will need the following information when setting up storage.

Installation Requirement	Value	Notes
Type of Storage	Choose: NFS or iSCSI or local	
Storage Server IP Address		
Storage Server Path		
Storage Size		
Secondary Storage Type	NFS	Only NFS is supported.
Secondary Storage IP Address(es)		
Secondary Storage Path		
Secondary Storage Size		
Existing data backed up?		Please back up any data on Primary and Secondary storage volumes, as they may be overwritten by CloudStack.

Management Server Single-Node Installation



This section describes installing a single Management Server and installing MySQL on the same node.

If you prefer to set up Management Server with multiple nodes for high availability, see Management Server Multi-Node Installation on page 26.

System Requirements for Management Server

- Operating system: RHEL/CentOS 5.4+ 64-bit or RHEL6 64-bit
- 64-bit x86 CPU (more cores results in better performance)
- 2 GB of memory
- 80 GB of local disk
- At least 1 NIC
- Statically allocated IP address
- Fully qualified domain name as returned by the hostname command

Management Server Installation Overview

The Management Server download includes everything you need to get started, except MySQL. This includes the Management Server software as well as dependencies.

The procedure for the installation is:

1. Prepare the operating system
2. Prepare NFS shares
3. Prepare the system VM template
4. Install the Management Server
5. Install MySQL

Operating System and OS Preparation

The Management Server requires RHEL/CentOS 5.4 64 bit or later. You can download CentOS 64-bit via the following link: http://isoredirect.centos.org/centos/5/isos/x86_64/. The OS must be prepared to host the Management Server using the following steps.

These steps should be done on all Management Servers.

1. Log in to your OS as root. All the following commands should be run as root.
2. Check for a fully qualified hostname.

```
# hostname --fqdn
```

   This should return a fully qualified hostname such as "kvm1.lab.example.org". If it does not, edit /etc/hosts so that it does.

3. Disable SELinux.

   Set SELinux up to be permissive by default.

   a. First, set the SELINUX variable in /etc/selinux/config to “permissive”. This ensures that the CloudStack Agent can run properly on system reboot.

   b. Then set SELinux to permissive until the system is rebooted:

```
# setenforce permissive
```

4. Make sure that the Management Server can reach the Internet.

```
# ping www.google.com
```

5. (RHEL 6.x) If you do not have a Red Hat Network account, you need to prepare a local Yum repository.

   a. If you are working with a physical host, insert the RHEL 6.x installation CD. If you are using a VM, attach the RHEL6 ISO.

   b. Mount the CDROM to /media.

   c. Create a repo file at /etc/yum.repos.d/rhel6.repo. In the file, insert the following lines:

```
[rhel]
name=rhel6
baseurl=file:///media
enabled=1
gpgcheck=0
```

6. Turn on NTP for time synchronization.

   a. Install NTP.

```
# yum install ntp
```

   b. Edit the NTP configuration file to point to your NTP server.

```
# vi /etc/ntp.conf
```

TIP

NTP is required to synchronize the clocks of the servers in your cloud.

c. Restart the NTP client.

```
# service ntpd restart
```

d. Make sure NTP will start again upon reboot.

```
# chkconfig ntpd on
```

Prepare NFS Shares

CloudStack needs a place to keep primary and secondary storage. Both of these can be NFS shares. This section tells how to set up the NFS shares before adding the storage to CloudStack.

For primary storage, you can use iSCSI instead; for details, see Add Primary Storage on page 47.

1. On the storage server, create an NFS share for secondary storage. Export it with “rw,no_root_squash.” The exact commands to use vary depending on your system.
2. On the management server, create a mount point. For example:

```
# mkdir /mnt/secondary
```

3. Mount the secondary storage on your Management Server. Replace the example NFS server name and NFS share paths below with your own.

```
# mount -t nfs nfsservername:/nfs/share/secondary /mnt/secondary
```

4. If you are using NFS for primary storage as well, repeat these steps with a different NFS share and mount point.

Install the Management Server (Single Node)

This section describes the procedure for performing a single node install where the Management Server and MySQL are on a single, shared OS instance. If you have multiple Management Servers or if you want to have MySQL on a separate server, see Management Server Multi-Node Install on page 26.

1. Install the CloudStack packages. You should have a file in the form of “CloudStack-NNNN.tar.gz”. Untar the file and then run the install.sh script inside it. Replace the file and directory names below with those you are using:

```
# tar xzf CloudStack-VERSION-1-rhel6.2.tar.gz
# cd CloudStack-VERSION-1-rhel6.2
# ./install.sh
Setting up the temporary repository...
Cleaning Yum cache...
Loaded plugins: fastestmirror
11 metadata files removed
Welcome to the CloudStack Installer.  What would you like to do?

    M) Install the Management Server
    A) Install the Agent
    B) Install BareMetal Agent
    S) Install the Usage Monitor
    D) Install the database server
    Q) Quit

    > M
```

2. Choose “M” to install the Management Server software.

Single Node Database Install

1. Re-run install.sh and choose “D” to install MySQL.

```
# ./install.sh
Setting up the temporary repository...
Cleaning Yum cache...
Loaded plugins: fastestmirror
11 metadata files removed
Welcome to the CloudStack Installer.  What would you like to do?

    A) Install the Agent
    B) Install BareMetal Agent
    S) Install the Usage Monitor
    D) Install the database server
    U) Upgrade the CloudStack packages installed on this computer
    R) Stop any running CloudStack services and remove the CloudStack packages from this
computer
    Q) Quit

    > D
```

2. Edit the MySQL configuration (/etc/my.cnf or /etc/mysql/my.cnf, depending on your OS) and insert the following lines in the [mysqld] section. You can put these lines below the datadir line. The max_connections parameter should be set to 350 multiplied by the number of Management Servers you are deploying. This example assumes 1 Management Server.

```
innodb_rollback_on_timeout=1
innodb_lock_wait_timeout=600
max_connections=350
log-bin=mysql-bin
binlog-format = 'ROW'
```

NOTE: The binlog-format variable is supported in MySQL versions 5.1 and greater. It is not supported in MySQL 5.0. In some versions of MySQL, an underscore character is used in place of the hyphen in the variable name. For the exact syntax and spelling of each variable, consult the documentation for your version of MySQL.

3. Restart the MySQL service:

```
# service mysqld restart
```

4. Best Practice: On RHEL and CentOS, MySQL does not set a root password by default. It is very strongly recommended that you set a root password as a security precaution. Run the following commands, and substitute your own desired root password for <password>.

```
# service mysqld start
# mysql -u root
```

```
mysql> SET PASSWORD = PASSWORD('<password>');
```

5. Set up the database. The following command creates the cloud user on the database.

   - In dbpassword, specify the password to be assigned to the cloud user. You can choose to provide no password.

   - In deploy-as, specify the username and password of the user deploying the database. In the following command, it is assumed the root user is deploying the database and creating the cloud user.

   ```
   # cloud-setup-databases cloud:<dbpassword>@localhost --deploy-as=root:<password>
   ```

6. Configure the OS for the Management Server using the following command. This command will set up iptables, sudoers, and start the Management Server.

   ```
   # cloud-setup-management
   ```

Prepare the System VM Template

Secondary storage must be seeded with a template that is used for CloudStack system VMs.

1. On the Management Server, run the following command to retrieve and decompress the system VM template. Run the command for each hypervisor type that you expect end users to run in this Zone.

   This process will require approximately 5 GB of free space on the local file system and up to 30 minutes each time it runs.

   - For vSphere:

   ```
   # /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
   /mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.ova -h vmware -F
   ```

   - For KVM:

   ```
   # /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
   /mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.qcow2.bz2 -h kvm -F
   ```

   - For XenServer:

   ```
   # /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
   /mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.vhd.bz2 -h xenserver
   -F
   ```

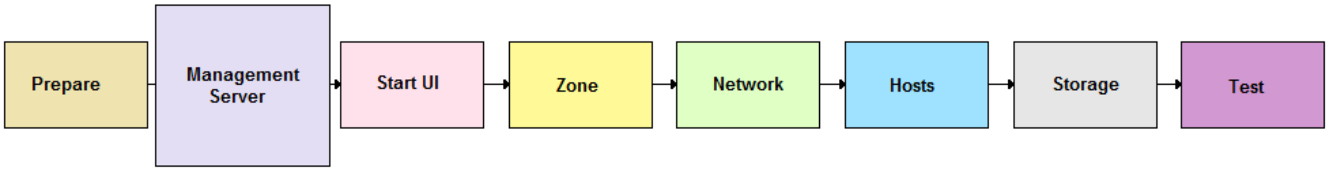
2. When the script has finished, unmount secondary storage and remove the created directory.

   ```
   # umount /mnt/secondary
   # rmdir /mnt/secondary
   ```

3. Repeat these steps for each secondary storage server.

This completes the single node install for the Management Server and database.

Management Server Multi-Node Installation



This section describes installing multiple Management Servers and installing MySQL on a node separate from the Management Servers.

System Requirements for Management Server

- Operating system: RHEL/CentOS 5.4+ 64-bit or RHEL6 64-bit
- 64-bit x86 CPU (more cores results in better performance)
- 2 GB of memory
- 80 GB of local disk
- At least 1 NIC
- Statically allocated IP address
- Fully qualified domain name as returned by the hostname command

Management Server Multi-Node Installation Overview

The Management Server download includes everything you need to get started, except MySQL. This includes the Management Server software as well as dependencies.

The procedure for the installation is:

1. Prepare the operating system
2. Prepare NFS shares
3. Prepare the system VM template
4. Install the first Management Server
5. Install MySQL on a separate node
6. Set up the MySQL database
7. Install additional Management Servers

Operating System and OS Preparation

The Management Server requires RHEL/CentOS 5.4 64 bit or later. You can download CentOS 64-bit via the following link: http://isoredirect.centos.org/centos/5/isos/x86_64/. The OS must be prepared to host the Management Server using the following steps.

These steps should be done on all Management Servers.

1. Log in to your OS as root. All the following commands should be run as root.
2. Check for a fully qualified hostname.

```
# hostname --fqdn
```

   This should return a fully qualified hostname such as "kvm1.lab.example.org". If it does not, edit /etc/hosts so that it does.

3. Disable SELinux.

   Set SELinux up to be permissive by default.

   a. First, set the SELINUX variable in /etc/selinux/config to “permissive”. This ensures that the CloudStack Agent can run properly on system reboot.
   b. Then set SELinux to permissive until the system is rebooted:

```
# setenforce permissive
```

4. Make sure that the Management Server can reach the Internet.

```
# ping www.google.com
```

5. (RHEL 6.x) If you do not have a Red Hat Network account, you need to prepare a local Yum repository.

   a. If you are working with a physical host, insert the RHEL 6.x installation CD. If you are using a VM, attach the RHEL6 ISO.
   b. Mount the CDROM to /media.
   c. Create a repo file at /etc/yum.repos.d/rhel6.repo. In the file, insert the following lines:

```
[rhel]
name=rhel6
baseurl=file:///media
enabled=1
gpgcheck=0
```

6. Turn on NTP for time synchronization.

   a. Install NTP.

```
# yum install ntp
```

   b. Edit the NTP configuration file to point to your NTP server.

```
# vi /etc/ntp.conf
```

TIP

NTP is required to synchronize the clocks of the servers in your cloud.

c. Restart the NTP client.

```
# service ntpd restart
```

d. Make sure NTP will start again upon reboot.

```
# chkconfig ntpd on
```

Prepare NFS Shares

CloudStack needs a place to keep primary and secondary storage. Both of these can be NFS shares. This section tells how to set up the NFS shares before adding the storage to CloudStack.

For primary storage, you can use iSCSI instead; for details, see Add Primary Storage on page 47.

1. On the storage server, create an NFS share for secondary storage. Export it with “rw,no_root_squash.” The exact commands to use vary depending on your system.
2. On one management server, create a mount point. For example:

```
# mkdir /mnt/secondary
```

3. Mount the secondary storage on your Management Server. Replace the example NFS server name and NFS share paths below with your own.

```
# mount -t nfs nfsservername:/nfs/share/secondary /mnt/secondary
```

4. If you are using NFS for primary storage as well, repeat these steps with a different NFS share and mount point.

Install the First Management Server

1. Install the CloudStack packages. You should have a file in the form of “CloudStack-NNNN.tar.gz”. Untar the file and then run the install.sh script inside it:

```
# tar xzf CloudStack-VERSION-1-rhel6.2.tar.gz
# cd CloudStack-VERSION-1-rhel6.2
# ./install.sh
```

2. Choose “M” to install the Management Server software.

Install the Database

1. Log in as root to your Database Node and run the following commands. If you are going to install a replica database then log in to the master.

```
# yum install mysql-server
# chkconfig --level 35 mysqld on
```

2. Edit the MySQL configuration (/etc/my.cnf or /etc/mysql/my.cnf, depending on your OS) and insert the following lines in the [mysqld] section. You can put these lines below the datadir line. The max_connections parameter

should be set to 350 multiplied by the number of Management Servers you are deploying. This example assumes 2 Management Servers.

```
innodb_rollback_on_timeout=1
innodb_lock_wait_timeout=600
max_connections=700
log-bin=mysql-bin
binlog-format = 'ROW'
```

NOTE: The binlog-format variable is supported in MySQL versions 5.1 and greater. It is not supported in MySQL 5.0. In some versions of MySQL, an underscore character is used in place of the hyphen in the variable name. For the exact syntax and spelling of each variable, consult the documentation for your version of MySQL.

3. Start the MySQL service, then invoke MySQL as the root user.

   ```
   # service mysqld start
   # mysql -u root
   ```

4. Best Practice: On RHEL and CentOS, MySQL does not set a root password by default. It is very strongly recommended that you set a root password as a security precaution. Run the following command, and substitute your own desired root password for <password>.

   ```
   mysql> SET PASSWORD = PASSWORD('<password>');
   ```

5. To grant access privileges to remote users, run the following command from the mysql prompt:

   ```
   mysql> GRANT ALL PRIVILEGES ON *.* TO ‘root’@’%’ WITH GRANT OPTION;
   ```

6. Restart the MySQL service.

   ```
   # service mysqld restart
   ```

7. Open the MySQL server port (3306) in the firewall to allow remote clients to connect.

   ```
   # iptables -I INPUT -p tcp --dport 3306 -j ACCEPT
   ```

8. Edit the /etc/sysconfig/iptables file and add the following lines at the beginning of the INPUT chain.

   ```
   -A INPUT -p tcp --dport 3306 -j ACCEPT
   ```

Create and Initialize the Database

1. Return to the root shell on your first Management Server.
2. Set up the database. The following command creates the cloud user on the database.
   - In dbpassword, specify the password to be assigned to the cloud user. You can choose to provide no password.
   - In dbhost, provide the hostname of the database node.
   - In deploy-as, specify the username and password of the user deploying the database. For example, if you originally installed MySQL with user “root” and password “password”, provide `--deploy-as=root:password`.

   ```
   # cloud-setup-databases cloud:<dbpassword>@<dbhost> --deploy-as=root:<password>
   ```

Best Practice: On RHEL and CentOS, SQL does not set a root password by default. It is very strongly recommended that you set a root password as a security precaution.

3. Configure the OS and start the Management Server:

```
# cloud-setup-management
```

Prepare and Start Additional Management Servers

For your second and subsequent Management Servers, you will install CloudStack, connect it to the database, and set up the OS for the Management Server.

1. Run these commands on each additional Management Server:

```
# tar xzf CloudStack-VERSION-1-rhel6.2.tar.gz
# cd CloudStack-VERSION-1-rhel6.2
# ./install.sh
```

2. Choose "M" to install the Management Server.
3. Configure the database client. Note the absence of the `--deploy-as` argument in this case.

```
# cloud-setup-databases cloud:<dbpassword>@<dbhost>
```

4. Configure the OS and start the Management Server:

```
# cloud-setup-management
```

The Management Server on this node should now be running.

5. Be sure to configure a load balancer for the Management Servers. See Management Server Load Balancing on page 111.

Prepare the System VM Template

Secondary storage must be seeded with a template that is used for CloudStack system VMs.

6. On the Management Server, run the following command to retrieve and decompress the system VM template. Run the command for each hypervisor type that you expect end users to run in this Zone.

When copying and pasting a command, be sure the command has pasted as a single line before executing. Some document viewers may introduce unwanted line breaks in copied text.

This process will require approximately 5 GB of free space on the local file system and up to 30 minutes each time it runs.

- For vSphere:

```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.ova -h vmware -F
```

- For KVM:

```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.qcow2.bz2 -h kvm -F
```

- For XenServer:

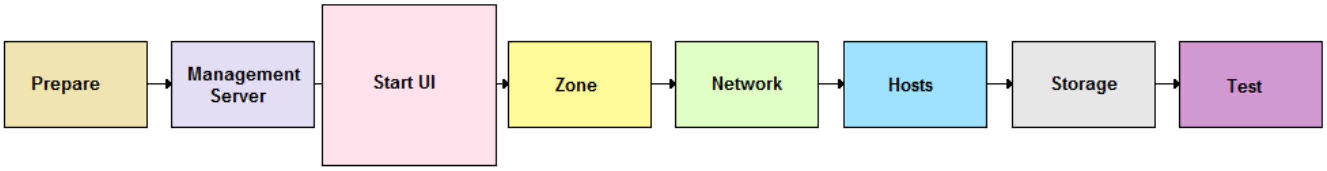
```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.vhd.bz2 -h xenserver
-F
```

7. When the script has finished, unmount secondary storage and remove the created directory.

```
# umount /mnt/secondary
# rmdir /mnt/secondary
```

8. Repeat these steps for each secondary storage server.

Log In to the CloudStack UI



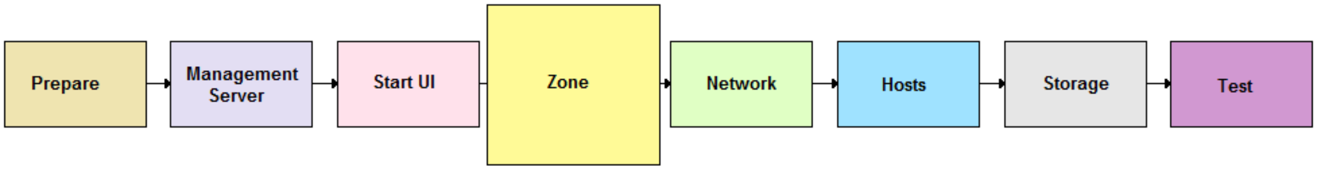
After installing the Management Server software, run the CloudStack user interface. This UI will help you set up your cloud infrastructure.

1. Open your favorite Web browser and go to this URL:

   ```
   http://management-server-ip-address:8080/client
   ```

2. Log in with the default administrator username "admin" and password "password." Leave Domain blank; this defaults to the ROOT domain.
3. The root administrator Dashboard appears.

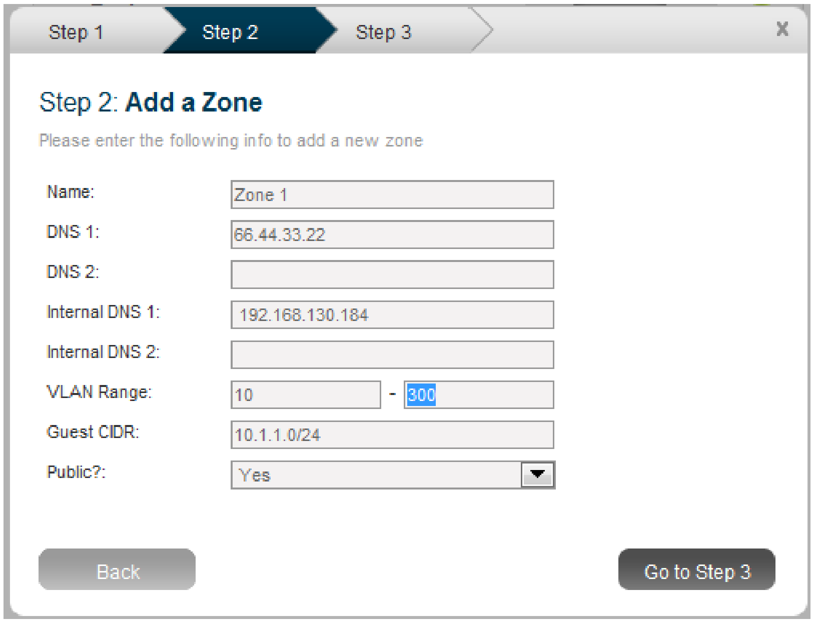
Add a Zone



Adding a Zone and Pod

Begin by using these steps to edit the default Zone.

1. In the left navigation tree, choose System, then Physical Resources.
2. Click on the + next to the Zone to add a Zone. The Zone creation wizard will appear.
3. Choose your network type.
   a. Basic: this is for use with Direct Untagged networking.
   b. Advanced: this is for use with Direct Tagged or Virtual Networking.



Step 2: **Add a Zone**

Please enter the following info to add a new zone

Name:	Zone 1
DNS 1:	66.44.33.22
DNS 2:	
Internal DNS 1:	192.168.130.184
Internal DNS 2:	
VLAN Range:	10 - 300
Guest CIDR:	10.1.1.0/24
Public?:	Yes

4. Enter the following details in the Add Zone dialog.
   - **Name**. The name of the Zone.
   - **DNS 1 and 2**. These are DNS servers for use by guest VMs in the Zone. These DNS servers will be accessed via the public network you will add later. The public IP addresses for the Zone must have a route to the DNS server named here.

- **Internal DNS 1 and 2**. These are DNS servers for use by system VMs in the Zone. These DNS servers will be accessed via the private network interface of the System VMs. The private IP address you provide for the Pods must have a route to the DNS server named here.
- **Zone VLAN Range (Advanced Only)**. This is the range of Zone VLANs that are used for provided VLANs to the guest networks. It is entered in the format “x-y”. (E.g. 800-900). You should set the Zone VLAN fields based on your planned VLAN allocation.

CloudStack distinguishes between internal and public DNS. Internal DNS is assumed to be capable of resolving internal-only hostnames, such as your NFS server’s DNS name. Public DNS is provided to the guest VMs for DNS resolution. You can enter the same DNS server for both types, but if you do so, you must make sure that both private and public IP addresses can route to the DNS server. Note that you must provide at least one public DNS server and at least one Internal DNS server.

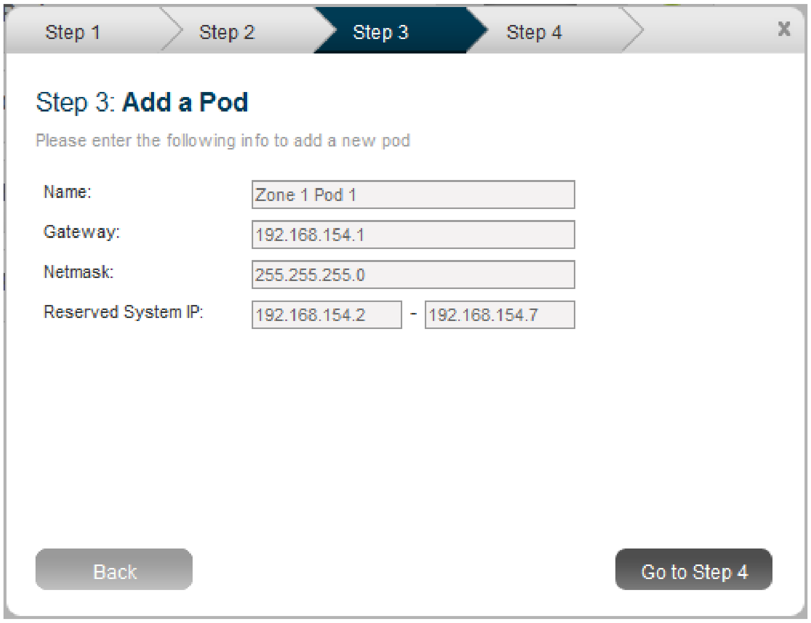
- **Guest CIDR (Advanced Only)**. This is the CIDR that describes the IP addresses in use in the guest virtual networks in this Zone. For example, 10.1.1.0/24. As a matter of good practice you should set different CIDRs for different Zones. This will make it easier to set up VPNs between virtual networks in different Zones in the future.

This field should be modified only if you are using Virtual Networking. The CloudStack uses it when VMs with Virtual Networking are created.

If you are using an external firewall, see **Advanced Networking: Adding an External Firewall (Optional)** on page 122.

- **Public**. Choose if this Zone is public or not. A public Zone is available to all users. A non-public Zone will be assigned to a particular domain. Only users in that domain will be allowed to create Guests in this Zone.

5. Add a Pod for the new Zone.



Step 3: **Add a Pod**

Please enter the following info to add a new pod

Name:	Zone 1 Pod 1
Gateway:	192.168.154.1
Netmask:	255.255.255.0
Reserved System IP:	192.168.154.2 - 192.168.154.7

6. Enter the following details in the Add Pod dialog.

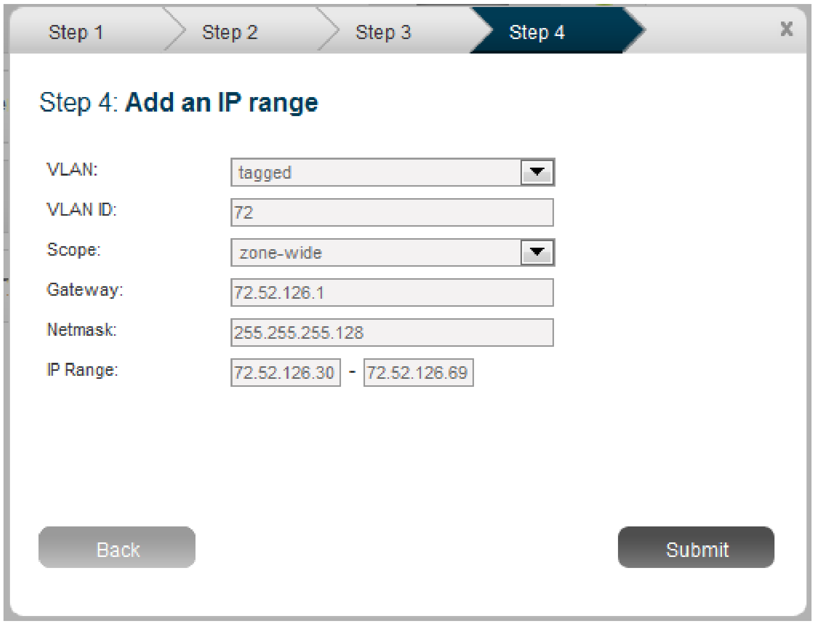
- **Name**. The name of the Pod.
- **Gateway**. The gateway for the hosts in that Pod.
- **CIDR**. The CIDR that encompasses the subnet in the Pod.

- **Reserved System IP**: This is the IP range in the private network that the CloudStack uses to manage Secondary Storage VMs and Console Proxy VMs. These IP addresses are taken from the same subnet as computing servers. You therefore need to make sure computing servers and Management Servers use IP addresses outside of this range. These two values combine to give the system control over a certain IP address range, and leave you in control of allocation for IP addresses in the CIDR but outside of the start and end range. In the screenshot we have start=192.168.154.2 and end=192.168.154.7. These computing servers and Management Servers can use IP addresses .8 to .254 and the CloudStack can use .2 to .7 for System VMs.

The recommended number of private IPs per Pod is described in Private IP Addresses on page 105. The example above allocates six IPs.

- **Guest IP Range** (Basic Mode Only). The range of IP addresses that will be available for allocation to guests in this Pod. If one NIC is used these IPs should be in the same CIDR as the Pod CIDR. If multiple NICs are used they may be in a different subnet.
- **Guest Netmask** (Basic Mode Only). The netmask in use on the subnet the guests will use.
- **Guest Gateway** (Basic Mode Only). The gateway that the guests should use.

7. In Advanced Networking, add a Public IP Address Range. If you intend to use Virtual Networking (with or without Direct Networking) you should enter the range of public IPs you have reserved for assignment to Virtual Routers and System VMs. If you intend to use Direct Networking only, you should enter the initial VLAN for the system, whether it is for a specific account or Zone-wide.



Step 4: **Add an IP range**

VLAN: tagged
VLAN ID: 72
Scope: zone-wide
Gateway: 72.52.126.1
Netmask: 255.255.255.128
IP Range: 72.52.126.30 - 72.52.126.69

8. Enter the following details in the Add IP Range dialog.

- **VLAN (tagged or untagged)**. Choose whether you will be using a tagged VLAN for public network traffic or an untagged VLAN. You must choose tagged if you are using a single NIC for all traffic.
- **VLAN ID**. The VLAN that will be used for public traffic if tagged VLAN was chosen**.**
- **Scope.** Choose if this is available to all accounts (Zone-wide) or is available for only one account (Account-specific). Almost all deployments should choose Zone-wide for the scope.
- **Gateway**. The gateway in use for these IP addresses.
- **Netmask**. The netmask associated with this IP range.

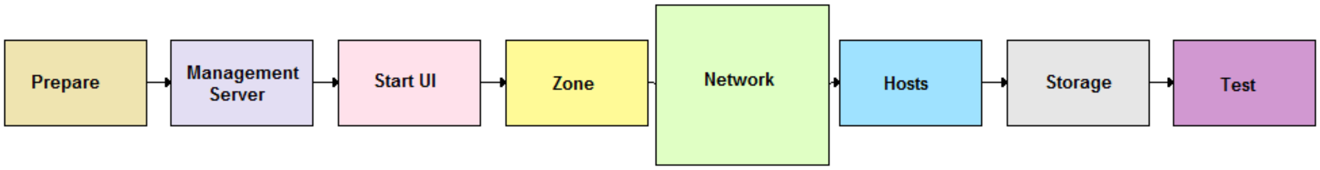
- **IP Range**. This takes a lower and an upper IP address that together form a range. These IP addresses are assumed to be accessible from the Internet and will be allocated for access to guest networks.

9. Click Submit.

Your first Zone and Pod are now added.

If you choose to add additional Zones, be sure to seed the system VM template as you did for the first Zone. See Add Secondary Storage on page 51.

Adding Networks (Advanced Networking)

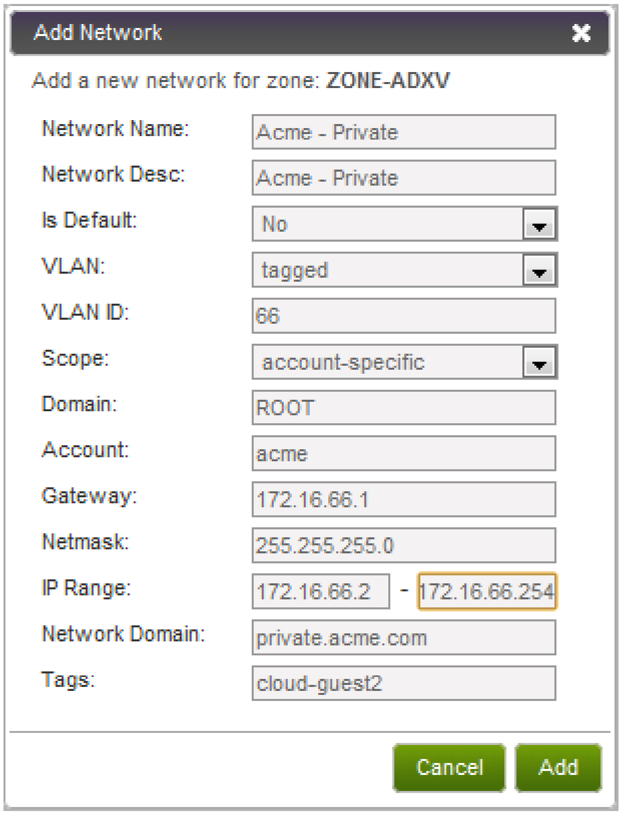


In a zone that uses advanced networking, CloudStack allows the provisioning of networks in addition to the base virtual network that is always available. These networks may span the Zone and be available to all accounts or they may be scoped to a single account. Each such network requires a Direct VLAN and an IP Range; a gateway for the network is optional.

A network in CloudStack is either eligible to the default network or ineligible to be the default network. A default network will be set as the default gateway for the guest using the DHCP response. Non-default networks will not be set to be the default gateway in the guest. Every guest must have exactly one default network. When multiple networks are present, the user is required to first choose a default network. Then they are given the choice to add zero or more non-default networks to their guest. Each network on the guest will be implemented as a NIC in the guest.

To add a new network to CloudStack:

1. In the Admin UI, go to System -> Physical Resources -> { target zone} -> Network.
2. Click "Add Network" in the top menu bar. You will see this dialog:



Add a new network for zone: **ZONE-ADXV**

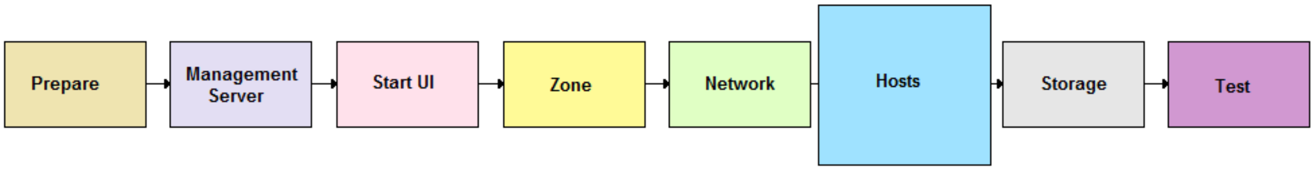
Network Name:	Acme - Private
Network Desc:	Acme - Private
Is Default:	No
VLAN:	tagged
VLAN ID:	66
Scope:	account-specific
Domain:	ROOT
Account:	acme
Gateway:	172.16.66.1
Netmask:	255.255.255.0
IP Range:	172.16.66.2 - 172.16.66.254
Network Domain:	private.acme.com
Tags:	cloud-guest2

This dialog requires the following information:

- Network Name. The name of the network. This will be user-visible.
- Network Desc: The description of the network. This will be user-visible.
- Is Default?: Choose yes if it is a default network, choose no if not.
- VLAN: The VLAN type. This is always tagged.
- VLAN ID: The VLAN tag for this network.
- Scope: Choose account-specific if you would like to limit a single account to accessing this VLAN. Choose zone-wide if all accounts with access to the Zone should be able to access the network.
- Domain/Account: If Scope is account-specific, enter the domain and account name for the account.
- Gateway: Enter the gateway for this network.
- Netmask: Enter the netmask for this network.
- IP Range: Enter the first and last IP addresses that define a range that CloudStack can assign to guests.
- Network Domain: (Optional) If you want to assign a special domain name to this network, specify the DNS suffix.
- Tags: (Optional) XenServer only. A name-label previously defined in the hypervisor to identify a particular network (see XenServer documentation for more details). Use this to map this network to any one of multiple networks defined in the hypervisor.

3. Click Add.

Add Cluster



You need to tell CloudStack about the hosts that it will manage. Hosts exist inside clusters, so before you begin adding hosts to the cloud, you must add at least one cluster.

About Clusters

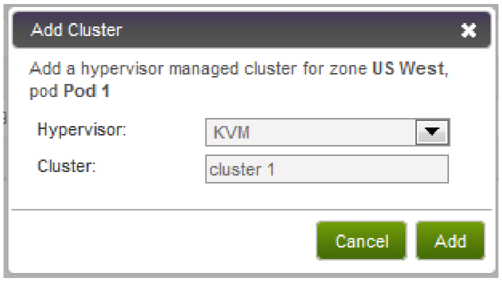
A Cluster is a XenServer server pool, a set of KVM servers, a set of OVM hosts, a VMware cluster preconfigured in vCenter, or a set of bare metal hosts. Within a Cluster, the Hosts may live migrate VMs to and from each other, and all access the same shared storage. Most deployments have a single Cluster per Pod, although CloudStack supports multiple Clusters per Pod.

Most deployments include more than one host in each cluster, because of the benefits of live migration. However, you may create a cluster with only one host.

Add Cluster: KVM or XenServer

To add a Cluster of hosts that run KVM or XenServer:

1. Navigate to System -> Physical Resources -> (Select Zone) -> (Select Pod).
2. Choose "Add Cluster" in the upper left.
3. The Add Cluster dialog will appear.



Add a hypervisor managed cluster for zone **US West**, pod **Pod 1**

Hypervisor: KVM

Cluster: cluster 1

4. Choose the hypervisor type for this Cluster.
5. Enter a name for the Cluster. This can be text of your choosing and is not used by CloudStack.
6. Complete the addition by clicking "Add".

Add Cluster: OVM

To add a Cluster of hosts that run Oracle VM (OVM):

1. Add a companion non-OVM cluster to the Pod. This cluster provides an environment where the CloudStack System VMs can run. You should have already installed a non-OVM hypervisor on at least one Host to prepare for this step. Depending on which hypervisor you used:
   - For VMWare, follow the steps in Add Cluster: vSphere on page 40. When finished, return here and continue with the next step.
   - For KVM or XenServer, follow the steps in Add Cluster: KVM or XenServer on page 39. When finished, return here and continue with the next step.
2. Navigate to System -> Physical Resources -> (Select Zone) -> (Select Pod). Select the same pod you used in step 1.
3. Choose "Add Cluster" in the upper left.
4. The Add Cluster dialog will appear.
5. In Hypervisor, choose OVM.
6. In Cluster, enter a name for the cluster.
7. Click Add.

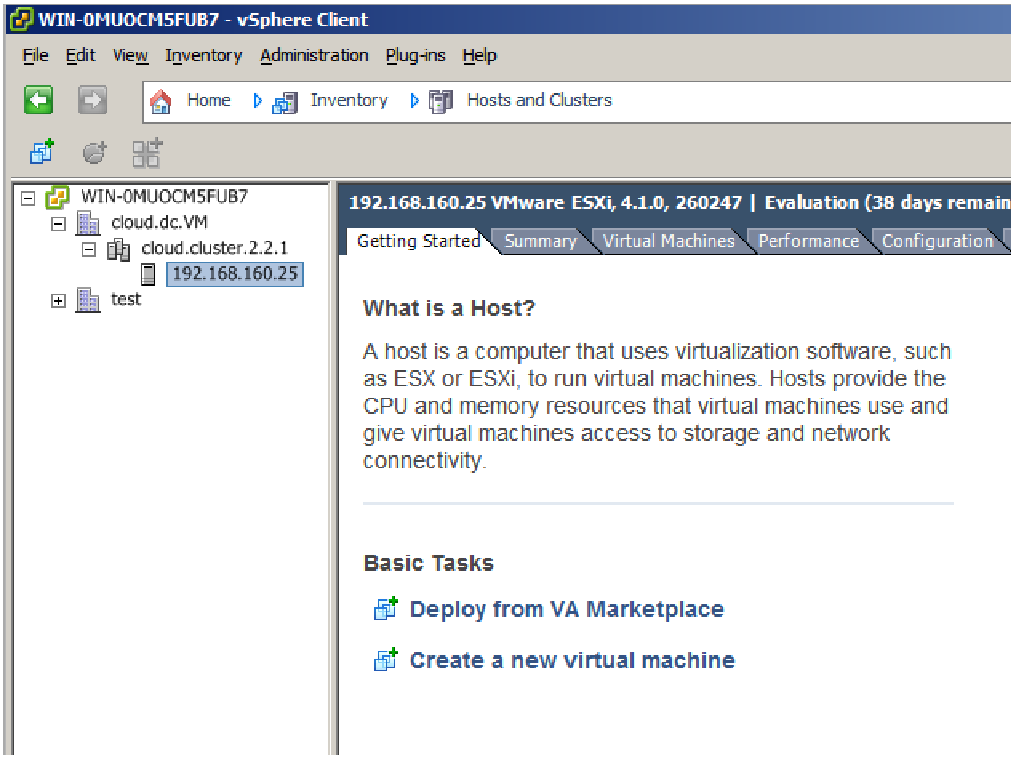
Add Cluster: vSphere

For vSphere servers, we recommend creating the cluster of hosts in vCenter and then adding the entire cluster to CloudStack. Follow these requirements:

- ➢ Do not put more than 8 hosts in a vSphere Cluster.

To add a vSphere cluster to CloudStack:

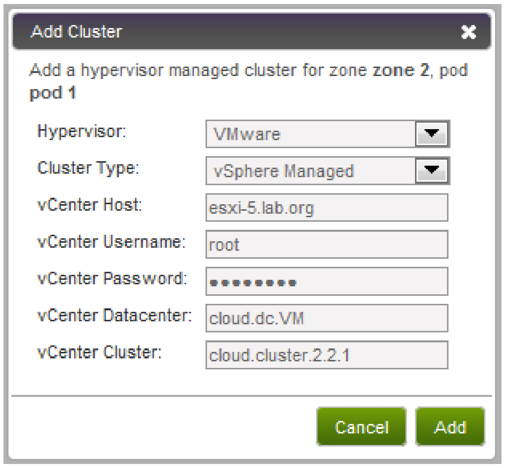
1. Create the cluster of hosts in vCenter. Follow the vCenter instructions to do this. You will create a cluster that looks something like this in vCenter.


WIN-0MUOCM5FUB7 - vSphere Client
File Edit View Inventory Administration Plug-ins Help
Home
Inventory
Hosts and Clusters
WIN-0MUOCM5FUB7
cloud.dc.VM
cloud.cluster.2.2.1
192.168.160.25
test
192.168.160.25 VMware ESXi, 4.1.0, 260247 | Evaluation (38 days remain
Getting Started
Summary
Virtual Machines
Performance
Configuration
Basic Tasks
Deploy from VA Marketplace
Create a new virtual machine

What is a Host?

A host is a computer that uses virtualization software, such as ESX or ESXi, to run virtual machines. Hosts provide the CPU and memory resources that virtual machines use and give virtual machines access to storage and network connectivity.

2. In the CloudStack UI, go to System -> Physical Resources -> (Select Zone) -> (Select Pod) -> Add Cluster.



Add a hypervisor managed cluster for zone **zone 2**, pod **pod 1**

Hypervisor:	VMware
Cluster Type:	vSphere Managed
vCenter Host:	esxi-5.lab.org
vCenter Username:	root
vCenter Password:	••••••••
vCenter Datacenter:	cloud.dc.VM
vCenter Cluster:	cloud.cluster.2.2.1

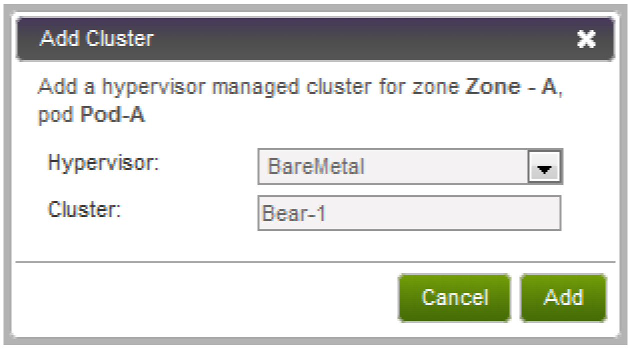
3. Provide the following information in the dialog. The fields below make reference to values from vCenter.
   - **Hypervisor**. Choose VMware.
   - **Cluster Type**. Choose vSphere Managed.
   - **vCenter Server**. Enter the hostname or IP address of the vCenter server.
   - **vCenter Username**. Enter the username that CloudStack should use to connect to vCenter. This user must have all administrative privileges.
   - **vCenter Password**. Enter the password for the user named above.
   - **vCenter Datacenter**. Enter the vCenter datacenter that the cluster is in. For example, "cloud.dc.VM".

- **vCenter Cluster**. Enter the name of the cluster you created in vCenter. For example, "cloud.cluster.2.2.1"

It may take a minute for the cluster to be provisioned. It will automatically display in the UI.

Add Cluster: Bare Metal

1. Before you can add a bare metal Cluster, you must have performed several other installation and setup steps to create a bare metal environment. See Bare Metal Installation on page 84.
2. Navigate to System -> Physical Resources -> (Select Zone) -> (Select Pod).
3. Choose "Add Cluster" in the upper left.
4. The Add Cluster dialog will appear.



Add Cluster

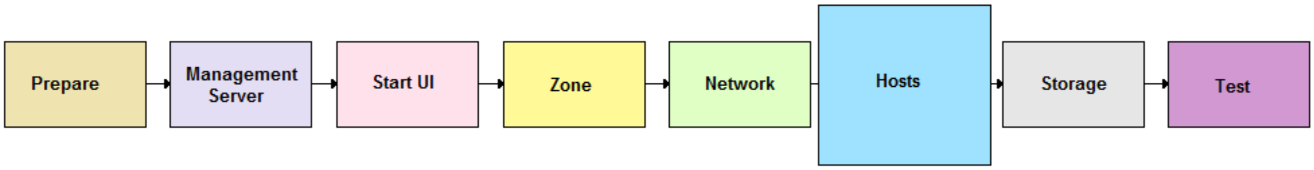
Add a hypervisor managed cluster for zone **Zone - A**, pod **Pod-A**

Hypervisor: BareMetal

Cluster: Bear-1

5. In Hypervisor, choose BareMetal.
6. In Cluster, enter a name for the Cluster. This can be any text you like.
7. Click Add.

Add Hosts



After adding at least one Cluster to your CloudStack configuration, you can start adding hosts. Additional Hosts may be added at any time up to the limit of nodes in a cluster for the selected Hypervisor type.

Add Hosts (KVM, OVM, or XenServer)

KVM, OVM, and XenServer hosts can be added to a Cluster at any time.

Requirements for XenServer, KVM, and OVM Hosts

Configuration requirements:

- Each Cluster must contain only hosts with the identical hypervisor.
- For XenServer, do not put more than 8 hosts in a Cluster.
- For KVM, do not put more than 16 hosts in a cluster.

For hardware requirements, see the appropriate section:

- Installing Citrix XenServer for CloudStack on page 55
- KVM Installation and Configuration on page 80
- Oracle VM (OVM) Installation and Configuration on page 84

XenServer Host Additional Requirements

If network bonding is in use, the administrator must cable the Host identically to other Hosts in the Cluster.

For all additional hosts to be added to the Cluster execute this step. This will cause the host to join the master in a XenServer pool.

```
# xe pool-join master-address=[master IP] master-username=root master-password=[your password]
```

With all hosts added to the XenServer pool, run the cloud-setup-bond script. This script will complete the configuration and setup of the bonds on the new Hosts in the Cluster.

1. Copy the script from the Management Server in /usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/cloud-setup-bonding.sh to the master Host and ensure it is executable.

2. Run the script:

```
# ./cloud-setup-bonding.sh
```

KVM Host Additional Requirements

If shared mountpoint storage is in use, the administrator should ensure that the new Host has all the same mountpoints (with storage mounted) as the other Hosts in the Cluster.

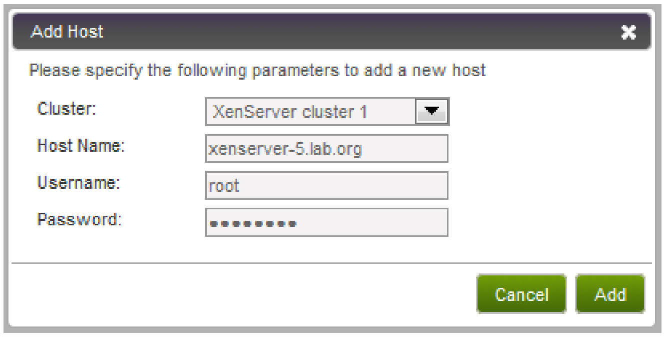
OVM Host Additional Requirements

Before adding a used host in CloudStack, as part of the cleanup procedure on the host, be sure to remove /etc/ovs-agent/db/.

Steps to Add a XenServer, KVM, or OVM Host

To add a host, follow these steps:

1. If you have not already done so, install the hypervisor software on the host. You will need to know which version of the hypervisor software version is supported by CloudStack and what additional configuration is required to ensure the host will work with CloudStack. To find these installation details, see:
   - Installing Citrix XenServer for CloudStack on page 55
   - KVM Installation and Configuration on page 80
   - Oracle VM (OVM) Installation and Configuration on page 0
2. In the CloudStack UI, go to System -> Physical Resources -> Zone -> Pod -> Add Host. The Add Host dialog displays.



Add Host

Please specify the following parameters to add a new host

Cluster: XenServer cluster 1

Host Name: xenserver-5.lab.org

Username: root

Password: ••••••••

3. Provide the following information in the Add Host dialog.
   - **Hypervisor**. The Hypervisor type for this Host.
   - **Cluster**. The Cluster to which this host will be added.
   - **Host Name**. The DNS name or IP address of the host.
   - **Username**. Usually root.
   - **Password**. This is the password for the user named above (from your XenServer, OVM, or KVM install).
   - **Agent Username/Password (OVM only)**. Enter the same agent user name and password that you specified during

If you ever remove an OVM host from a CloudStack OVM cluster, CloudStack will not umount the primary storage device on the host. This is to prevent accidental data loss. You should umount the device yourself or stop the OCFS2 service on the host.

OVM hypervisor installation.

- **Host Tags (Optional)**. Any labels that you use to categorize hosts for ease of maintenance.

It may take a minute for the host to be provisioned. It should automatically display in the UI.

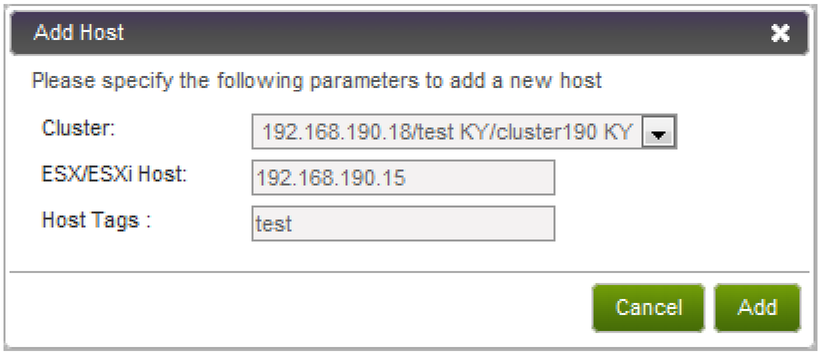
4. Repeat for additional Hosts.

Add Hosts (vSphere)

For vSphere servers, we recommend creating the cluster of hosts in vCenter and then adding the entire cluster to CloudStack. See Add Cluster: vSphere on page 40.

However, you can add an individual host to an existing vSphere cluster, either through vCenter or through the CloudStack UI.

To add a vSphere host through the CloudStack UI, navigate to the vSphere cluster, then click Add Host. The following dialog appears.



Add Host

Please specify the following parameters to add a new host

Cluster: 192.168.190.18/test KY/cluster190 KY

ESX/ESXi Host: 192.168.190.15

Host Tags : test

Provide the following:

- **Cluster:** The cluster you selected before clicking Add Host is automatically filled in for you.
- **ESX/ESXi Host:** The host IP address.
- **Host Tags:** (Optional) Any desired labels that you use to categorize hosts for ease of maintenance.

Add Hosts (Bare Metal)

To add a bare metal host follow these steps:

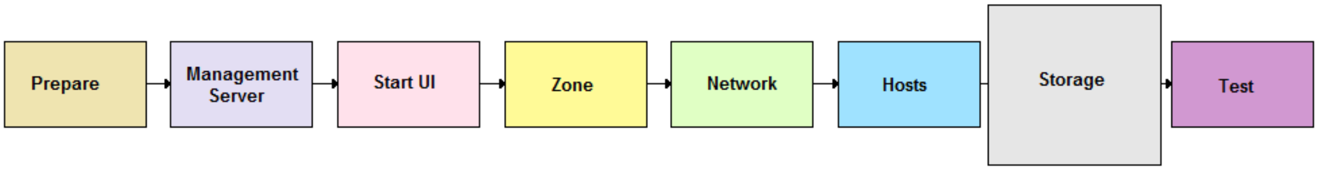
1. Before you can add a bare metal Host, you must have performed several other installation and setup steps to create a bare metal cluster and environment. See Bare Metal Installation on page 84.
2. Go to System -> Physical Resources -> Zone -> Pod -> Add Host.
3. Provide the following information in the Add Host dialog.
   - **Hypervisor**. Choose BareMetal.

- **Cluster**. The Cluster to which this host will be added. Give the name of a bare metal cluster that you created earlier (see Add Cluster: Bare Metal on page 42).
- **Host Name**. The IPMI IP address of the machine.
- **Username**. User name you set for IPMI.
- **Password**. Password you set for IPMI.
- **# of CPU Cores**. Number of CPUs on the machine.
- **CPU (in MHZ).** Frequency of CPU.
- **Memory (in MB).** Memory capacity of the new host.
- **Host MAC.** MAC address of the PXE NIC.
- **Tags.** Set to large. You will use this tag later when you create the service offering.

It may take a minute for the host to be provisioned. It should automatically display in the UI.

Repeat for additional bare metal hosts.

Add Primary Storage



You need to tell CloudStack about the primary and secondary storage devices that are available. You can add multiple primary storage servers to a Cluster. At least one is required.

If you intend to use only local disk for your installation, you can skip to Add Secondary Storage on page 50. Local disk is not supported for OVM.

For more concepts and details, see Primary Storage in the Administration Guide.

System Requirements for Primary Storage

Hardware requirements:

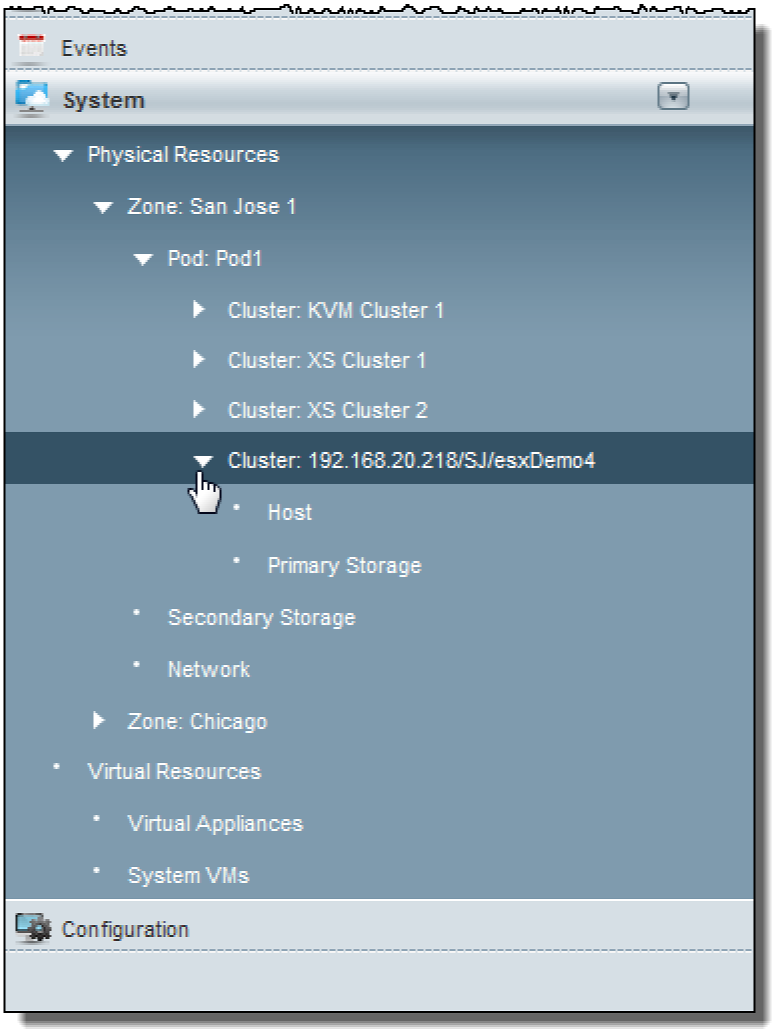
- Any standards-compliant iSCSI or NFS server that is supported by the underlying hypervisor.
- The storage server should be a machine with a large number of disks. The disks should ideally be managed by a hardware RAID controller.
- Minimum required capacity depends on your needs.

When setting up primary storage, follow these restrictions:

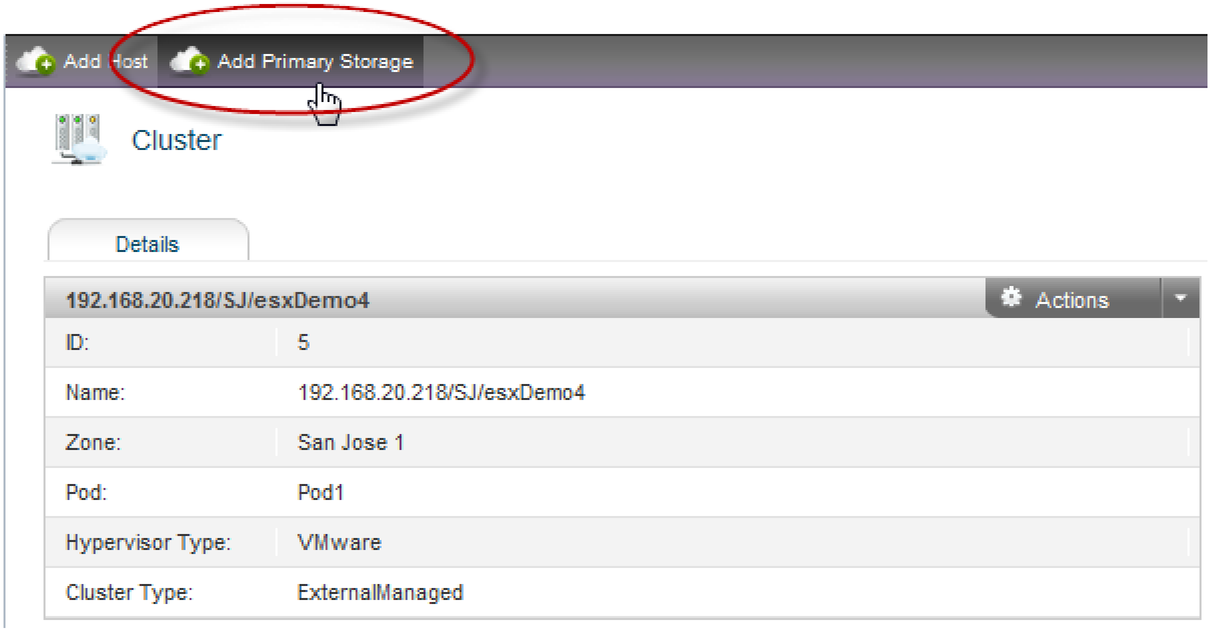
- Primary storage cannot be added until a Host has been added to the Cluster.
- If you do not provision shared storage for primary storage, you will not be able to create additional volumes (via Storage > Volumes > Add Volume).
- If you do not provision shared primary storage, you must set system.vm.local.storage.required to true, or else you will not be able to start VMs.

Steps to Add Primary Storage

1. To display the CloudStack cluster to which you want to add storage, click System, then click the triangles to expand Physical Resources, then the desired Zone, Pod, and Cluster.



2. Click Add Primary Storage.



192.168.20.218/SJ/esxDemo4

ID:	5
Name:	192.168.20.218/SJ/esxDemo4
Zone:	San Jose 1
Pod:	Pod1
Hypervisor Type:	VMware
Cluster Type:	ExternalManaged

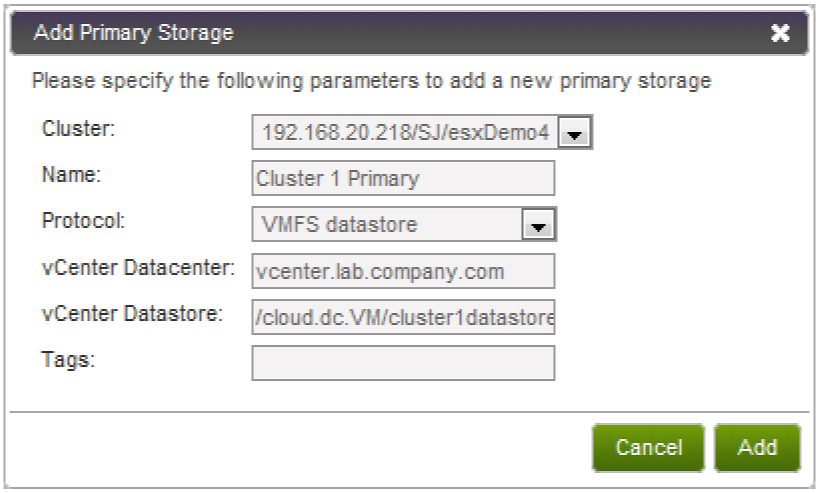
The Add Primary Storage dialog displays.

3. Provide the following information in the Add Primary Storage dialog. The information required varies depending on your choice in Protocol.
   - **Cluster**. The Cluster for the storage device.
   - **Name**. The name of the storage device.
   - **Protocol**. For XenServer, choose either NFS, iSCSI, or PreSetup. For KVM, choose NFS or SharedMountPoint. For vSphere choose either VMFS (iSCSI or FiberChannel) or NFS. For OVM, choose NFS or OCFS2.
   - **Server (for NFS, iSCSI, or PreSetup)**. The IP address or DNS name of the storage device.
   - **Server (for VMFS)**. The IP address or DNS name of the vCenter server.
   - **Path (for NFS)**. In NFS this is the exported path from the server.
   - **Path (for VMFS).** In vSphere this is a combination of the datacenter name and the datastore name. The format is "/" datacenter name "/" datastore name. For example, "/cloud.dc.VM/cluster1datastore".
   - **Path (for SharedMountPoint).** With KVM this is the path on each Host that is where this primary storage is mounted. For example, "/mnt/primary".
   - **Path (for OCFS2).** With OVM, this is a unique handle to reach the storage device; as a best practice, we recommend using a softlink, which you set up during OVM installation. For example, /dev/ovm-iscsi0. The path must be unique within the cloud. For more information, see Primary Storage Setup for OVM on page 85.
   - **SR Name-Label (for PreSetup).** Enter the name-label of the SR that has been set up outside CloudStack.
   - **Target IQN (for iSCSI)**. In iSCSI this is the IQN of the target. For example, iqn.1986-03.com.sun:02:01ec9bb549-1271378984
   - **Lun # (for iSCSI)**. In iSCSI this is the LUN number. For example, 3.

- **Tags (optional)**. The comma-separated list of tags for this storage device. It should be an equivalent set or superset of the tags on your disk offerings.

The tag sets on primary storage across clusters in a Zone must be identical. For example, if Cluster A provides primary storage that has tags T1 and T2, all other clusters in the Zone must also provide primary storage that has tags T1 and T2.

Here are some sample dialogs.

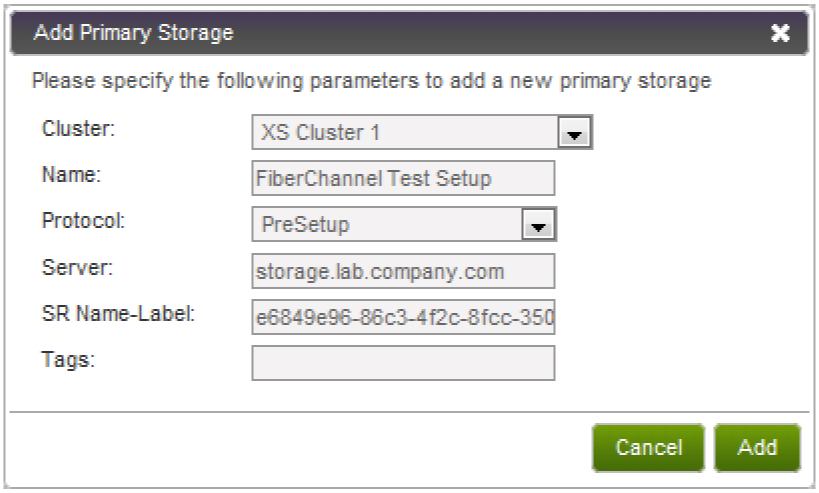


Add Primary Storage

Please specify the following parameters to add a new primary storage

Cluster:	192.168.20.218/SJ/esxDemo4
Name:	Cluster 1 Primary
Protocol:	VMFS datastore
vCenter Datacenter:	vcenter.lab.company.com
vCenter Datastore:	/cloud.dc.VM/cluster1datastore
Tags:	

Figure 1 Adding VMFS Primary Storage



Add Primary Storage

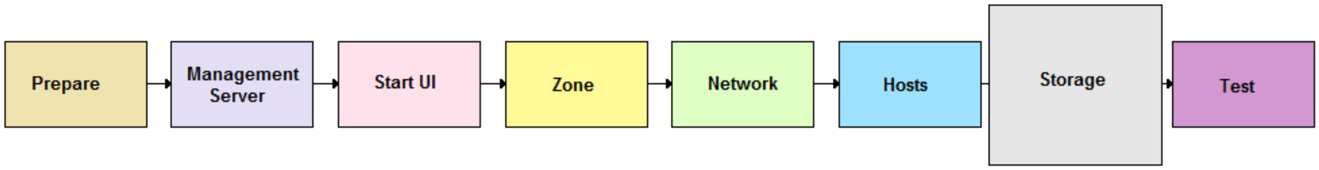
Please specify the following parameters to add a new primary storage

Cluster:	XS Cluster 1
Name:	FiberChannel Test Setup
Protocol:	PreSetup
Server:	storage.lab.company.com
SR Name-Label:	e6849e96-86c3-4f2c-8fcc-350
Tags:	

Figure 2 Adding Primary Storage That Was Set Up Manually (PreSetup)

4. Click Add.

Add Secondary Storage



You will need to add secondary storage.   Secondary storage is used to store templates, ISOs, and snapshots.

For more concepts and details, see Secondary Storage in the Administration Guide.

System Requirements for Secondary Storage

- NFS storage appliance or Linux NFS server
- 100GB minimum capacity

Steps to Add Secondary Storage

1. Be sure you created and mounted an NFS share during Management Server installation. See Prepare NFS Shares on page 23.
2. Secondary storage in all Zones must be seeded with a template that is used for system VMs such as the Virtual Router. This must be done for each Secondary storage server. First, mount the secondary storage on your Management Server. This example assumes the path on the secondary storage server is /nfs/share.

The 2.1 series system VM template will not work on CloudStack 2.2. You must use the 2.2 system VM template.

```
# mount -v -t nfs -o vers=3 servername:/nfs/share /mnt/secondary
```

3. On the Management Server, run one or more of the following commands to retrieve and decompress the system VM template. Run the command for each hypervisor type that you expect end users to run in this Zone.

   This process will require approximately 10 GB of free space on the local file system and up to 30 minutes each time it runs.

   - For vSphere:

```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.ova -h vmware -F
```

   - For KVM:

```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.qcow2.bz2 -h kvm -F
```

   - For XenServer:

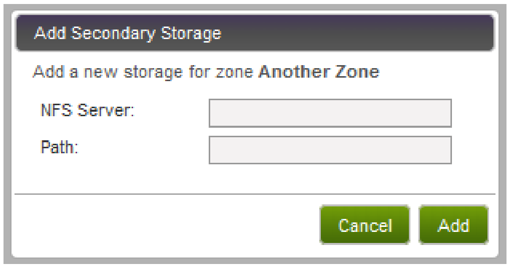
```
# /usr/lib64/cloud/agent/scripts/storage/secondary/cloud-install-sys-tmplt -m
/mnt/secondary -u http://download.cloud.com/releases/2.2.0/systemvm.vhd.bz2 -h xenserver
-F
```

- For OVM: The system VMs do not run on OVM, but on the non-OVM hypervisor installed on one of the clusters in the OVM Zone. Use the command version that applies to the hypervisor you chose to install in that cluster.

4. When the script has finished, unmount secondary storage.

```
# umount /mnt/secondary
```

5. Repeat steps 1 to 4 for each secondary storage server.
6. In the CloudStack UI, choose System -> Physical Resources, then click your Zone.
7. Click Add Secondary Storage.

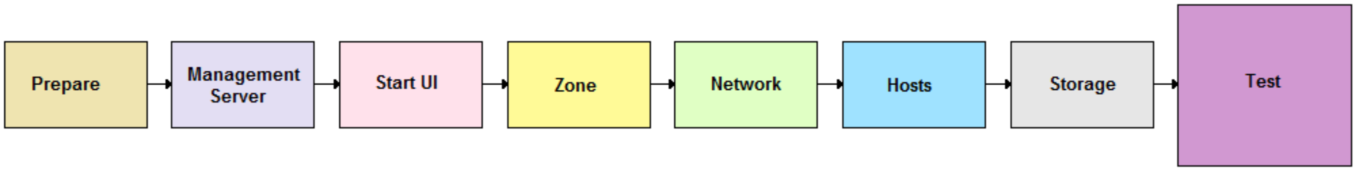


Add Secondary Storage

Add a new storage for zone **Another Zone**

8. Provide the details for the Secondary Storage server:
   - **Server**. The IP address of the server.
   - **Path**. The exported path from the server.
9. Repeat these steps to add more Secondary Storage servers to the zone.

Initialization and Testing



You should have one Java process running the Management Server software on each node. This is the Management Server process.

After everything is configured, CloudStack will perform its initialization. This can take 30 minutes or more, depending on the speed of your network. During this initialization process several things happen:

- CloudStack will start the Secondary Storage VM and Console Proxy VM from the system VM template downloaded into each Zone. In the System Section, Virtual Resources, System VMs section you will see the status of these VMs listed first as Creating, then as Starting, then as Running. You can click on Refresh is the upper right to update the status.
- After the Secondary Storage VM is running, the Management Server will initiate downloads of the CentOS templates. One is downloaded for each hypervisor type. The Management Server requests that the Secondary Storage VM perform this download. To check download status, log in to the UI as root and go to Templates - My Templates. The status will show “Storage agent or storage VM disconnected” until the Secondary Storage VM is running. Then the status will change to show that the download is in progress. You can click Refresh to update the download percentages.
- Once the CentOS templates are downloaded, they will be uncompressed by the Secondary Storage VM. This is a large file and this operation will take several minutes. The Management Server will then update each template’s status to Ready.

Confirm that these steps have completed successfully. Then use the following steps to try creating a virtual machine:

1. Create a new user account. Click on Accounts and then My Accounts. Click "Add Account" at the top.

   Follow the prompts to create a new user.
2. Log out, then log in again as the new user.
3. Go to the Instances tab. Click on My Instances.
4. Click Add Instance and follow the steps in the wizard.
   a. The template selection screen requires selecting a template. At this point you likely have only the provided CentOS template available.
   b. Select a service offering. Be sure that the hardware you have allows starting the selected service offering.
   c. Add any additional "data disk". This is a second volume that will be available to but not mounted in the guest. For example, in Linux on XenServer you will see /dev/xvdb in the guest after rebooting the VM. A reboot is not required if you have a PV-enabled OS kernel in use.
   d. Choose the primary network for the guest. Most likely you have only one option here. But if you entered additional networks, as in the direct tagged case, you may have more than one option.

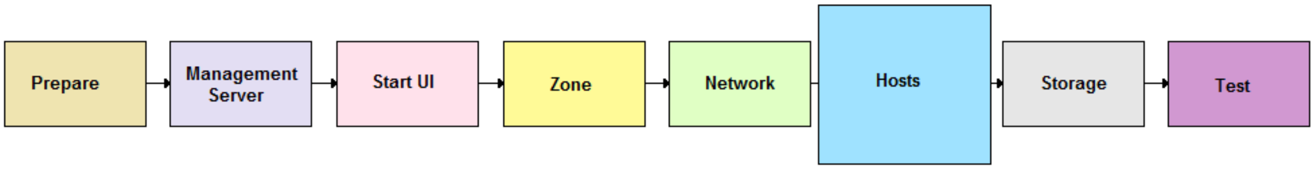
e. If applicable, select the desired security groups. Security groups are used to isolate groups of users in Direct Untagged networks using KVM or XenServer.

f. Optionally give your VM a name and a group. Use any descriptive text you would like.

g. Click Submit. Your VM will be created and started.

If you decide to grow your deployment, you can add more Hosts, Primary Storage, Zones, Pods, and Clusters.

Installing Citrix XenServer for CloudStack



If you want to use the Citrix XenServer hypervisor to run guest virtual machines, install XenServer 5.6, 5.6 FP1, or 5.6 SP2, with or without CSP, on the host(s) in your cloud. For an initial installation, follow the steps below. If you have previously installed XenServer and want to upgrade to another version, see Upgrading XenServer Versions on page 63.

System Requirements for XenServer Hosts

- Citrix XenServer® 5.6, 5.6 FP1, 5.6 SP2, or 5.6 SP2 + CSP
- The host must be certified as compatible with your chosen version of XenServer. See the Citrix Hardware Compatibility Guide: http://hcl.xensource.com/
- All hosts must be 64-bit and must support HVM (Intel-VT or AMD-V enabled).
- All Hosts within a Cluster must be homogenous. That means the CPUs must be of the same type, count, and feature flags. Seehttp://docs.vmd.citrix.com/XenServer/4.0.1/reference/ch02.html for more information on homogenous XenServer hosts.
- You must re-install Citrix XenServer if you are going to re-use a host from a previous install.
- 64-bit x86 CPU (more cores results in better performance)
- Hardware virtualization support required
- 4 GB of memory
- 30 GB of local disk
- At least 1 NIC
- Statically allocated IP Address

XenServer Installation Steps

1. Download one of the supported versions of XenServer from the Citrix website, http://www.citrix.com/English/ss/downloads/index.asp. The supported versions are listed in System Requirements for XenServer Hosts on page 55.
2. Install XenServer. Follow the instructions in the Citrix XenServer Installation Guide.
3. Following installation, perform the following additional configuration steps, which are described in the next few sections:

Required	Optional
Username and password	Install CSP package
Time synchronization	Configure public network with a dedicated NIC
Licensing	NIC bonding
	iSCSI multipath setup

Username and Password

All XenServers in a Cluster must have the same username and password as configured in CloudStack.

Time Synchronization

The XenServer host must be set to use NTP or another clock synchronization mechanism. All Hosts in a Pod must have the same time. You can use the NTP servers provided by Citrix:

```
0.xenserver.pool.ntp.org
1.xenserver.pool.ntp.org
2.xenserver.pool.ntp.org
3.xenserver.pool.ntp.org
```

Licensing

Citrix XenServer Free version provides 30 days usage without a license. Following the 30 day trial, XenServer requires a free activation and license. You can choose to install a license now or skip this step. If you skip this step, you will need to install a license when you activate and license the XenServer.

Getting and Deploying a License

If you choose to install a license now you will need to use the XenCenter to activate and get a license.

1. In XenCenter, click on **Tools > License manager**.
2. Select your XenServer and select **Activate Free XenServer**.
3. Request a license.

You can install the license with XenCenter or using the xe command line tool.

Install CloudStack XenServer Support Package (CSP)

To enable security groups on XenServer 5.6 SP2, you can download and install the CloudStack XenServer Support Package (CSP). After installing XenServer 5.6, perform the following additional steps on each XenServer host:

1. Download the CSP software onto the XenServer host from:

   http://download.cloud.com/releases/2.2.0/xenserver-cloud-supp.tgz

2. Extract the required file:

   ```
   # tar xf xenserver-cloud-supp.tgz
   ```

3. Change to the supplemental pack directory:

   ```
   # cd packages.cloud-supp/
   ```

4. Erase the previous iptables rules:

   ```
   yum erase iptables
   ```

5. Install the software packages:

   ```
   # yum -y --nogpgcheck install kernel-csp-xen-2.6.32.12-
   0.7.1.xs5.6.100.327.170613csp.i686.rpm
   # yum -y --nogpgcheck install csp-pack-5.6.100-50867p.noarch.rpm
   # yum -y --nogpgcheck install ebtables-2.0.9-1.el5.1.xs.i386.rpm
   # yum -y --nogpgcheck install arptables-0.0.3-4.i686.rpm
   # yum -y --nogpgcheck install iptables-1.3.5-5.3.el5_4.1.1.xs26.i386.rpm
   # yum -y --nogpgcheck install iptables-ipv6-1.3.5-5.3.el5_4.1.1.xs26.i386.rpm
   # yum -y --nogpgcheck install iptables-devel-1.3.5-5.3.el5_4.1.1.xs26.i386.rpm
   # yum -y --nogpgcheck install ipset-modules-xen-2.6.32.12-0.7.1.xs5.6.100.327.170613csp-
   4.5-1.xs26.i686.rpm
   # yum -y --nogpgcheck install ipset-4.5-1.xs26.i686.rpm
   ```

6. Change to the installation directory, make the post-installation script executable, and run it:

   ```
   # cd ..
   # chmod a+x post-install.sh
   #./post-install.sh
   ```

7. Reboot the XenServer:

```
# reboot
```

The XenServer host is now ready to be added to CloudStack.

Physical Networking Setup for XenServer

Once XenServer has been installed you may need to do some additional network configuration. At this point in the installation, you should have a plan for what NICs the Host will have and what traffic each NIC will carry. The NICs should be cabled as necessary to implement your plan.

> If a single NIC is your desired NIC configuration, there is no need for further configuration. You can skip this section.

If you plan on using NIC bonding, the NICs on all hosts in the Cluster must be cabled exactly the same. For example, if eth0 is in the private bond on one host in a cluster, then eth0 must be in the private bond on all hosts in the cluster.

The IP address assigned for the private network interface must be static. It can be set on the host itself or obtained via static DHCP.

CloudStack configures network traffic of various types to use different NICs or bonds on the XenServer host. You can control this process and provide input to the Management Server through the use of XenServer network name labels. The name labels are placed on physical interfaces or bonds and configured in CloudStack. In some simple cases the name labels are not required.

Configuring Public Network with a Dedicated NIC for XenServer (optional)

CloudStack supports the use of a second NIC (or bonded pair of NICs, described in NIC Bonding for XenServer (optional) on page 60) for the public network. If bonding is not used, the public network can be on any NIC and can be on different NICs on the Hosts in a cluster. For example, the public network can be on eth0 on node A and eth1 on node B. However, the XenServer name-label for the public network must be identical across all hosts. The following examples set the network label to “cloud-public”. After the management server is installed and running you must configure it with the name of the chosen network label (e.g. “cloud-public”); this is discussed in Management Server on page 21.

When a dedicated NIC is present for the public network, the public network can be implemented using a tagged or untagged VLAN.

If you are using two NICs bonded together to create a public network, see NIC Bonding.

If you are using a single dedicated NIC to provide public network access, follow this procedure on each new host that is added to CloudStack before adding the host.

1. Run `xe network-list` and find the public network. This is usually attached to the NIC that is public. Once you find the network make note of its UUID. Call this <UUID-Public>.
2. Run the following command.

```
# xe network-param-set name-label=cloud-public uuid=<UUID-Public>
```

Configuring Multiple Guest Networks for XenServer (optional)

CloudStack supports the use of multiple guest networks with the XenServer hypervisor. Each network is assigned a name-label in XenServer. For example, you might have two networks with the labels “cloud-guest” and “cloud-guest2”. After the management server is installed and running, you must add the networks and use these labels so that CloudStack is aware of the networks; this is discussed in Adding Networks on page 37.

Follow this procedure on each new host before adding the host to CloudStack:

1. Run `xe network-list` and find one of the guest networks. Once you find the network make note of its UUID. Call this <UUID-Guest>.
2. Run the following command, substituting your own name-label and uuid values.

   ```
   # xe network-param-set name-label=<cloud-guestN> uuid=<UUID-Guest>
   ```

3. Repeat these steps for each additional guest network, using a different name-label and uuid each time.

Separate Storage Network for XenServer (optional)

You can optionally set up a separate storage network. This should be done first on the host, before implementing the bonding steps below. This can be done using one or two available NICs. With two NICs bonding may be done as above. It is the administrator’s responsibility to set up a separate storage network.

Give the storage network a different name-label than what will be given for other networks.

For the separate storage network to work correctly, it must be the only interface that can ping the primary storage device’s IP address. For example, if eth0 is the private network NIC, `ping -I eth0 <primary storage device IP>` must fail. In all deployments, secondary storage devices must be pingable from the private NIC or bond. If a secondary storage device has been placed on the storage network, it must also be pingable via the storage network NIC or bond on the hosts as well.

You can set up two separate storage networks as well. For example, if you intend to implement iSCSI multipath, dedicate two non-bonded NICs to multipath. Each of the two networks needs a unique name-label.

If bonding is done, you should follow the procedure described in the private network section to set up the bond on the first host in the cluster as well as second and subsequent hosts in the cluster.

If no bonding is done the administrator must setup and name-label the separate storage network on all hosts (masters and slaves).

Here is an example to set up eth5 to access a storage network on 172.16.0.0/24.

```
# xe pif-list host-name-label=`hostname` device=eth5
uuid ( RO)                  : ab0d3dd4-5744-8fae-9693-a022c7a3471d
                device ( RO): eth5
# xe pif-reconfigure-ip DNS=172.16.3.3 gateway=172.16.0.1 IP=172.16.0.55 mode=static
netmask=255.255.255.0 uuid=ab0d3dd4-5744-8fae-9693-a022c7a3471d
```

NIC Bonding for XenServer (optional)

XenServer supports Source Level Balancing (SLB) NIC bonding. Two NICs can be bonded together to carry public, private, and guest traffic, or some combination of these. Separate storage networks are also possible. Here are some example supported configurations:

- 2 NICs on private, 2 NICs on public, 2 NICs on storage
- 2 NICs on private, 1 NIC on public, storage uses private network
- 2 NICs on private, 2 NICs on public, storage uses private network
- 1 NIC for private, public, and storage

All NIC bonding is optional.

XenServer expects all nodes in a cluster will have the same network cabling and same bonds implemented. In an installation the master will be the first host that was added to the cluster and the slave hosts will be all subsequent hosts added to the cluster. The bonds present on the master set the expectation for hosts added to the cluster later. The procedure to set up bonds on the master and slaves are different, and are described below. There are several important implications of this:

- You must set bonds on the first host added to a cluster. Then you must use xe commands as below to establish the same bonds in the second and subsequent hosts added to a cluster.
- Slave hosts in a cluster must be cabled exactly the same as the master. For example, if eth0 is in the private bond on the master, it must be in the private network for added slave hosts.

Private Network Bonding

The administrator must bond the private network NICs prior to adding the host to CloudStack.

Creating a Private Bond on the First Host in the Cluster

Use the following steps to create a bond in XenServer. These steps should be run on only the first host in a cluster. This example creates the cloud-private network with two physical NICs (eth0 and eth1) bonded into it.

1. Find the physical NICs that you want to bond together.

   ```
   # xe pif-list host-name-label=`hostname` device=eth0
   # xe pif-list host-name-label=`hostname` device=eth1
   ```

   These command shows the eth0 and eth1 NICs and their UUIDs. Substitute the ethX devices of your choice. Call the UUID's returned by the above command slave1-UUID and slave2-UUID.

2. Create a new network for the bond. For example, a new network with name "cloud-private".

   This label is important. CloudStack looks for a network by a name you configure. You must use the same name-label for all Hosts in the cloud for the private network.

   ```
   # xe network-create name-label=cloud-private
   # xe bond-create network-uuid=[uuid of cloud-private created above]
     pif-uuids=[slave1-uuid],[slave2-uuid]
   ```

Now you have a bonded pair that can be recognized by CloudStack as the private network.

Public Network Bonding

Bonding can be implemented on a separate, public network. The administrator is responsible for creating a bond for the public network if that network will be bonded and will be separate from the private network.

Creating a Public Bond on the First Host in the Cluster

These steps should be run on only the first host in a cluster. This example creates the cloud-public network with two physical NICs (eth2 and eth3) bonded into it.

1. Find the physical NICs that you want to bond together.

   ```
   # xe pif-list host-name-label=`hostname` device=eth2
   # xe pif-list host-name-label=`hostname` device=eth3
   ```

   These commands show the eth2 and eth3 NICs and their UUIDs. Substitute the ethX devices of your choice. Call the UUID's returned by the above command slave1-UUID and slave2-UUID.

2. Create a new network for the bond. For example, a new network with name "cloud-public".

   This label is important. CloudStack looks for a network by a name you configure. You must use the same name-label for all Hosts in the cloud for the public network.

   ```
   # xe network-create name-label=cloud-public
   # xe bond-create network-uuid=[uuid of cloud-public created above]
     pif-uuids=[slave1-uuid],[slave2-uuid]
   ```

Now you have a bonded pair that can be recognized by CloudStack as the public network.

Adding Additional Hosts to the Cluster

With the bonds (if any) established on the master you should add additional, slave hosts. For all additional hosts to be added to the Cluster execute this step. This will cause the host to join the master in a single XenServer pool.

```
# xe pool-join master-address=[master IP] master-username=root
  master-password=[your password]
```

Complete the Bonding Setup Across the Cluster

With all hosts added to the pool, run the cloud-setup-bond script. This script will complete the configuration and set up of the bonds across all hosts in the cluster.

1. Copy the script from the Management Server in /usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/cloud-setup-bonding.sh to the master Host and ensure it is executable.

2. Run the script

   ```
   # ./cloud-setup-bonding.sh
   ```

Now the bonds are set up and configured properly across the cluster.

Primary Storage Setup for XenServer

CloudStack natively supports NFS, iSCSI and local storage. If you are using one of these storage types, there is no need to create the XenServer Storage Repository ("SR").

If, however, you would like to use storage connected via some other technology such as FiberChannel, you must set up the SR yourself. To do so, perform the following steps. If you have your hosts in a XenServer pool, perform the steps on the master node. If you are working with a single XenServer which is not part of a cluster, perform the steps on that XenServer.

1. Connect FiberChannel cable to all hosts in the cluster and to the FiberChannel storage host.
2. Rescan the SCSI bus. Either use the following command or use XenCenter to perform an HBA rescan.

   ```
   # scsi-rescan
   ```

3. Repeat step 2 on every host.
4. Check to be sure you see the new SCSI disk.

   ```
   # ls /dev/disk/by-id/scsi-360a98000503365344e6f6177615a516b -l
   ```

   The output should look like this, although the specific file name will be different (scsi-<scsiID>):

   ```
   lrwxrwxrwx 1 root root 9 Mar 16 13:47
   /dev/disk/by-id/scsi-360a98000503365344e6f6177615a516b -> ../../sdc
   ```

5. Repeat step 4 on every host.
6. On the storage server, run this command to get a unique ID for the new SR.

   ```
   # uuidgen
   ```

   The output should look like this, although the specific ID will be different:

   ```
   e6849e96-86c3-4f2c-8fcc-350cc711be3d
   ```

7. Create the FiberChannel SR. In name-label, use the unique ID you just generated.

   ```
   # xe sr-create type=lvmohba shared=true
     device-config:SCSIid=360a98000503365344e6f6177615a516b
     name-label="e6849e96-86c3-4f2c-8fcc-350cc711be3d"
   ```

   This command returns a unique ID for the SR, like the following example (your ID will be different):

   ```
   7a143820-e893-6c6a-236e-472da6ee66bf
   ```

8. To create a human-readable description for the SR, use the following command. In uuid, use the SR ID returned by the previous command. In name-description, set whatever friendly text you prefer.

   ```
   # xe sr-param-set uuid=7a143820-e893-6c6a-236e-472da6ee66bf
     name-description="Fiber Channel storage repository"
   ```

   Make note of the values you will need when you add this storage to CloudStack later (see Add Primary Storage on page 47). In the Add Primary Storage dialog, in Protocol, you will choose PreSetup. In SR Name-Label, you will enter the name-label you set earlier (in this example, e6849e96-86c3-4f2c-8fcc-350cc711be3d).

9. (Optional) If you want to enable multipath I/O on a FiberChannel SAN, refer to the documentation provided by the SAN vendor.

If you are not going to do iSCSI Multipath Setup, go on to Management Server on page 21.

iSCSI Multipath Setup for XenServer (optional)

When setting up the storage repository on a Citrix XenServer, you can enable multipath I/O, which uses redundant physical components to provide greater reliability in the connection between the server and the SAN. To enable multipathing, use a SAN solution that is supported for Citrix servers and follow the procedures in Citrix documentation. The following links provide a starting point:

- http://support.citrix.com/article/CTX118791
- http://support.citrix.com/article/CTX125403

You can also ask your SAN vendor for advice about setting up your Citrix repository for multipathing.

Make note of the values you will need when you add this storage to the CloudStack later (see Add Primary Storage on page 47). In the Add Primary Storage dialog, in Protocol, you will choose PreSetup. In SR Name-Label, you will enter the same name used to create the SR.

If you encounter difficulty, address the support team for the SAN provided by your vendor. If they are not able to solve your issue, contact cloud.com support.

Upgrading XenServer Versions

This section tells how to upgrade XenServer software on CloudStack hosts. The actual upgrade is described in XenServer documentation, but there are some additional steps you must perform before and after the upgrade. The following upgrades are supported:

- XenServer 5.6 to XenServer 5.6 FP1
- XenServer 5.6 to XenServer 5.6 SP2
- XenServer 5.6 FP1 to XenServer 5.6 SP2

To upgrade XenServer software on CloudStack hosts:

1. Upgrade the database. On the Management Server node:

   a. Stop the management server and usage server:

```
# service cloud-usage stop
# service cloud-management stop
```

   b. Back up the database:

```
# mysqldump -h <management-server-IPaddress> -u cloud -p cloud> cloud.backup.sql
  Enter password:   <enter>
# mysqldump --user=root --databases cloud_usage > cloud_usage.backup.sql
```

c. Restart the management server and usage server:

```
# service cloud-management start
# service cloud-usage start
```

Note: you only need to do this once for all clusters.

2. Disconnect the XenServer cluster from CloudStack.

   a. Log in to the CloudStack UI as root.

   b. Navigate to the XenServer cluster, and click Actions – Unmanage.

   c. Watch the cluster status until it shows Unmanaged.

3. Log in to one of the hosts in the cluster, and run this command to clean up the VLAN:

```
# . /opt/xensource/bin/cloud-clean-vlan.sh
```

4. Still logged in to the host, run the upgrade preparation script:

```
# /opt/xensource/bin/cloud-prepare-upgrade.sh
```

Troubleshooting: If you see the error "can't eject CD," log in to the VM and umount the CD, then run the script again.

5. Upgrade the XenServer software on all hosts in the cluster. Upgrade the master first.

   a. Live migrate all VMs on this host to other hosts. See the instructions for live migration in the Administrator's Guide.

Troubleshooting: You might see the following error when you migrate a VM:

```
[root@xenserver-qa-2-49-4 ~]# xe vm-migrate live=true host=xenserver-qa-2-49-5 vm=i-2-8-
VM
You attempted an operation on a VM which requires PV drivers to be installed but the
drivers were not detected.
vm: b6cf79c8-02ee-050b-922f-49583d9f1a14 (i-2-8-VM)
```

To solve this issue, run the following:

```
# /opt/xensource/bin/make_migratable.sh  b6cf79c8-02ee-050b-922f-49583d9f1a14
```

   b. Reboot the host.

   c. Upgrade to the newer version of XenServer (5.6 FP1 or SP2). Use the steps in XenServer documentation.

   d. After the upgrade is complete, copy the following files from the management server to this host, in the directory locations shown below:

Copy this Management Server file…	…to this location on the XenServer host
/usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/xenserver56fp1/NFSSR.py	/opt/xensource/sm/NFSSR.py
/usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/setupxenserver.sh	/opt/xensource/bin/setupxenserver.sh
/usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/make_migratable.sh	/opt/xensource/bin/make_migratable.sh
/usr/lib64/cloud/agent/scripts/vm/hypervisor/xenserver/cloud-clean-vlan.sh	/opt/xensource/bin/cloud-clean-vlan.sh

e. Run the following script:

```
# /opt/xensource/bin/setupxenserver.sh
```

Troubleshooting: If you see the following error message, you can safely ignore it.

```
mv: cannot stat `/etc/cron.daily/logrotate': No such file or directory
```

f. Plug in the storage repositories (physical block devices) to the XenServer host:

```
# for pbd in `xe pbd-list currently-attached=false| grep ^uuid | awk '{print $NF}'`; do
xe pbd-plug uuid=$pbd ; done
```

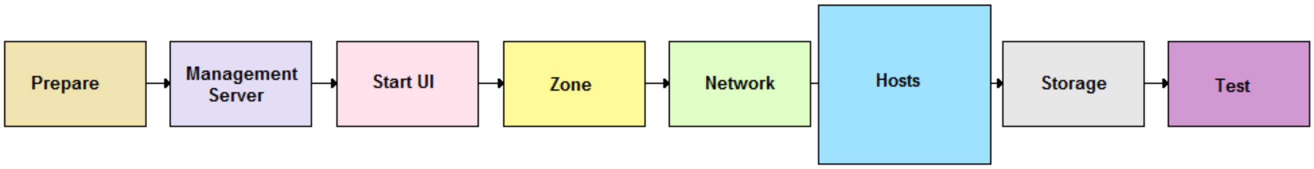
6. Repeat these steps to upgrade every host in the cluster to the same version of XenServer.
7. Run the following command on one host in the XenServer cluster to clean up the host tags:

```
# for host in $(xe host-list | grep ^uuid | awk '{print $NF}') ; do xe host-param-clear
uuid=$host param-name=tags; done;
```

8. Reconnect the XenServer cluster to CloudStack.
   a. Log in to the CloudStack UI as root.
   b. Navigate to the XenServer cluster, and click Actions – Manage.
   c. Watch the status to see that all the hosts come up.
9. After all hosts are up, run the following on one host in the cluster:

```
# /opt/xensource/bin/cloud-clean-vlan.sh
```

VMware vSphere Installation and Configuration



If you want to use the VMware vSphere hypervisor to run guest virtual machines, install vSphere on the host(s) in your cloud.

System Requirements for vSphere Hosts

Software requirements:

- vSphere and vCenter, both version 4.1.

  vSphere Standard is recommended. Note however that customers need to consider the CPU constraints in place with vSphere licensing. See http://www.vmware.com/files/pdf/vsphere_pricing.pdf and discuss with your VMware sales representative.

  vCenter Server Standard is recommended.

Hardware requirements:

- The host must be certified as compatible with vSphere. See the VMware Hardware Compatibility Guide at http://www.vmware.com/resources/compatibility/search.php.
- All hosts must be 64-bit and must support HVM (Intel-VT or AMD-V enabled).
- All Hosts within a Cluster must be homogenous. That means the CPUs must be of the same type, count, and feature flags.
- 64-bit x86 CPU (more cores results in better performance)
- Hardware virtualization support required
- 4 GB of memory
- 30 GB of local disk
- At least 1 NIC
- Statically allocated IP Address

vCenter Server requirements:

- Processor – 2 CPUs 2.0GHz or higher Intel or AMD x86 processors. Processor may be higher if the database runs on the same machine.
- Memory – 3GB RAM. RAM requirements may be higher if your database runs on the same machine.
- Disk storage – 2GB. Disk requirements may be higher if your database runs on the same machine.

- Microsoft SQL Server 2005 Express disk requirements. The bundled database requires up to 2GB free disk space to decompress the installation archive.
- Networking – 1Gbit or 10Gbit.

  For more information, see "vCenter Server and the vSphere Client Hardware Requirements" at http://pubs.vmware.com/vsp40/wwhelp/wwhimpl/js/html/wwhelp.htm#href=install/c_vc_hw.html.

Other requirements:

- VMware vCenter Standard Edition 4.1 must be installed and available to manage the vSphere Hosts.
- vCenter must be configured to use the standard port 443 so that it can communicate with the CloudStack Management Server.
- You must re-install VMware ESXi if you are going to re-use a host from a previous install.
- CloudStack requires VMware vSphere 4.1. VMware vSphere 4.0 is not supported.
- All hosts must be 64-bit and must support HVM (Intel-VT or AMD-V enabled). All Hosts within a Cluster must be homogenous. That means the CPUs must be of the same type, count, and feature flags.
- The CloudStack private network **must not** be configured as a separate tagged virtual network. The CloudStack private network is the same as the vCenter management network, and will inherit its configuration. See Configure vCenter Management Network on page 73.
- CloudStack requires ESXi. ESX is not supported.
- All resources used for CloudStack must be used for CloudStack only. CloudStack cannot shares instance of ESXi or storage with other management consoles. Do not share the same storage volumes that will be used by CloudStack with a different set of ESXi servers that are not managed by CloudStack.
- Put all target ESXi hypervisors in a Cluster in a separate Datacenter in vCenter.
- The cluster that will be managed by CloudStack should not contain any VMs. Do not run the management server, vCenter or any other VMs on the cluster that is designated for CloudStack use. Create a separate cluster for use of CloudStack and make sure that they are no VMs in this cluster.
- All the required VLANS must be trunked into all the ESXi hypervisor hosts. These would include the VLANS for Management, Storage, vMotion, and guest VLANs. The guest VLAN (used in Advanced Networking; see Network Setup on page 21) is a contiguous range of VLANs that will be managed by CloudStack. CloudStack does not support Distributed vSwitches in VMware.

Preparation Checklist for VMware

For a smoother installation, gather the following information before you start:

- Preparation Checklist on page 18
- Networking Checklist on page 68
- vCenter Checklist on page 69

Networking Checklist for VMware

You will need the following information about the VLAN.

Networking Requirement	Value	Notes
ESXi VLAN		VLAN on which all your ESXi hypervisors reside.
ESXI VLAN IP Address		IP Address Range in the ESXi VLAN. One address per Virtual Router is used from this range.
ESXi VLAN IP Gateway		
ESXi VLAN Netmask		
Management Server VLAN		VLAN on which the CloudStack Management server is installed.
Public VLAN		VLAN for the Public Network.
Public VLAN IP Address Range		Range of Public IP Addresses available for CloudStack use. These addresses will be used for virtual router on CloudStack to route private traffic to external networks.
Public VLAN Gateway		
Public VLAN Netmask		
VLAN Range for Customer use		A contiguous range of non-routable VLANs. One VLAN will be assigned for each customer.

vCenter Checklist

You will need the following information about vCenter.

vCenter Requirement	Value	Notes
vCenter User		This user must have admin privileges.
vCenter User Password		Password for the above user.
vCenter Datacenter Name		Name of the datacenter.
vCenter Cluster Name		Name of the cluster.

vSphere Installation Steps

1. Download and purchase vSphere from the VMware Website (https://www.vmware.com/tryvmware/index.php?p=vmware-vsphere&lp=1) and install it by following the VMware vSphere Installation Guide.
2. Following installation, perform the following additional configuration steps, which are described in the next few sections:

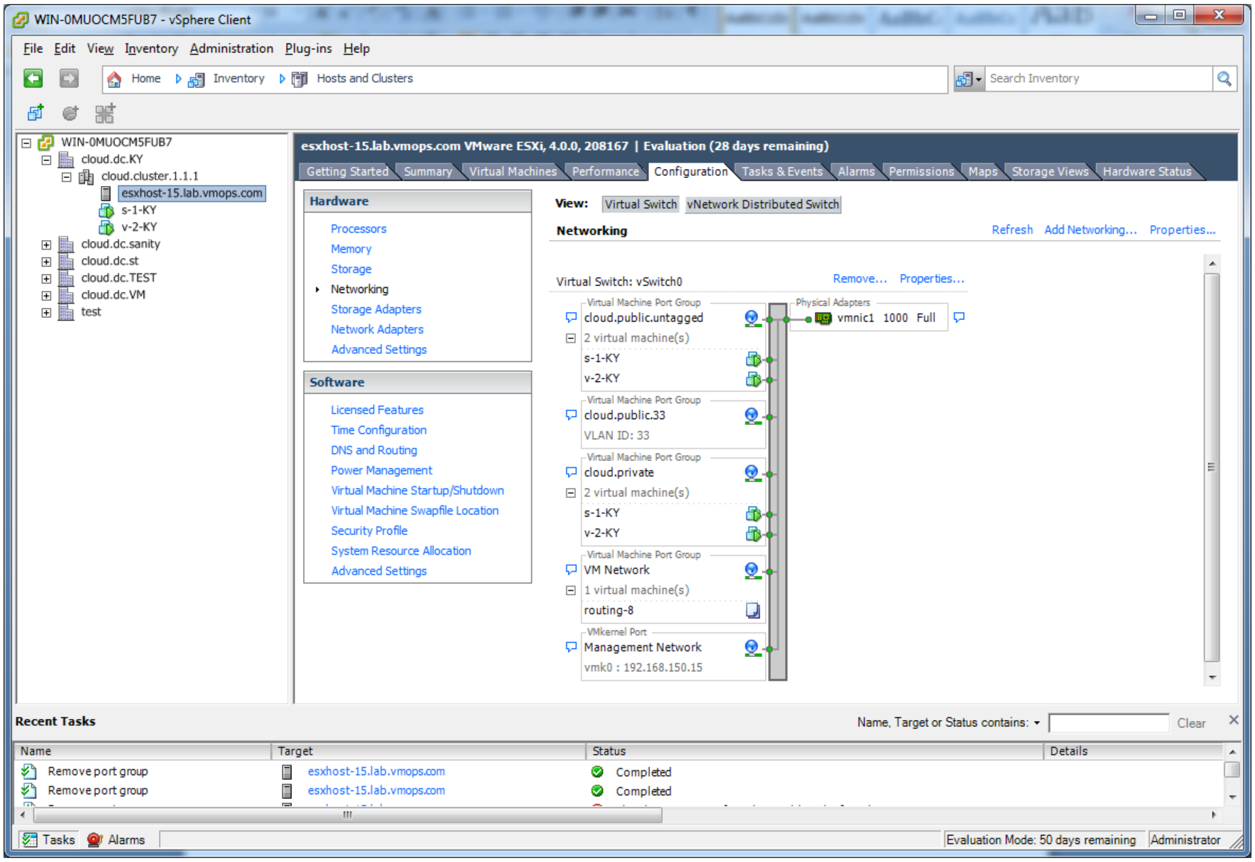
Required	Optional
ESXi host setup	NIC bonding
Configure host physical networking, virtual switch, vCenter Management Network, and extended port range	Multipath storage
Prepare storage for iSCSI	
Configure clusters in vCenter and add hosts to them, or add hosts without clusters to vCenter	

ESXi Host setup

All ESXi hosts should enable CPU hardware virtualization support in BIOS. Please note hardware virtualization support is not enabled by default on most servers.

Physical Host Networking

You should have a plan for cabling the vSphere hosts. Proper network configuration is required before adding a vSphere Host to CloudStack. To configure an ESXi host, you can use vClient to add it as standalone host to vCenter first. Once you see the host appearing in the vCenter inventory tree, click the host node in the inventory tree, and navigate to the Configuration tab.



Name	Target	Status	Details
Remove port group	esxhost-15.lab.vmops.com	Completed	
Remove port group	esxhost-15.lab.vmops.com	Completed	

In the host configuration tab, click the “Hardware/Networking” link to bring up the networking configuration page as above.

Configure Virtual Switch

A default virtual switch vSwitch0 is created. CloudStack requires all ESXi hosts in the cloud to use the same set of virtual switch names. If you change the default virtual switch name, you will need to configure one or more CloudStack configuration variables as well. See Edit the Global Configuration Settings on page 118.

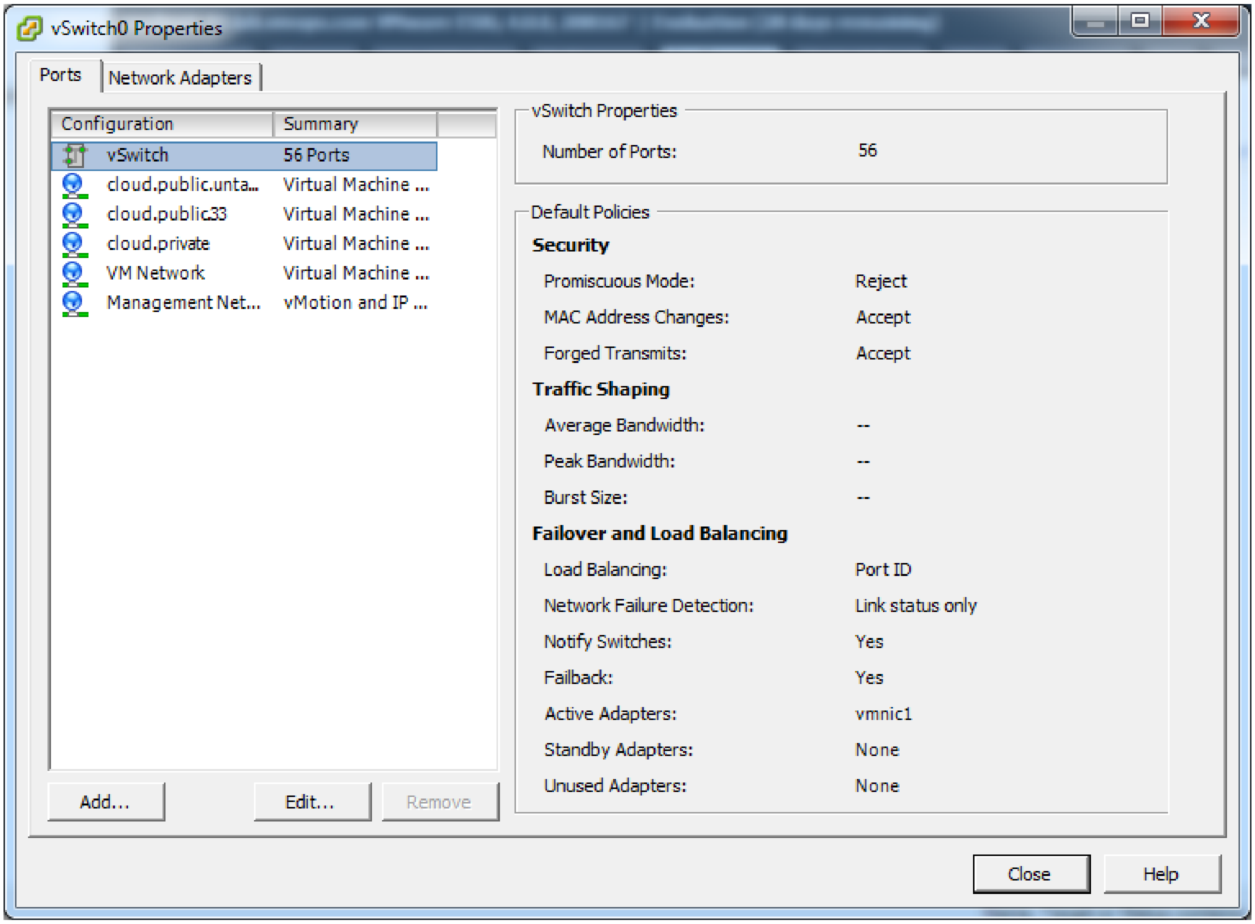
Separating Traffic

CloudStack allows you to use vCenter to configure three separate networks per ESXi host. These networks are identified by the name of the vSwitch they are connected to. The allowed networks for configuration are public (for traffic to/from the public internet), guest (for guest-guest traffic), and private (for management and usually storage traffic). You can use the default virtual switch for all three, or create one or two other vSwitches for those traffic types.

If you want to separate traffic in this way you should first create and configure vSwitches in vCenter according to the vCenter instructions. Take note of the vSwitch names you have used for each traffic type. You will configure CloudStack to use these vSwitches as described in Edit the Global Configuration Settings on page 118.

Increasing Ports

By default a virtual switch on ESXi hosts is created with 56 ports. We recommend setting it to 4096, the maximum number of ports allowed. To do that, click the “Properties…” link for virtual switch (note this is not the Properties link for Networking).



Configuration	Summary
vSwitch	56 Ports
cloud.public.unta...	Virtual Machine ...
cloud.public33	Virtual Machine ...
cloud.private	Virtual Machine ...
VM Network	Virtual Machine ...
Management Net...	vMotion and IP ...

vSwitch Properties

Number of Ports: 56

Default Policies

Security

Promiscuous Mode:	Reject
MAC Address Changes:	Accept
Forged Transmits:	Accept

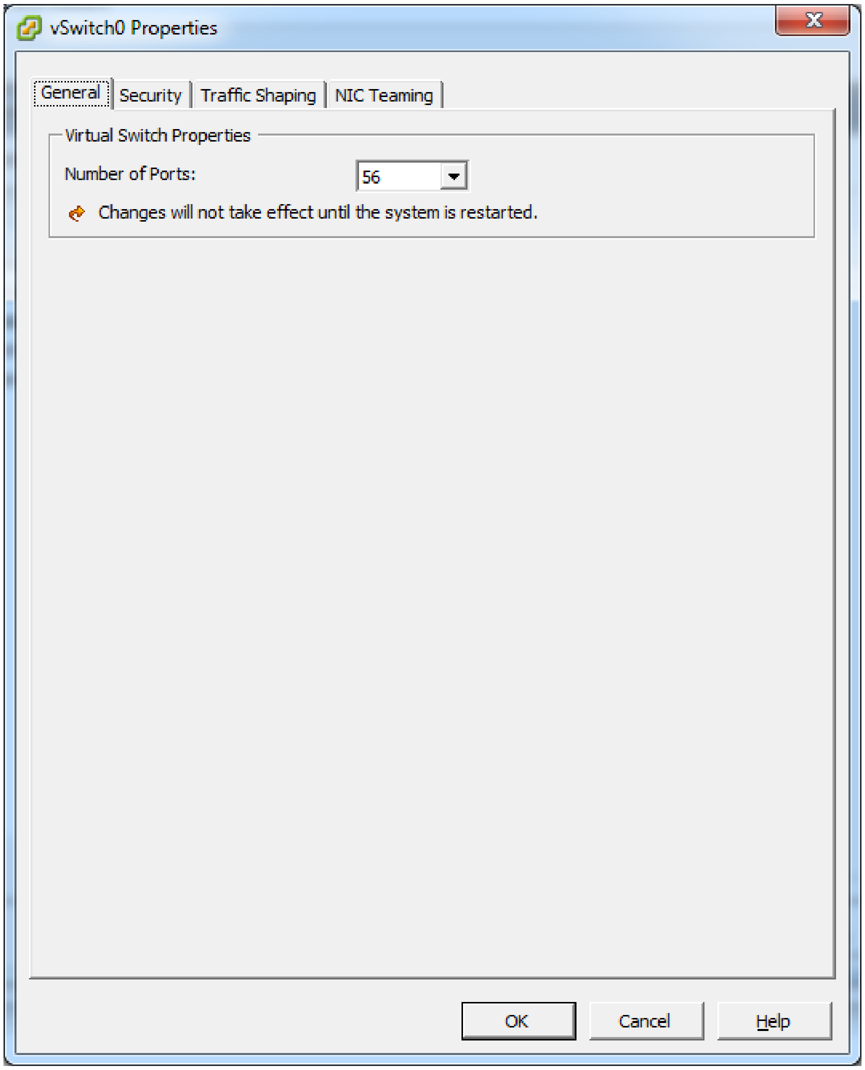
Traffic Shaping

Average Bandwidth:	--
Peak Bandwidth:	--
Burst Size:	--

Failover and Load Balancing

Load Balancing:	Port ID
Network Failure Detection:	Link status only
Notify Switches:	Yes
Failback:	Yes
Active Adapters:	vmnic1
Standby Adapters:	None
Unused Adapters:	None

In vSwitch properties dialog, select the vSwitch and click Edit. You should see the following dialog:



Virtual Switch Properties

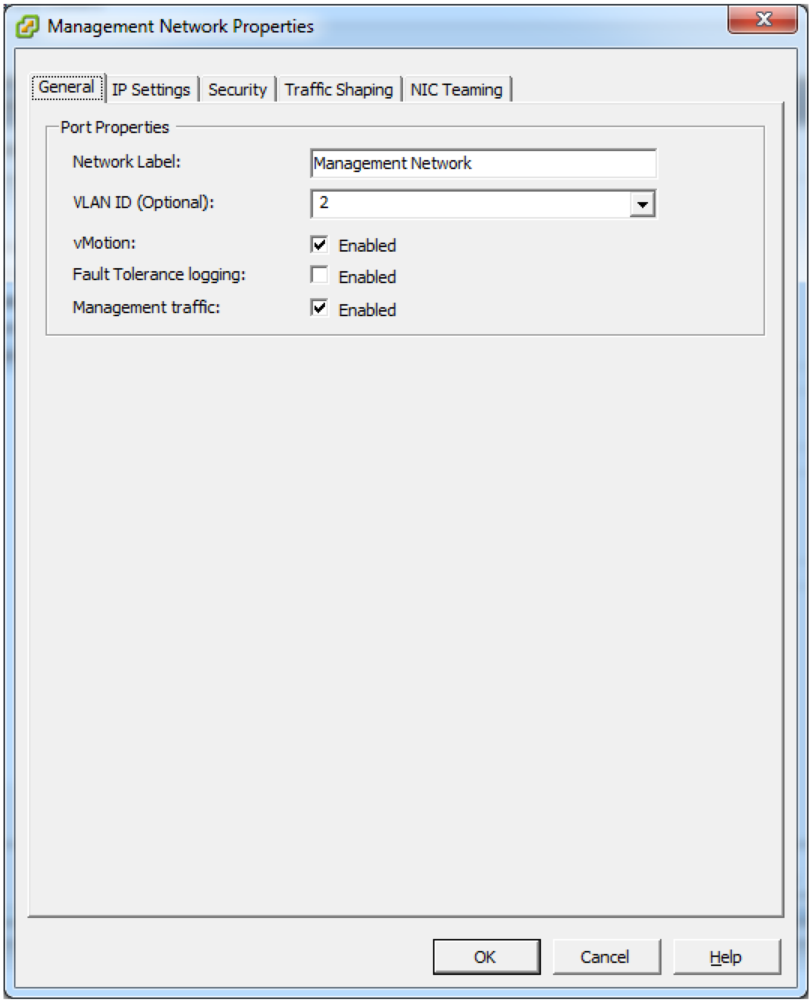
Number of Ports: 56

Changes will not take effect until the system is restarted.

In this dialog, you can change the number of switch ports. After you've done that, ESXi hosts are required to reboot in order for the setting to take effect.

Configure vCenter Management Network

In the vSwitch properties dialog box, you may see a vCenter management network. This same network will also be used as the CloudStack private network. CloudStack requires the vCenter management network to be configured properly. Select the management network item in the dialog, then click Edit.



Port Properties

Network Label: Management Network

VLAN ID (Optional): 2

vMotion: ☑ Enabled

Fault Tolerance logging: ☐ Enabled

Management traffic: ☑ Enabled

Make sure the following values are set:

- VLAN ID set to either the desired ID (for a tagged VLAN) or None (for an untagged network).
- vMotion enabled.
- Management traffic enabled.

If the ESXi hosts have multiple VMKernel ports, and ESXi is not using the default value "Management Network" as the management network name, you must follow these guidelines to configure the management network port group so that CloudStack can find it:

- Use one label for the management network port across all ESXi hosts.
- In the CloudStack UI, go to Configuration – Global Settings and set vmware.management.portgroup to the management network label from the ESXi hosts.

Extend Port Range for CloudStack Console Proxy

You need to extend the range of firewall ports that the console proxy works with on the hosts. This is to enable the console proxy to work with VMware-based VMs. The default additional port range is 59000-60000. To extend the port range, log in to the VMware ESX service console on each host and run the following commands:

```
esxcfg-firewall -o 59000-60000,tcp,in,vncextras
esxcfg-firewall -o 59000-60000,tcp,out,vncextras
```

Configure NIC Bonding for vSphere

NIC bonding on vSphere Hosts may be done according to the vSphere installation guide.

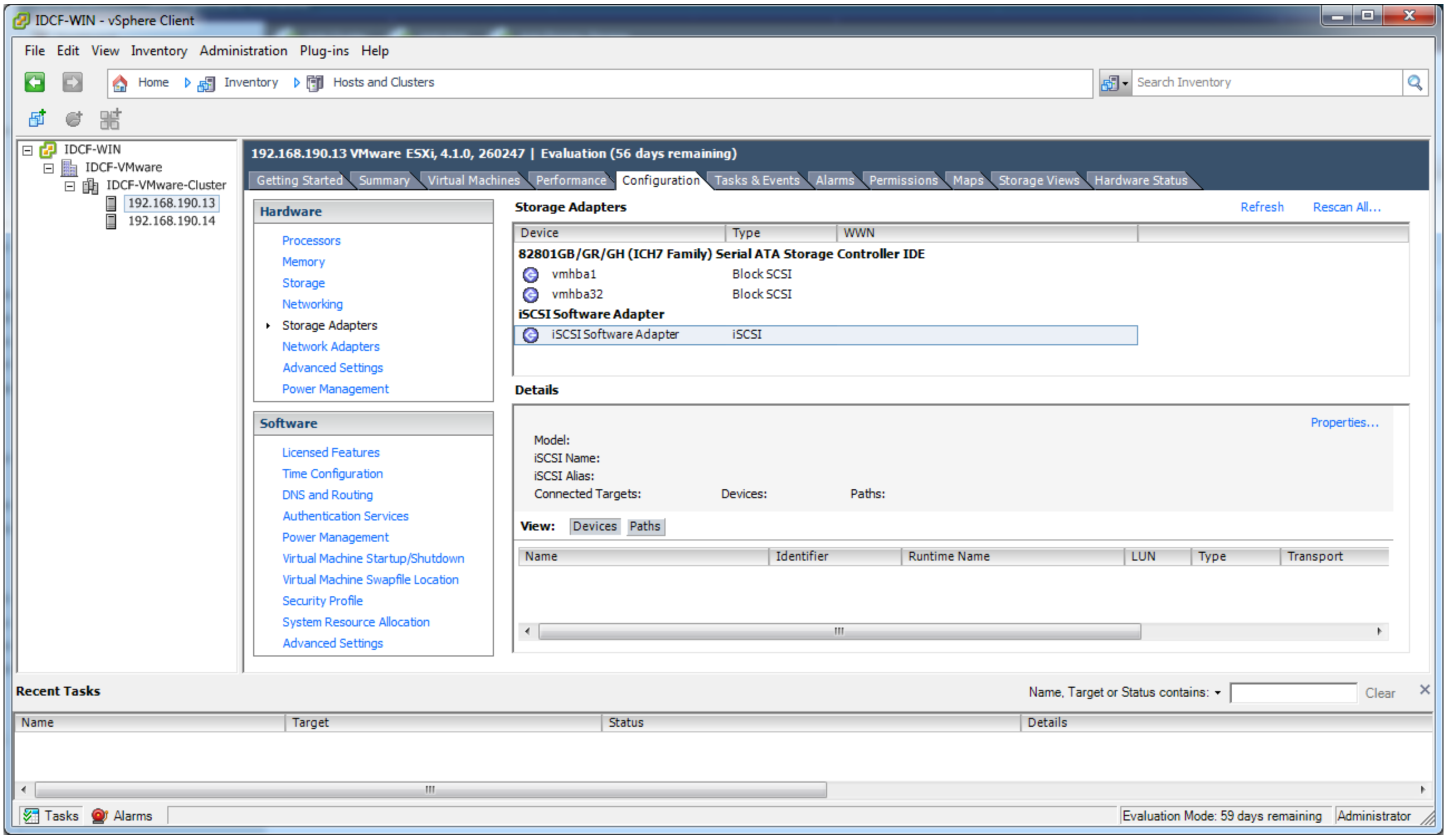
Storage Preparation for vSphere (iSCSI only)

Use of iSCSI requires preparatory work in vCenter. You must add an iSCSI target and create an iSCSI datastore.

If you are using NFS, skip this section.

Enable iSCSI initiator for ESXi hosts

1. In vCenter, go to Hosts and Clusters/Configuration, and click Storage Adapters link. You will see:



192.168.190.13 VMware ESXi, 4.1.0, 260247 | Evaluation (56 days remaining)

Storage Adapters

Device	Type	WWN
82801GB/GR/GH (ICH7 Family) Serial ATA Storage Controller IDE		
vmhba1	Block SCSI	
vmhba32	Block SCSI	
iSCSI Software Adapter		
iSCSI Software Adapter	iSCSI	

Details

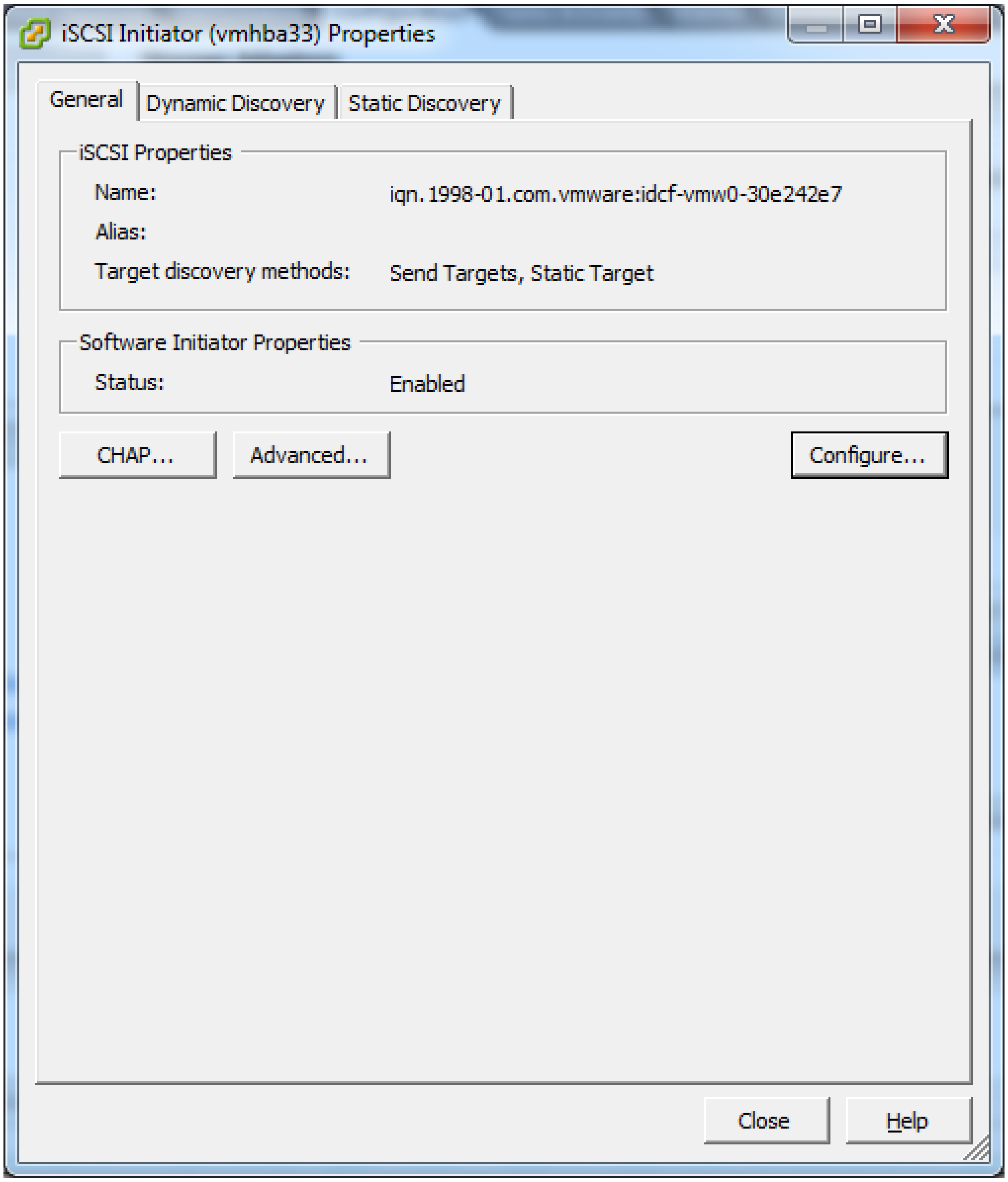
Model:
iSCSI Name:
iSCSI Alias:
Connected Targets: Devices: Paths:

View: Devices Paths

Name	Identifier	Runtime Name	LUN	Type	Transport

Name	Target	Status	Details

2. Select iSCSI software adapter and click Properties.



iSCSI Initiator (vmhba33) Properties

iSCSI Properties

Name: iqn.1998-01.com.vmware:idcf-vmw0-30e242e7

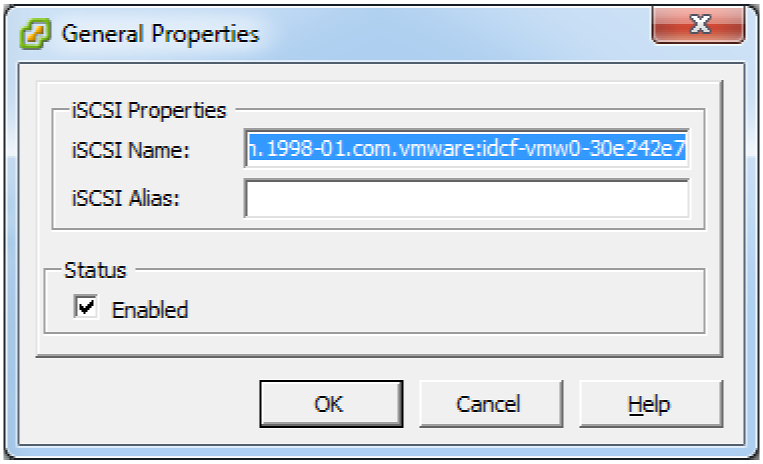
Alias:

Target discovery methods: Send Targets, Static Target

Software Initiator Properties

Status: Enabled

3. Click the Configure… button.



iSCSI Properties

iSCSI Name: n.1998-01.com.vmware:idcf-vmw0-30e242e7

iSCSI Alias:

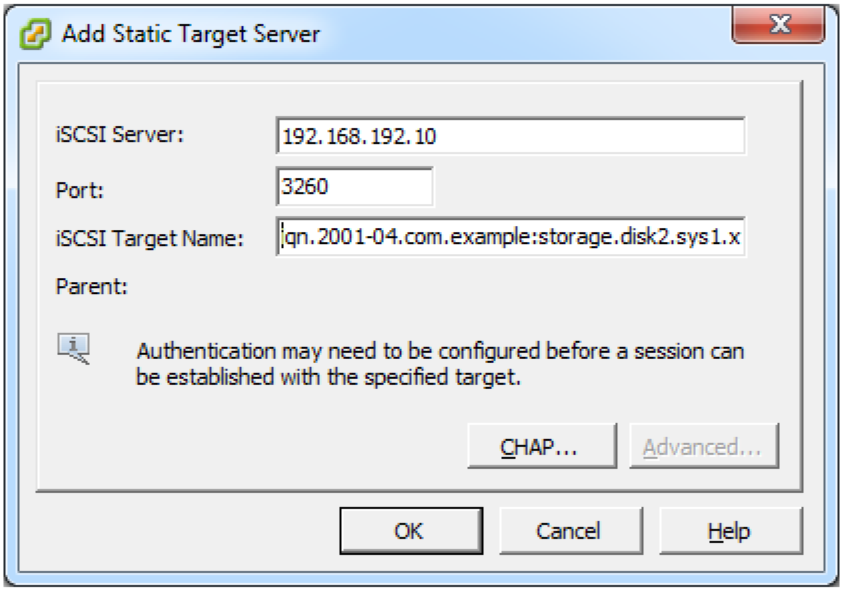
Status

☑ Enabled

4. Check Enabled to enable the initiator.
5. Click OK to save.

Add iSCSI target

Under the properties dialog, add the iSCSI target info:



iSCSI Server: 192.168.192.10

Port: 3260

iSCSI Target Name: iqn.2001-04.com.example:storage.disk2.sys1.x

Parent:

Authentication may need to be configured before a session can be established with the specified target.

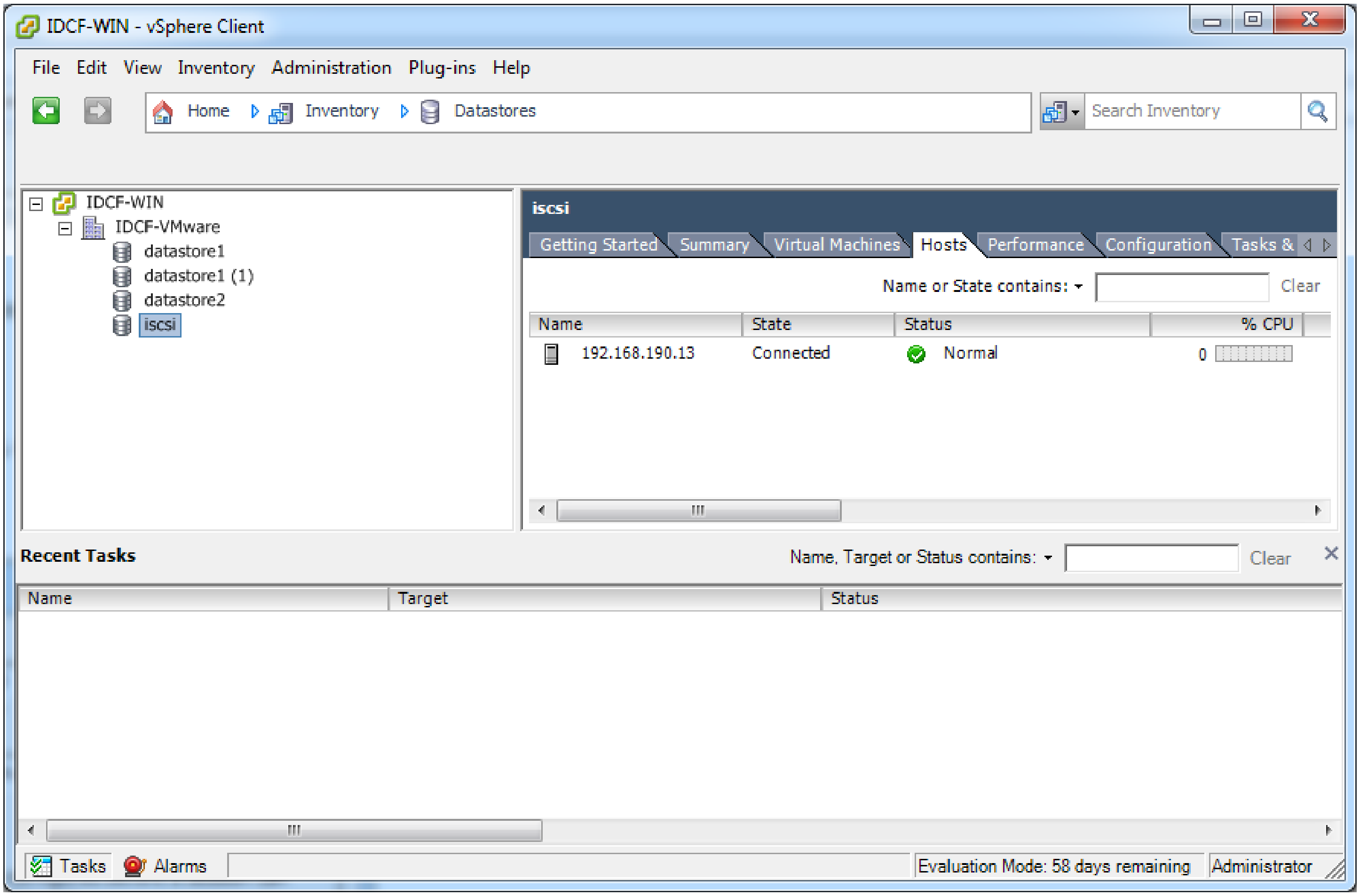
Repeat these steps for all ESXi hosts in the cluster.

Create an iSCSI datastore

You should now create a VMFS datastore. Follow these steps to do so:

1. Select Home/Inventory/Datastores.
2. Right click on the datacenter node.
3. Choose Add Datastore… command.
4. Follow the wizard to create a iSCSI datastore.

This procedure should be done on one host in the cluster. It is not necessary to do this on all hosts.



Name	State	Status	% CPU
192.168.190.13	Connected	Normal	0

Name	Target	Status

Multipathing for vSphere (Optional)

Storage multipathing on vSphere nodes may be done according to the vSphere installation guide.

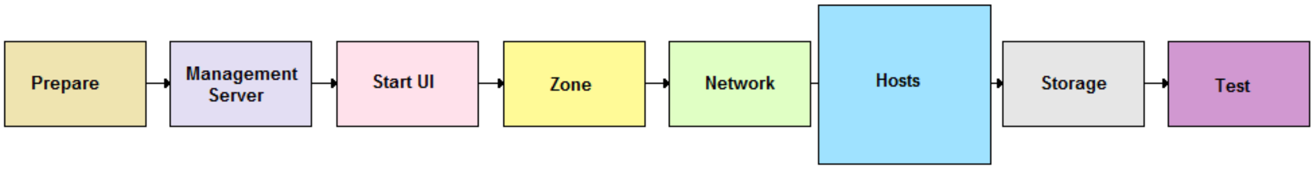
Add Hosts or Configure Clusters (vSphere)

Host management for vSphere is done through a combination of vCenter and the CloudStack admin UI. CloudStack requires that all hosts be in a CloudStack Cluster, but the Cluster may consist of a single host. As an administrator you must decide if you would like to use clusters of one host or of multiple hosts. Clusters of multiple hosts allow for features like live migration. Clusters also require shared storage such as NFS or iSCSI.

Creating vCenter Clusters

Use vCenter to create a vCenter cluster and add your desired hosts to the cluster. You will later add the entire cluster to CloudStack. (see Add Cluster: vSphere on page 40).

KVM Installation and Configuration



If you want to use the KVM hypervisor to run guest virtual machines, install KVMon the host(s) in your cloud. The material in this section doesn't duplicate KVM installation docs, but it does give some CloudStack-specific tweaks.

Supported Operating Systems

KVM is included with a variety of Linux-based operating systems. Those supported for use with CloudStack can be downloaded from the following websites and installed by following the Installation Guide provided with each operating system.

Officially supported OS version for KVM hosts:

- RHEL 6.0 or 6.1: https://access.redhat.com/downloads

The following are also available for community use. We do not guarantee access to CloudStack support personnel for users of these versions:

- RHEL versions 5.5 – 5.x, 6.0, and 6.1: https://access.redhat.com/downloads
- CentOS versions 5.5 – 5.x, 6.0, and 6.1: http://www.centos.org/modules/tinycontent/index.php?id=15
- Ubuntu 10.04: http://releases.ubuntu.com/lucid/
- Fedora 14: https://mirrors.fedoraproject.org/publiclist/Fedora/14/

System Requirements for KVM Hosts

- Must be certified as compatible with the selected operating system. For example, see the RHEL Hardware Compatibility Guide at https://hardware.redhat.com/.
- Must support HVM (Intel-VT or AMD-V enabled).
- Within a single cluster, the hosts must be homogenous. The CPUs must be of the same type, count, and feature flags.
- Within a single cluster, the hosts must be of the same kernel version. For example, if one Host is RHEL6 64 bit, they must all be RHEL6 64 bit.
- 64-bit x86 CPU (more cores results in better performance)
- Hardware virtualization support required
- 4 GB of memory
- 30 GB of local disk

- At least 1 NIC
- Statically allocated IP Address

KVM Installation Steps

1. Download KVM from one of the websites listed in Supported Operating Systems on page 80 and install it by following the Installation Guide provided with your chosen operating system.
2. Following installation, perform the following configuration tasks, which are described in the next few sections:

Required	Optional
Install the CloudStack agent on the host	Primary storage setup
Physical network configuration	

Installing the CloudStack Agent on a KVM Host

Each KVM Host must have the CloudStack Agent installed on it. Install the CloudStack Agent on each Host using the following steps. Some of the steps in the installation procedure apply only to hosts running certain operating systems; these are noted at the beginning of the step.

1. (RHEL 6.x/Fedora) Check for a fully qualified hostname.

```
# hostname --fqdn
```

   This should return a fully qualified hostname such as "kvm1.lab.example.org". If it does not edit /etc/hosts so that it does.

2. (RHEL 6.x) Remove qemu-kvm. CloudStack provides a patched version.

```
# yum erase qemu-kvm
```

3. (RHEL 6.x) If you do not have a Red Hat Network account, you need to prepare a local Yum repository.
   a. If you are working with a physical host, insert the RHEL 6.x installation CD. If you are using a VM, attach the RHEL6 ISO.
   b. Mount the CDROM to /media.
   c. Create a repo file at /etc/yum.repos.d/rhel6.repo. In the file, insert the following lines:

```
[rhel]
name=rhel6
baseurl=file:///media
enabled=1
gpgcheck=0
```

4. Install the CloudStack packages. You should have a file in the form of “CloudStack-NNNN.tar.gz”.

   Untar the file and then run the install.sh script inside it:

   On CentOS or RHEL, be sure you have downloaded the tar file for your version. Different files are provided for versions 6.0 and 6.1.

   ```
   # tar xzf CloudStack-VERSION-1-rhel6.2.tar.gz
   # cd CloudStack-VERSION-1-rhel6.2
   # ./install.sh
   Setting up the temporary repository...
   Cleaning Yum cache...
   Loaded plugins: fastestmirror
   11 metadata files removed
   Welcome to the CloudStack Installer.  What would you like to do?

       M) Install the Management Server
       A) Install the Agent
       B) Install BareMetal Agent
       S) Install the Usage Monitor
       D) Install the database server
       Q) Quit

       > A
   ```

   Choose “A” to install the Agent software.

5. (CentOS 6.0) When the agent installation is finished, log in to the host as root and run the following commands to start essential services:

   ```
   # service rpcbind start
   # service nfs start
   # chkconfig nfs on
   # chkconfig rpcbind on
   ```

6. On the KVM host, edit the file /etc/libvirt/qemu.conf file and make sure the line "vnc_listen = 0.0.0.0" is uncommented. If necessary, uncomment the line and restart /etc/init.d/libvirtd.

   ```
   # vi /etc/libvirt/qemu.conf
   # /etc/init.d/libvirtd restart
   ```

The CloudStack Agent is now installed.

Physical Network Configuration for KVM

You should have a plan for how the Hosts will be cabled and which physical NICs will carry what types of traffic. By default, CloudStack will use the device that is used for the default route. This device will be placed in a CloudStack-created bridge.

If a system has multiple NICs or bonding is desired, the admin may configure the networking on the host. The admin must create a bridge and place the desired device into the bridge. This may be done for each of the public network and the private network. Then edit /etc/cloud/agent/agent.properties and add values for the following:

- public.network.device
- private.network.device

These should be set to the name of the bridge that the user created for the respective traffic type. For example:

- public.network.device=publicbondbr0

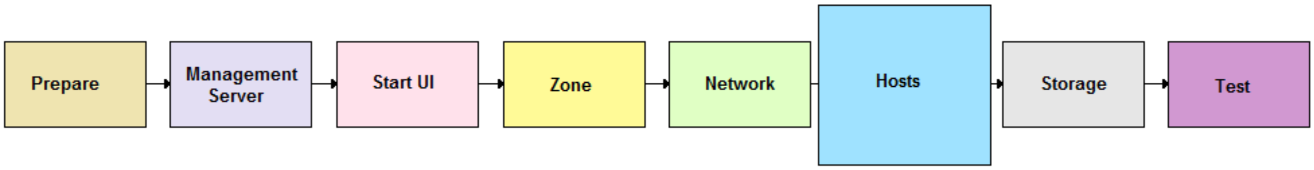
This should be done after the install of the software as described previously.

Primary Storage Setup for KVM (Optional)

CloudStack allows administrators to set up shared Primary Storage that uses iSCSI or fiber channel. With KVM, the storage is mounted on each Host, and the storage is based on some clustered file system technology like OCFS2. This is called "SharedMountPoint" storage and is an alternative to NFS. With SharedMountPoint storage:

- Each node in the KVM cluster mounts the storage in the same local location (e.g., /mnt/primary)
- A shared clustered file system is used
- The administrator manages the mounting and unmounting of the storage
- If you want to use SharedMountPoint storage you should set it up on the KVM hosts now. Note the mountpoint that you have used on each host; you will use that later to configure CloudStack.

Oracle VM (OVM) Installation and Configuration



If you want to use the Oracle VM Server (OVM) hypervisor to run guest virtual machines, install OVM on the host(s) in your cloud.

System Requirements for OVM Hosts

CloudStack works with the following version:

- OVM Server 2.3

The OVM hosts must follow these restrictions:

- All hosts must be 64-bit and must support HVM (Intel-VT or AMD-V enabled). All Hosts within a Cluster must be homogenous. That means the CPUs must be of the same type, count, and feature flags.
- Within a single cluster, the hosts must be of the same kernel version. For example, if one Host is OVM 2.3 64 bit, they must all be OVM 2.3 64 bit.

OVM Installation Overview

Certain essential CloudStack software components can not run on OVM, so your OVM Zone will need to include at least two clusters: one cluster containing the OVM hosts, and another cluster with a different hypervisor (KVM, XenServer, or VMWare), where the CloudStack system VMs will run.

Installing OVM on the Host(s)

1. Download the OVM template from the Oracle website (http://www.oracle.com/virtualization) and install it using the OVM Installation Guide. The software download should be a .zip file that contains two files, an image (.img) file and vm.cfg. You need only the .img file. The default template password is ovsroot.
2. Unzip the file and copy the .img file to your HTTP server.
3. Follow the instructions in the OVM Installation Guide to install OVM on each host. During installation, you will be prompted to set an agent password and a root password. You can specify any desired text or accept the default.

   Make a note of these passwords – you will need them later.
4. Repeat for any additional hosts that will be part of the OVM cluster.

NOTE: After ISO installation, the installer reboots into the operating system. Due to a known issue in OVM Server, the reboot will place the VM in the Stopped state. In the CloudStack UI, detach the ISO from the VM (so that the VM will not boot from the ISO again), then click the Start button to restart the VM.

Primary Storage Setup for OVM

CloudStack natively supports NFS, iSCSI and local storage. Each iSCSI LUN can be assigned to exactly one OVM cluster as the cluster's primary storage device. Following is a summary of the steps that you need to do. For details, see Oracle documentation on preparing storage repositories at http://download.oracle.com/docs/cd/E15458_01/doc.22/e15444/storage.htm#sthref65.

1. Map your iSCSI device to the OVM host's local device. The exact steps to use depend on your system's peculiarities.
2. On every host in the cluster, create the same softlink name so CloudStack can use a consistent path to refer to the iSCSI LUN from any host. For example, if the softlink name is /dev/ovm-iscsi0:

   ```
   ln -s /dev/disk/by-path/<output of previous command> /dev/ovm-iscsi0
   ```

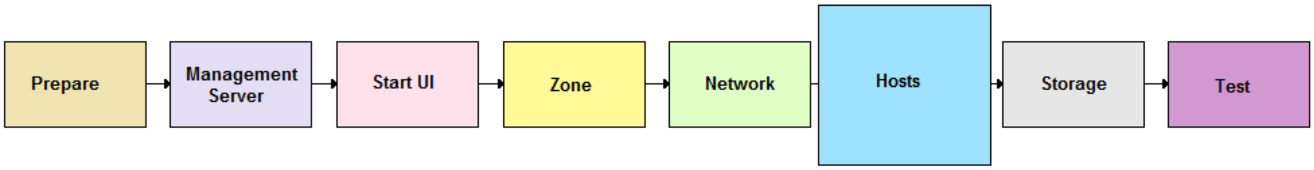
   Make a note of your softlink name. You will need it later.
3. Exactly once on any ONE host in the OVM cluster, format the OCFS2 file system on the iSCSI device.

Set Up Host(s) for System VMs

Before proceeding to install the CloudStack Management Server, you need to install a non-OVM hypervisor on at least one host that will run the CloudStack System VMs (which are not supported by OVM).

1. Install the non-OVM hypervisor on at least one host by following one of the instructions below, depending on which hypervisor you want to use:
   - Installing Citrix XenServer on page 55
   - VMware vSphere Installation and Configuration on page 63
   - KVM Installation and Configuration on page 80
2. When you set up the pod that will contain the OVM cluster, remember to include this non-OVM host in its own cluster along with the OVM cluster in the same pod.

Bare Metal Installation



You can set up bare metal hosts in a CloudStack cloud and manage them with the Management Server. Bare metal hosts do not run hypervisor software. You do not install the operating system – that is done using PXE when an instance is created from the bare metal template (which you are going to create as part of this Installation procedure). Bare metal hosts use direct untagged networking. A cloud can contain a mix of bare metal instances and virtual machine instances.

Bare Metal Concepts

In a traditional IaaS cloud, hypervisors are manually provisioned and added to IaaS infrastructure as hosts. These hosts (hypervisors) are used as resources (CPU/memory) to deploy virtual images (user work loads). Some of the heavy workloads require an entire physical resource to be exposed by these physical machines. In such scenarios, it is efficient to deploy the OS image instead of a virtual image on hypervisor. Some enterprise-class software runs more efficiently on bare metal hardware than on virtualized systems. Bare-metal provisioning feature in CloudStack addresses these use cases. IaaS clouds built using CloudStack can service and manage not only virtual hosts but also physical hosts with all the benefits of self-service, elastic and multi-tenant features.

Bare Metal Architecture

In order to explain the architecture of bare metal provisioning in CloudStack, it is important to understand how virtual machines (VMs) are provisioned within CloudStack. CloudStack provisions CPU/memory, storage, and network. CPU/Memory are resources provided by hosts. Hosts are physical machines with hypervisors installed on them. These hosts are in a logical component called a cluster. Each cluster has one are more hosts that are homogeneous. One or more clusters form a pod and one or more pods form a zone. The storage resources are where CloudStack allows users to manage templates and ISOs. VMs are created from these templates (based on the service offering) and provisioned on one of the hosts (based on allocation algorithms) in a given cluster.

Bare metal provisioning works in a similar way, with a few differences.

Bare metal hosts can be set up and managed in the same cloud as virtual instances. Bare metal hosts do not run hypervisor software. It is not required to have an operating system installed on the machine in advance. The operating system is provisioned to the physical host using PXE Boot when a host is provisioned from the bare metal template. In the current version of CloudStack, bare metal provisioning is supported only with the direct untagged network model.

Primary technologies that are part of the bare metal feature are IPMI (Intelligent Platform Management Interface), PXE (pre-boot execution environment), DHCP, CIFS and PING (Partimage Is Not Ghost). It is important to understand these technologies to understand the architecture of this CloudStack feature. It is beyond the scope of this document to discuss those technologies in detail.

How Does Bare Metal Provisioning Work?

Bare metal machines are provisioned with PING images (templates) using PXE boot protocol. CloudStack requires bare metal templates in PING format. To create the templates, install the OS on identical hardware and use PING tools. These images are stored on a CIFS server where the PXE server can access them.

The following events take place when a user requests to provision a bare metal host specifying a template and service offering in a pod:

1. CloudStack programs the PXE server with the host MAC address, host IP address, and boot image file based on the bare metal template the user has chosen.
2. CloudStack programs the DHCP server with the MAC address and IP.
3. CloudStack enables PXE boot on the host and powers it on using IPMI interface.
4. The host broadcasts a DHCP request and receives a reply from the DHCP server. This reply consists of an IP address for the host, PXE boot server address, and a pointer to the boot image file.
5. The host then contacts the PXE boot server and downloads the image file using TFTP protocol. The image file is a live kernel and initrd with PING software installed.
6. The host starts the boot process using the downloaded file from the TFTP server.
7. After these steps complete successfully, the host is ready for the workload.

Bare Metal Deployment Architecture

The deployment architecture for bare metal provisioning is similar to CloudStack deployment with the direct untagged network model, with the addition of PXE and DHCP servers in every pod. These servers can reside on the same machine. The PXE server needs access to a CIFS server that can reside on any routable network.

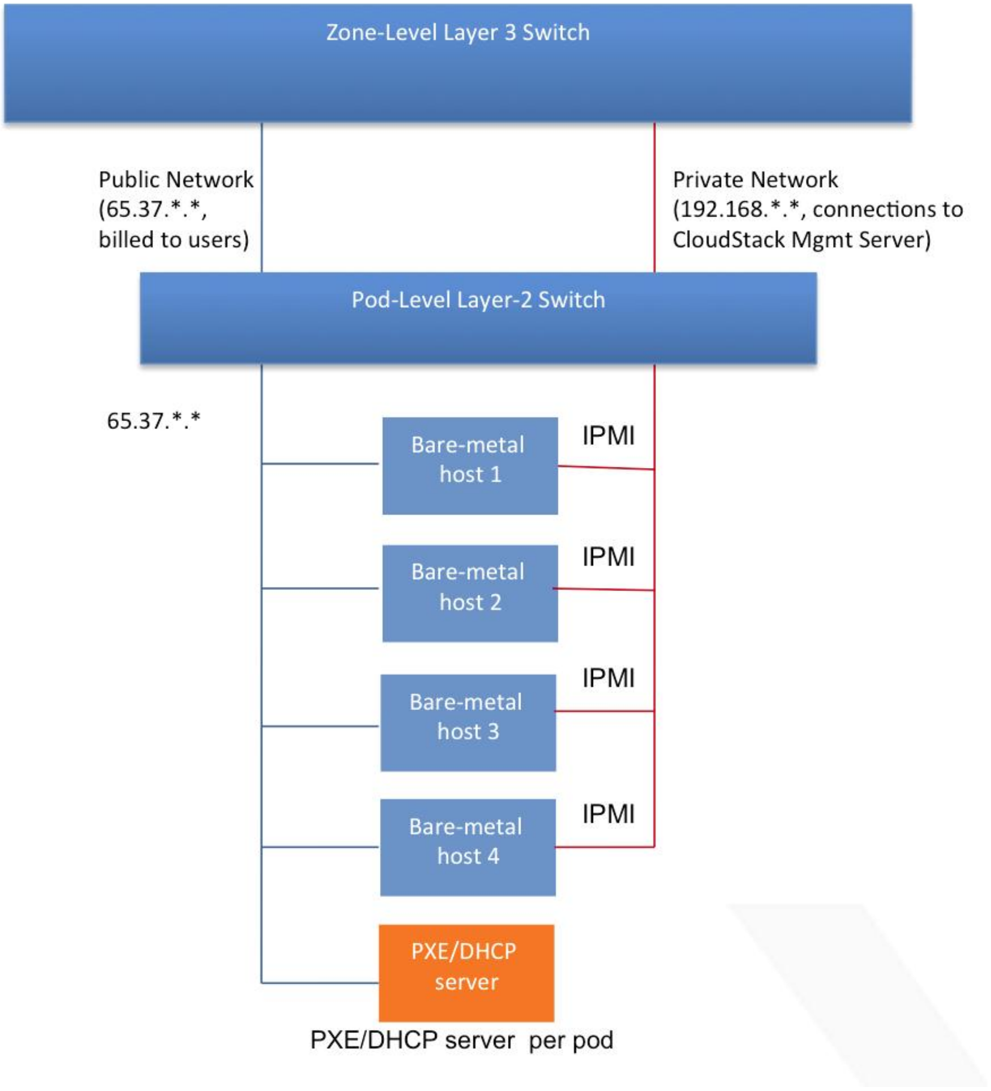


Figure 3 Pod with Bare Metal Hosts

Bare Metal Installation Checklist

Prepare to add bare metal instances to your cloud by performing the setup procedures in the next few sections.

1. Install the Management Server (either single-node, p. 21, or multi-node, p. 26) ❑
2. Set Up the Firewall for Direct Untagged Networking (p. 89) ❑
3. Set Up IPMI (p. 92) ❑
4. Enable PXE on the Bare Metal Host (p. 93) ❑
5. Install the PXE and DHCP Servers (p. 93) ❑
6. Set Up a CIFS File Server (p. 94) ❑
7. Create a Bare Metal Image (p. 94) ❑
8. Add the PXE Server and DHCP Server to Your Deployment (p. 94) ❑
9. Add a Cluster, Host, and Firewall (p. 95) ❑
10. Add a Service Offering and Template (p. 95) ❑

Set Up the Firewall for Direct Untagged Networking

Bare metal uses direct untagged networking. If you are using a firewall, you must set it up for this type of networking. This section gives the instructions for setting up a Juniper SRX for direct untagged networking. For more about adding the Juniper to your network, see External Guest Firewall Integration for Juniper (optional) on page 109.

Who Should Read This

If you are using a firewall, you need to follow the steps in this section. If you are not using a firewall, skip to Set Up IPMI on page 92.

1. Connect to the SRX as root.
   - It is recommended to connect using the serial port to ensure a reliable connection.
   - Connect with a 9600 baud rate.
2. Clear out factory default settings.
   a. Delete existing interfaces.

```
# delete interfaces
```

   b. Delete VLANs.

```
delete vlans
```

   c. Delete firewall rules.

```
delete firewall
```

   d. Delete NAT rules.

```
delete security nat
```

e. Delete security zones.

```
delete security zones
```

f. Make sure the initial security policies are correct.

```
# show security policies
```

The output should look like this:

```
from-zone trust to-zone untrust {
   policy trust-to-untrust {
      match {
         source-address any;
         destination-address any;
         application any;
      }
      then {
         permit;
      }
   }
}
```

If you do not see the above output, follow these steps to re-create the base security policy.

```
# delete security policies

# set security policies from-zone trust to-zone untrust policy trust-to-untrust
match source-address any destination-address any application any

# set security policies from-zone trust to-zone untrust policy trust-to-untrust then
permit
```

3. Set up interfaces, security zones, and the default route.

a. Determine which SRX ports you are using. There should be one private interface and one public interface. For example, ge-0/0/1 might be private, and ge-0/0/2 might be public.

b. Program the interfaces with correct IP addresses. Run the following command once for each interface, substituting your own values.

```
# set interfaces <name> unit 0 family inet address <address>/<subnet size>
```

For example:

```
# set interfaces ge-0/0/1 unit 0 family inet address 192.168.240.1/24
# set interfaces ge-0/0/2 unit 0 family inet address 192.168.1.91/24
```

c. Create two security zones: one called "trust" (private zone) and the other called "untrust" (public zone).

```
# set security zones security-zone trust
# set security zones security-zone untrust
```

d. Add the private interface to the trust zone and the public interface to the untrust zone. Run the following command once for each interface, substituting your own values.

```
# set security zones security-zone <zone> interfaces <name>
```

For example:

```
# set security zones security-zone trust interfaces ge-0/0/1
# set security zones security-zone untrust interfaces ge-0/0/2
```

e. Allow host inbound traffic on both security zones.

```
# set security zones security-zone trust host-inbound-traffic system-services all
# set security zones security-zone trust host-inbound-traffic protocols all
# set security zones security-zone untrust host-inbound-traffic system-services all
# set security zones security-zone untrust host-inbound-traffic protocols all
```

f. Create a default route. This can be through the public interface. Substitute your own values for the IP addresses in this example.

```
# set routing-options static route 0.0.0.0/0 next-hop 192.168.1.1 install
```

g. Verify your interface setup. Your IP addresses should appear instead of our examples.

```
# show interfaces
  ge-0/0/1 {
     unit 0 {
        family inet {
           address 192.168.240.1/24;
        }
     }
  }
  ge-0/0/2 {
     unit 0 {
        family inet {
           address 192.168.1.91/24;
        }
     }
  }
```

h. Verify your zone setup. Your interface names should appear instead of our examples.

```
# show security zones
  security-zone trust {
     host-inbound-traffic {
        system-services {
           all;
        }
        protocols {
           all;
        }
     }
     interfaces {
        ge-0/0/1;
     }
  }
  security-zone untrust {
     host-inbound-traffic {
        system-services {
           all;
        }
        protocols {
           all;
        }
     }
     interfaces {
        ge-0/0/2;
     }
  }
```

i. Verify your route. Your IP addresses should appear instead of our examples.

```
# show routing-options
  static {
     route 0.0.0.0/0 {
        next-hop 192.168.1.1;
        install;
     }
  }
```

4. Set up system services and a root login.

   a. Allow CloudStack to program the SRX through the API.

```
# set system services xnm-clear-text
```

   b. Set a root password for the system. You will be prompted to enter and re-enter a password.

```
# set system root-authentication plain-text-password
```

5. Commit your changes.

```
# commit
```

   Wait for confirmation.

Set Up IPMI

The procedure to access IPMI settings varies depending on the type of hardware. Consult your manufacturer's documentation if you do not already know how to display the IPMI settings screen.

Once you are there, set the following:

- IP address of IPMI NIC
- Netmask
- Gateway
- Username and password for IPMI NIC

Enable PXE on the Bare Metal Host

The bare metal host needs to use PXE to boot over the network. Access the BIOS setup screen (or equivalent for your hardware) and do the following:

- Set hard disk as the first priority device in the boot order.
- Make sure the connected NIC on the bare metal machine is PXE-enabled.
- Make a note of the MAC address of the PXE-enabled NIC. You will need it later.

Install the PXE and DHCP Servers

Each bare metal host must be able to reach a PXE server and a DHCP server. Use the following steps to install the needed servers on another machine (or virtual machine). All commands should be run as root.

1. Log in as root to a host or virtual machine running CentOS 5.3, 5.4, or 5.5.
2. You should have access to a file in the form of “CloudStack-NNNN.tar.gz." Copy that file to the machine.
3. Untar the file and then run the install.sh script inside it:

```
# tar xzf CloudStack-VERSION-1-rhel6.2.tar.gz
# cd CloudStack-VERSION-1-rhel6.2
# ./install.sh
Setting up the temporary repository...
Cleaning Yum cache...
Loaded plugins: fastestmirror
11 metadata files removed
Welcome to the CloudStack Installer.  What would you like to do?

    M) Install the Management Server
    A) Install the Agent
    B) Install BareMetal Agent
    S) Install the Usage Monitor
    D) Install the database server
    Q) Quit

    > B
```

   Choose “B” to install the software that is needed for bare metal.

4. Run the bare metal setup script.

```
# cloud-setup-baremetal
```

5. Make note of the TFTP root directory that is displayed by this script. You will need it later.

Set Up a CIFS File Server

The bare metal image will be stored on a file server. We recommend using a Windows machine with its built-in CIFS file sharing functionality. If you prefer to use Linux as the file server, use the Samba Windows interoperability suite. Set up the CIFS file server using these steps:

1. On the file server, set up the following directory structure: Share\Ping_Backup. Share is the folder that will be the CIFS root directory, and Ping_Backup will store images created with the Partimage Is Not Ghost (PING) tool.
2. Share the root directory. On Windows, right-click the Share folder and choose Share with… .

Create a Bare Metal Image

Create an image which can be installed on bare metal hosts later, when bare metal instances are added to the cloud. To create the image, you will be using the Partimage Is Not Ghost (PING) tool. For information about how to use PING, see http://ping.windowsdream.com/ping/howto-2.01.html.

1. Install the desired OS on a machine with hardware identical to what you intend to use for the bare metal machines.

   Be sure the hardware is identical.
2. Use PING to create an image and store it in the Share\Ping_Backup directory on the CIFS file server you have set up.
   a. PING will prompt you for the storage location. Use <IP of CIFS server>\Share\Ping_Backup.
   b. PING will prompt you for a name for the image. Give any name you find helpful, such as win7_64bit.

Add the PXE Server and DHCP Server to Your Deployment

As part of describing your deployment to CloudStack, you will need to add the PXE server and DHCP server that you created in Install the PXE and DHCP Servers on page 93.

1. Add a zone and pod using the Management Server UI. When creating the zone, in network type, choose Basic.

   Follow the steps in Add a Zone on page 33.
2. In the left navigation tree, choose System, then Physical Resources.
3. Select your Pod.
4. Click Add Network Device. Input the following and click Save:
   - **Type:** PxeServer
   - **URL:** IP of PXE server
   - **Username:** username of PXE server
   - **Password:** password of PXE server
   - Pxe Server Type: PING

- **PING Storage IP:** IP of CIFS server
- **PING Directory:** Share/Ping_Backup
- **TFTP Directory:** The directory displayed by cloud-setup-baremetal earlier. For example, /tftpboot.
- **PING CIFS Username:** username of CIFS server (optional)
- **PING CIFS Password:** password of CIFS server (optional)

5. Click Add Network Device again. Input the following and click Save:
    - **Type:** ExternalDhcp.
    - **URL:** IP of DHCP server. This is the same value you used for PXE server in the previous step. The DHCP and PXE servers must both be installed on one machine, and the install.sh script does this (see Install the PXE and DHCP Servers on page 93).
    - **Username:** username of DHCP server
    - **Password:** password of DHCP server
    - DHCP Server Type: Dhcpd

Add a Cluster, Host, and Firewall

Add the following entities to CloudStack using the Management Server UI.

1. Add a bare metal cluster as described in Add Cluster: Bare Metal on page 42, then return here for the next step.
2. Add one or more bare metal hosts as described in Add Hosts (Bare Metal) on page 45, then return here for the next step.
3. Add the firewall as described in **Advanced Networking: Adding an External Firewall (Optional)** on page 122. Then continue to the next section, Add a Service Offering and Template.

Add a Service Offering and Template

This is the final phase of a bare metal installation and deployment.

1. Create a bare metal service offering. In the Management Server UI, click Configuration – Service Offering – Add Service Offering. In the dialog box, fill in these values:
    - **Name.** Any desired name for the service offering.
    - **Display.** Any desired display text.
    - Storage Type. Shared.
    - **# of CPU Cores.** Use the same value as when you added the host.
    - **CPU (in MHZ).** Use the same value as when you added the host.
    - **Memory (in MB).** Use the same value as when you added the host.
    - Offer HA? No.
    - **Tags.** large

- Public? Yes.

2. Add a bare metal template as described in Creating a Bare Metal Template in the Administrator's Guide.

Your bare metal installation is complete! Now you can create a bare metal instance from the Instances screen of the UI.

If you want to allow inbound network traffic to the bare metal instances through public IPs, set up public IPs and a port forwarding rules. Follow the steps in How to Set Up Port Forwarding in the Administrator's Guide.

Design Considerations

Choosing a Deployment Architecture

The architecture used in a deployment will vary depending on the size and purpose of the deployment. This section contains examples of deployment architecture, including a small-scale deployment useful for test and trial deployments and a fully-redundant large-scale setup for production deployments.

Who Should Read This

If you need help figuring out how many nodes to include, how they fit together, how to scale your deployment, or how the various parts of a CloudStack work together in different scenarios, this section is for you.

Small-Scale Deployment

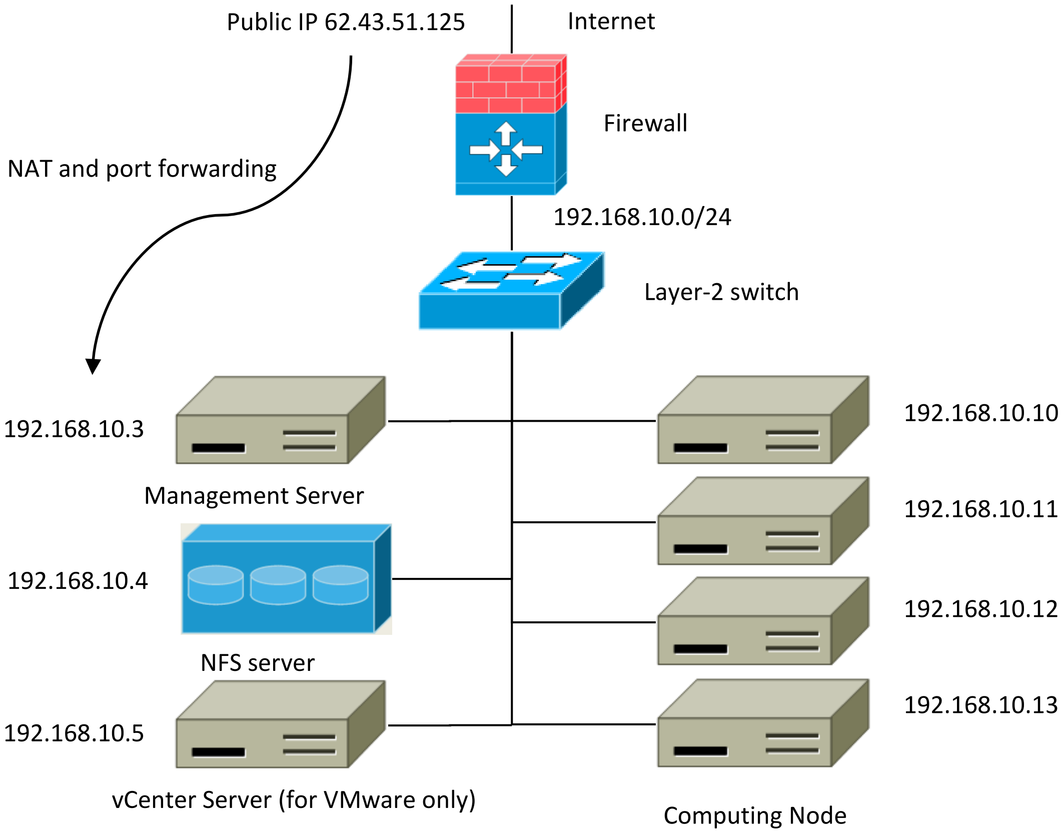


Figure 4 Small-Scale Deployment

Figure 1 illustrates the network architecture of a small-scale CloudStack deployment.

- A firewall provides a connection to the Internet. The firewall is configured in NAT mode. The firewall forwards HTTP requests and API calls from the Internet to the Management Server. The Management Server resides on the private network.

- A layer-2 switch connects all physical servers and storage.
- A single NFS server functions as both the primary and secondary storage.
- The Management Server is connected to the private network.

Large-Scale Redundant Setup

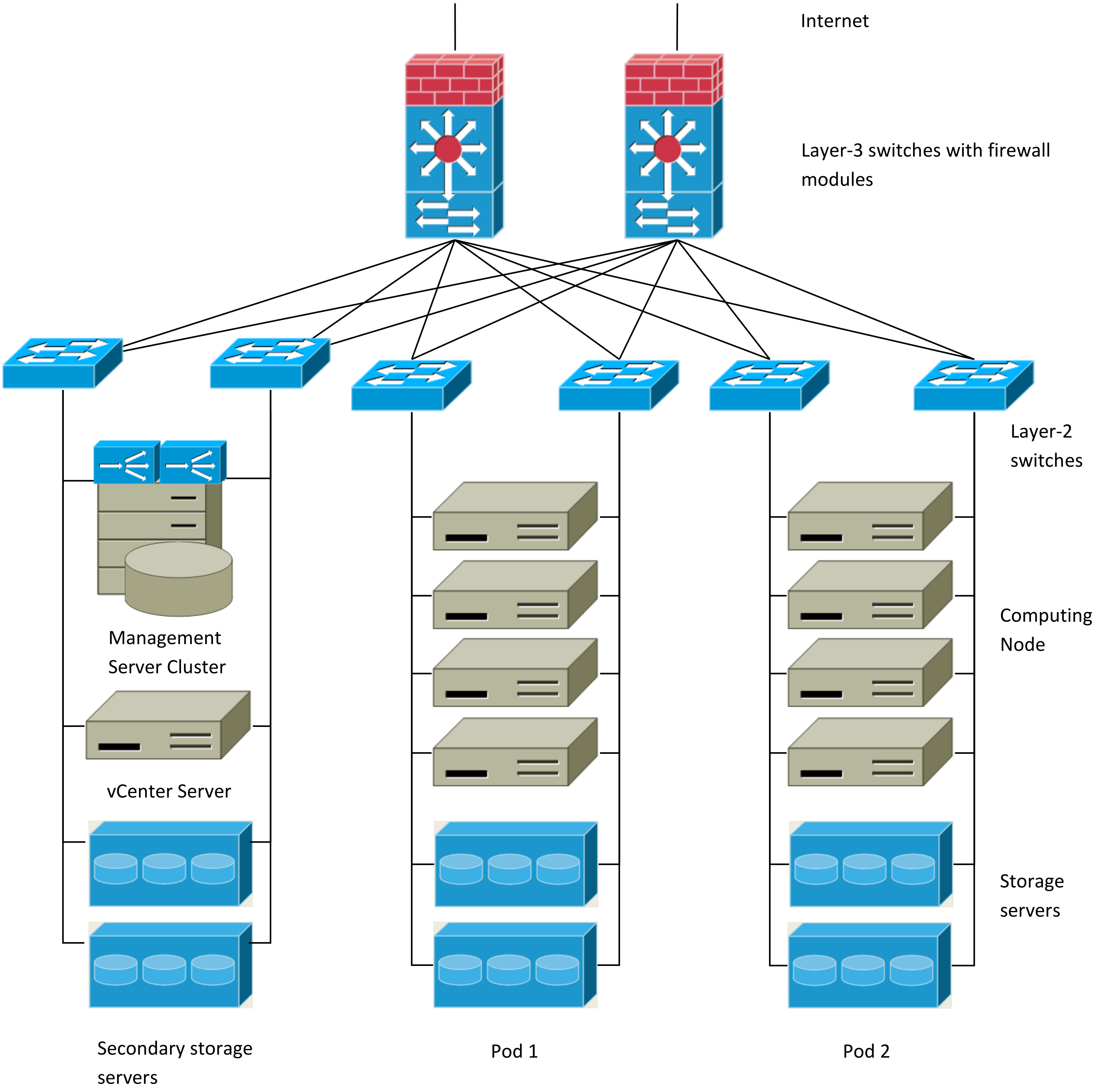

Internet
Layer-3 switches with firewall modules
Layer-2 switches
Management Server Cluster
vCenter Server
Computing Node
Storage servers
Secondary storage servers
Pod 1
Pod 2

Figure 5 Large-Scale Deployment Architecture

Figure 5 illustrates the network architecture of a large-scale CloudStack deployment.

- A layer-3 switching layer is at the core of the data center. A router redundancy protocol like VRRP should be deployed. Typically high-end core switches also include firewall modules. Separate firewall appliances may also be used if the layer-3 switch does not have integrated firewall capabilities. The firewalls are configured in NAT mode. The firewalls provide the following functions:
- Forwards HTTP requests and API calls from the Internet to the Management Server. The Management Server resides on the private network.
- When the cloud spans multiple availability Zones, the firewalls should enable site-to-site VPN such that servers in different availability Zones can directly reach each other.
- A layer-2 access switch layer is established for each Pod. Multiple switches can be stacked to increase port count. In either case, redundant pairs of layer-2 switches should be deployed.
- The Management Server cluster (including front-end load balancers, Management Server nodes, and the MySQL database) is connected to the private network through a pair of load balancers.
- Secondary storage servers are connected to the private network.
- Each Pod contains storage and computing servers. Each storage and computing server should have redundant NICs connected to separate layer-2 access switches.

Separate Storage Network

In the Large-Scale Redundant setup described in the previous section, storage traffic can overload the private network. A separate storage network is optional for deployments. Storage protocols such as iSCSI are sensitive to network delays. A separate storage network ensures guest network traffic contention does not impact storage performance.

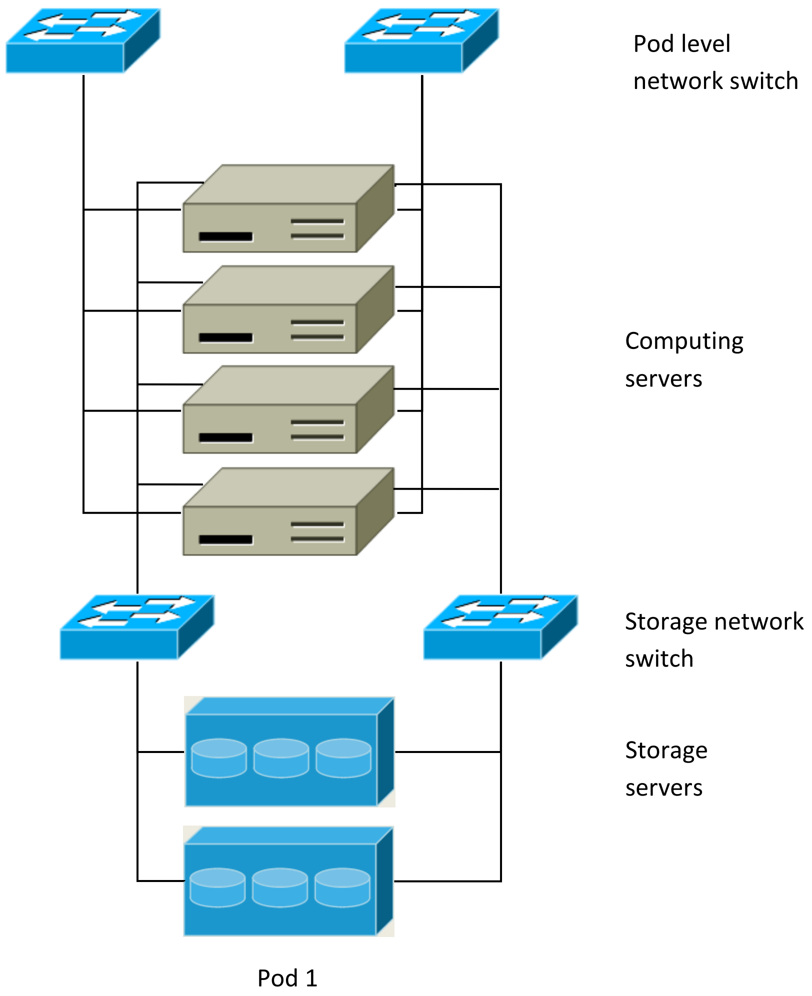


Figure 6 Separate Storage Network

Figure 6 illustrates a setup with a separate storage network. Each server has four NICs, two connected to Pod-level network switches and two connected to storage network switches.

There are two ways to configure the storage network:

- Bonded NIC and redundant switches can be deployed for NFS. In NFS deployments, redundant switches and bonded NICs still result in one network (one CIDR block+ default gateway address).
- iSCSI can take advantage of two separate storage networks (two CIDR blocks each with its own default gateway). Multipath iSCSI client can failover and load balance between separate storage networks.

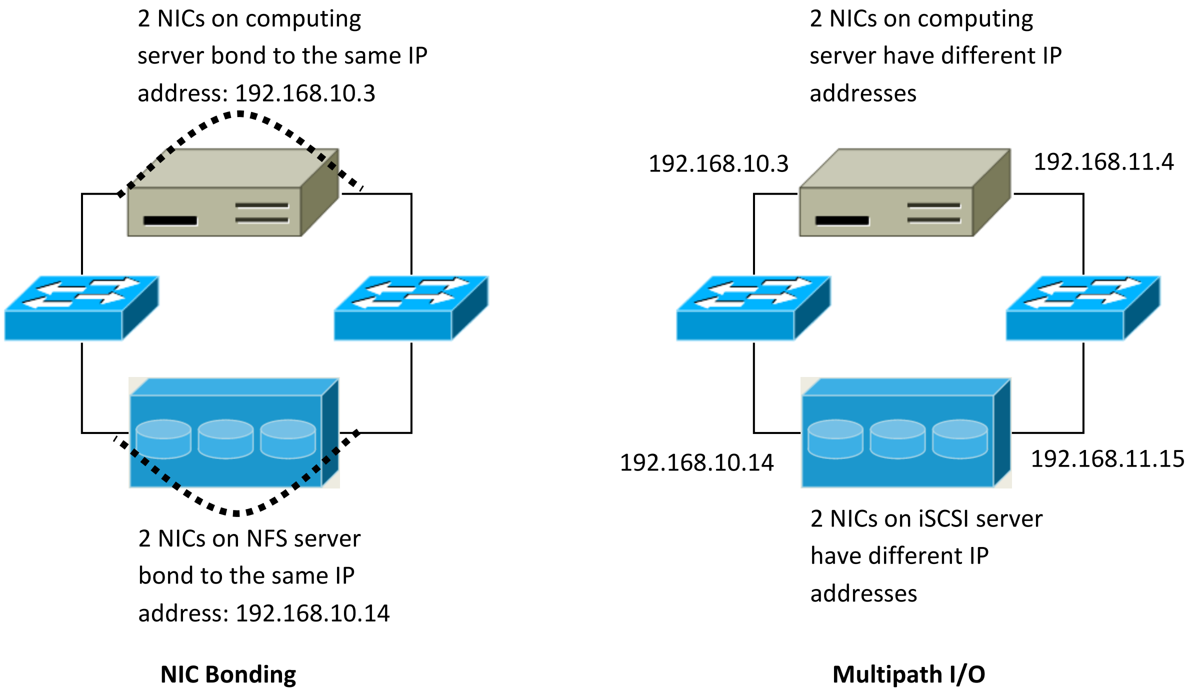

2 NICs on computing server bond to the same IP address: 192.168.10.3
2 NICs on NFS server bond to the same IP address: 192.168.10.14
NIC Bonding
2 NICs on computing server have different IP addresses
192.168.10.3
192.168.11.4
192.168.10.14
192.168.11.15
2 NICs on iSCSI server have different IP addresses
Multipath I/O

Figure 7 NIC Bonding and Multipath I/O

Figure 7 illustrates the differences between NIC bonding and Multipath I/O (MPIO). NIC bonding configuration involves only one network. MPIO involves two separate networks.

Network Setup

CloudStack provides two primary networking modes:

- Advanced. Advanced mode uses VLANs for isolation and consists of two types of guest networks:
- Virtual networks
- Direct attached networks with tagged VLANs ("Direct tagged")
- Basic. Basic mode does not use VLANs for isolation. It is also known as "Direct Untagged".

The following table highlights the differences between the three types of guest networks.

Network Attribute	Basic	Advanced - Virtual	Advanced - Direct
Uses VLANs for Isolation	No	Yes	Yes
Uses untagged VLAN	Yes	No	No
Uses Security Groups for Isolation (KVM only)	Yes	No	No
Virtual Router is Gateway	No	Yes	No
VPN available	No	Yes	No
Load Balancer available	No	Yes	No
DHCP and DNS available	Yes	Yes	Yes
1:1 NAT available	No	Yes	No
Public IP Addresses Required	No	Yes	No
Metering data available	No	Yes	No

The three types of networking may be in use in the same cloud. However, a given Zone must use either Basic Networking or Advanced Networking.

The style of networking that a particular guest receives is determined by a combination of the networks that the administrator has made available and the network that is chosen by the user at guest creation. An administrator configures Basic or Advanced Networking for a Zone. In Advanced networking, the administrator can add VLANs to CloudStack to create one or more networks that are available for use by the users of that Zone. Virtual networks use "Zone VLANs" and direct tagged networks use "Direct VLANs".

This chapter discusses network setup that is common to all modes as well as network setup that is specific to one or two of the modes.

VLAN Setup with Basic Networking

When Basic Networking is used in a Zone, all guests allocated in that Zone share a single untagged VLAN. The network interface in the host that is connected to this untagged VLAN must be named cloud-guest. For example, in XenServer, the network name-label must be “cloud-guest”.

VLAN Allocation with Advanced Networking

CloudStack networking enables many different deployment styles. Your deployment may not need one or more of the types of VLANs. You should decide what services you want to offer to your users before provisioning VLANs.

Warning

The Zone VLAN range must be configured before any user VMs are started.

VLAN allocation must be considered if Advanced Mode is used for a Zone. CloudStack is designed to utilize hardware VLANs to isolate guest virtual network traffic. There are three types of VLANs in CloudStack.

1. **Public VLAN**. A range of VLAN IDs will be reserved for public IP addresses. These VLANs are trunked into every Pod.
2. **Zone VLAN**. A range of VLAN IDs will be reserved for guest virtual networks. These VLANs are trunked into every Pod. One VLAN is allocated per guest virtual network that has active instances.
3. **Direct VLAN.** A range of VLAN IDs will be reserved for direct tagged networks. These networks may either be Zone wide, meaning guests from multiple accounts can use them, or account-specific, where the entire VLAN is reserved for use by a single account. These VLANs are trunked into every Pod. The administrator provisions these VLANs in CloudStack one at a time; a range is not given to CloudStack to manage as is the case with Zone VLANs.

Figure 5 illustrates VLAN allocation in an Availability Zone:

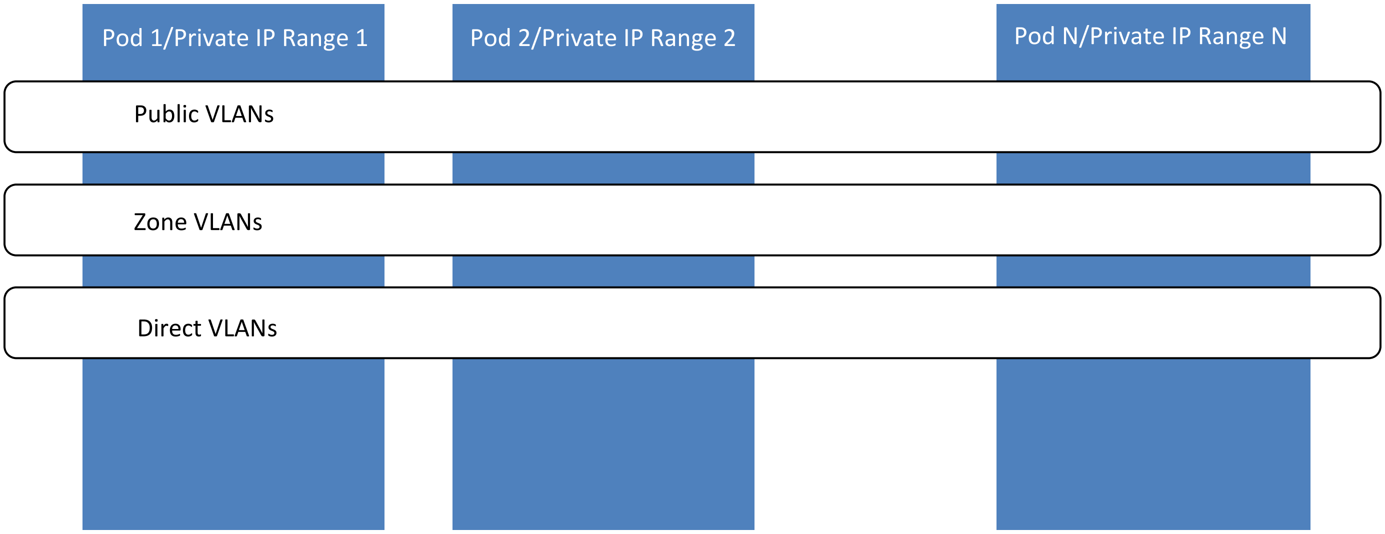


Figure 5 VLAN Allocation in an Availability Zone

The hosts and storage servers reside on an untagged private network. The untagged private network in each Pod is mapped to a unique VLAN and private IP range in the 22 switch. The layer-3 switch serves as the default gateway for each private network and ensures a packet can be routed from one host to any other host in the same Zone.

VLAN Allocation with Virtual Networking

With Virtual Networking, VLANs are required for the guests at the Zone level. The following is an example of a VLAN allocation scheme when Virtual Networking is used:

VLAN IDs	Use
< 500	Reserved for administrative purposes and untagged private network for each pod
500-599	Public VLANs
600-999	Zone VLANs
> 1000	Reserved for future use

VLAN Allocation with Direct Tagged Networking

With Direct Tagged networking, there is no need for Public VLANs nor Zone VLANs. There is a new need for a VLAN range for the Direct Attached guests.

VLAN IDs	Use
< 300	Reserved for administrative purposes and untagged private network for each pod
300-499	Directed Attached VLANs
> 500	Reserved for future use

VLAN Allocation with Virtual Networking and Direct Tagged Networking

CloudStack supports deployments that have both Virtual Networking and Direct Tagged guests. In this case it will be necessary to allocate VLANs for both types of guests.

VLAN IDs	Use
< 300	Reserved for administrative purposes and untagged private network for each pod
300-499	Direct Tagged VLANs
500-599	Public VLANs
600-999	Zone VLANs
> 1000	Reserved for future use

IP Address Allocation

Several types of IP addresses must be provisioned in CloudStack. The required types depend on the networking mode that is in use.

Public IP Addresses

When Advanced networking is used, CloudStack provisions one public IP address per account for use as the source NAT IP address. If a Juniper SRX firewall is used, CloudStack can instead use a single public IP address as an interface NAT IP for all accounts, reducing the number of IP addresses consumed. Users may request additional public IP addresses. The administrator must configure one or more ranges of public IP addresses for use by CloudStack. These IP addresses could be RFC1918 addresses in private clouds.

Private IP Addresses

The Hosts in a Pod are assigned private IP addresses. These are typically RFC1918 addresses. The Console Proxy and Secondary Storage system VMs are also allocated private IP addresses in the CIDR of the Pod that they are created in.

The administrator should provide private IPs for the system in each Pod and provision them in CloudStack.

- For vSphere with advanced virtual networking, we recommend provisioning enough private IPs for your total number of customers, plus enough for the required CloudStack System VMs. Typically, about 10 additional IPs are required for the System VMs. For more information about System VMs, see Working with System Virtual Machines in the Administrator's Guide.

- For KVM and XenServer, the recommended number of private IPs per Pod is one per host. If you expect a Pod to grow, add enough private IPs now to accommodate the growth.

When Advanced Virtual networking is being used, the number of private IP addresses available in each Pod varies depending on which hypervisor is running on the nodes in that Pod. Citrix XenServer and KVM use link-local addresses, which in theory provide more than 65,000 private IP addresses within the address block. As the Pod grows over time, this should be more than enough for any reasonable number of hosts as well as IP addresses for guest virtual routers. VMWare ESXi, by contrast uses any administrator-specified subnetting scheme, and the typical administrator provides only 255 IPs per Pod. Since these are shared by physical machines, the guest virtual router, and other entities, it is possible to run out of private IPs when scaling up a Pod whose nodes are running ESXi.

To ensure adequate headroom to scale private IP space in an ESXi Pod when Advanced Virtual networking is enabled, use one or more of the following techniques:

- Specify a larger CIDR block for the subnet. A subnet mask with a /20 suffix will provide more than 4,000 IP addresses.
- Create multiple pods, each with its own subnet. For example, if you create 10 Pods and each pod has 255 IPs, this will provide 2,550 IP addresses.

Direct IP Addresses

In Basic Mode, CloudStack will assign IP addresses in the CIDR of the Pod to the guests in that Pod. The administrator must add a Direct IP range on the Pod for this purpose. These IPs are in the same untagged VLAN as the Hosts.

In Advanced Mode, the administrator can create additional networks for use by the guests. These networks can be Zone wide, in which case they are available to all guests, or account-specific, in which case only the named account may create guests that attach to these networks. The networks are defined by a VLAN ID, IP range, and gateway. The administrator may provision thousands of these networks if desired.

Guest IP Addresses - Virtual Networking

With virtual networking, CloudStack manages the IP address assignment for the guests in an account. The administrator should set a global configuration parameter to name the CIDR, but there is no need to manage the CIDR on a per-account basis. All virtual networks in the Cloud will use the same CIDR for DHCP and IP address allocation.

Layer-3 Switch

The layer-3 switch is the core switching layer at the Availability Zone level. The layer-3 switch should be programmed as follows:

- If direct tagged or virtual networking is in use, the layer-3 switch trunks public VLANs, Zone VLANs, and Direct Attached VLANs into each Pod.
- The layer-3 switch functions as the gateway for the untagged private network. A separate VLAN is created in the layer-3 switch for each private IP address range. The layer-3 switch should allow packets to flow between private IP ranges.

The "Virtual Network and Direct Tagged" VLAN allocation in this section is used in the configurations described for layer 2 and layer 3 switches. You can adjust VLAN allocation according to your specific needs.

Example Configurations

This section contains an example configuration of specific switch models for Zone-level layer-3 switching. It assumes VLAN management protocols, such as VTP or GVRP, have been disabled. The example scripts must be changed appropriately if you choose to use VTP or GVRP.

Dell 62xx

The following steps show how a Dell 62xx is configured for Zone-level layer-3 switching. These steps assume VLAN 201 is used to route untagged private IPs for Pod 1, and Pod 1's layer-2 switch is connected to Ethernet port 1/g1.

The Dell 62xx Series switch supports up to 1024 VLANs.

1. Configure all the VLANs in the database.

```
vlan database
vlan 200-999
exit
```

2. Configure Ethernet port 1/g1.

```
interface ethernet 1/g1
switchport mode general
switchport general pvid 201
switchport general allowed vlan add 201 untagged
switchport general allowed vlan add 300-999 tagged
exit
```

The statements configure Ethernet port 1/g1 as follows:

- VLAN 201 is the native untagged VLAN for port 1/g1.
- Public VLANs (500-599) and Zone VLANs (600-999) and Direct Attached VLANs (300-499) are passed to all the Pod-level layer-2 switches.

Cisco 3750

The following steps show how a Cisco 3750 is configured for Zone-level layer-3 switching. These steps assume VLAN 201 is used to route untagged private IPs for Pod 1, and Pod 1's layer-2 switch is connected to GigabitEthernet1/0/1.

1. Setting VTP mode to transparent allows us to utilize VLAN IDs above 1000. Since we only use VLANs up to 999, vtp transparent mode is not strictly required.

```
vtp mode transparent
vlan 200-999
exit
```

2. Configure GigabitEthernet1/0/1.

```
interface GigabitEthernet1/0/1
switchport trunk encapsulation dot1q
switchport mode trunk
switchport trunk native vlan 201
exit
```

The statements configure GigabitEthernet1/0/1 as follows:

- VLAN 201 is the native untagged VLAN for port GigabitEthernet1/0/1.
- Cisco passes all VLANs by default. As a result Public VLANs (500-599), Zone VLANs (600-999), and Direct Attached VLANs (300-499) are passed to all the Pod-level layer-2 switches.

Layer-2 Switch

The layer-2 switch is the access switching layer inside the Pod.

- It should trunk Public VLANs, Zone VLANs, and Direct Attached VLANs into every computing host.
- It should switch untagged traffic for the private network containing computing and storage hosts. The layer-3 switch will serve as the gateway for the private network.

Example Configurations

This section contains example configurations for specific switch models for Pod-level layer-2 switching. It assumes VLAN management protocols such as VTP or GVRP have been disabled. The scripts must be changed appropriately if you choose to use VTP or GVRP.

Dell 62xx

The following steps show how a Dell 62xx is configured for Pod-level layer-2 switching.

1. Configure all the VLANs in the database.

```
vlan database
vlan 300-999
exit
```

2. VLAN 201 is used to route untagged private IP addresses for Pod 1, and Pod 1 is connected to this layer-2 switch.

```
interface range ethernet all
switchport mode general
switchport general allowed vlan add 300-999 tagged
exit
```

The statements configure all Ethernet ports to function as follows:

- All ports are configured the same way.
- Public VLANs (500-599) , Zone VLANs (600-999) and Direct Attached VLANs (300-499) are passed through all the ports of the layer-2 switch.

Cisco 3750

The following steps show how a Cisco 3750 is configured for Pod-level layer-2 switching.

1. Setting VTP mode to transparent allows us to utilize VLAN IDs above 1000. Since we only use VLANs up to 999, vtp transparent mode is not strictly required.

```
vtp mode transparent
```

```
vlan 300-999
exit
```

2. Configure all ports to dot1q and set 201 as the native VLAN.

```
interface range GigabitEthernet 1/0/1-24
switchport trunk encapsulation dot1q
switchport mode trunk
switchport trunk native vlan 201
exit
```

By default, Cisco passes all VLANs. Cisco switches complain of the native VLAN IDs are different when 2 ports are connected together. That's why we specify VLAN 201 as the native VLAN on the layer-2 switch.

Hardware Firewall

All deployments should have a firewall protecting the management server; see Generic Firewall Provisions. Optionally, some deployments may also have a Juniper firewall that will be the default gateway for the guest virtual networks; see External Guest Firewall Integration for Juniper (optional).

Generic Firewall Provisions

The hardware firewall is required to serve two purposes:

- Protect the Management Server. NAT and port forwarding should be configured to direct traffic from the public Internet to the Management Server.
- Route private network traffic between multiple Availability Zones. Site-to-site VPN should be configured between multiple Availability Zones.

To achieve the above purposes you must set up fixed configurations for the firewall. Firewall rules and policies need not change as users are provisioned into the cloud. Any brand of hardware firewall that supports NAT and site-to-site VPN can be used.

External Guest Firewall Integration for Juniper (optional)

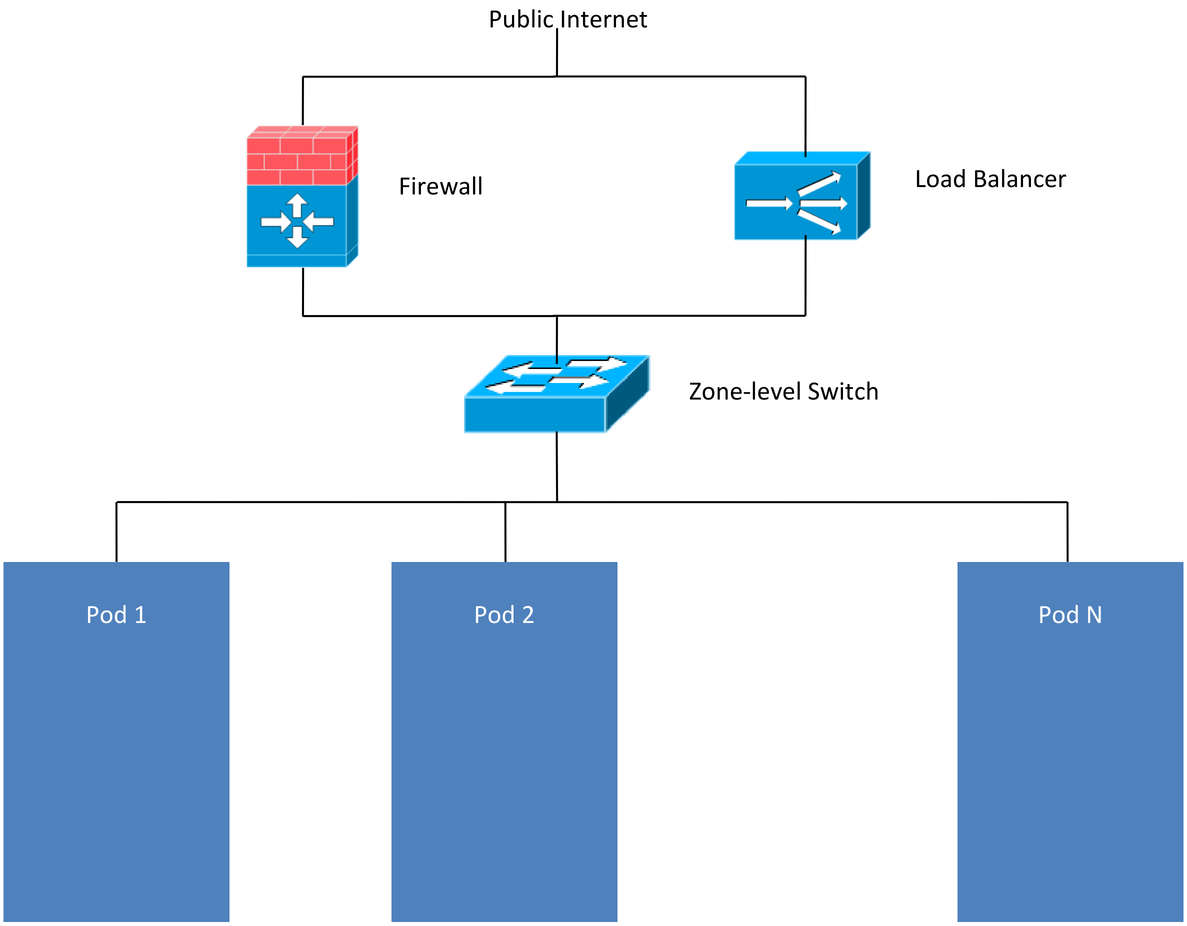
Available only for guests using virtual networking (in Advanced Mode).

CloudStack provides for direct management of the Juniper SRX series of firewalls. This enables CloudStack to establish static NAT mappings from public IPs to guest VMs, and to use the Juniper device in place of the virtual router for firewall services. You can have one Juniper SRX per zone. This feature is optional. If Juniper integration is not provisioned, CloudStack will use the virtual router for these services.

> All accounts with guests in a given zone will use the external firewall if it is configured. It is not currently possible to have some guests or some accounts in a zone use the external firewall and others use the virtual router.

The Juniper SRX can optionally be used in conjunction with an external load balancer.

External Network elements can be deployed in a side-by-side or inline configuration.



The SRX software must be version 10.3 or higher.

CloudStack requires a Juniper configuration as follows.

1. Install your SRX appliance according to the vendor's instructions.
2. Connect one interface to the private network and one interface to the public network. Alternatively, you can connect the same interface to both networks and a use a VLAN for the public network.
3. Make sure "vlan-tagging" is enabled on the private interface.
4. Record the public and private interface names. If you used a VLAN for the public interface, add a ".[VLAN TAG]" after the interface name. For example, if you are using ge-0/0/3 for your public interface and VLAN tag 301, your public interface name would be "ge-0/0/3.301". Your private interface name should always be untagged because the CloudStack software automatically creates tagged logical interfaces.
5. Create a public security zone and a private security zone. By default, these will already exist and will be called "untrust" and "trust". Add the public interface to the public zone and the private interface to the private zone. Note down the security zone names.
6. Make sure there is a security policy from the private zone to the public zone that allows all traffic.

7. Note the username and password of the account you want the CloudStack software to log in to when it is programming rules.

8. Make sure the "ssh" and "xnm-clear-text" system services are enabled.

9. If traffic metering is desired, create an incoming firewall filter and an outgoing firewall filter. These filters should be the same names as your public security zone name and private security zone name respectively. The filters should be set to be "interface-specific". For example, here is the configuration where the public zone is "untrust" and the private zone is "trust":

```
root@cloud-srx# show firewall
filter trust {
    interface-specific;
}

filter untrust {
    interface-specific;
}
```

10. If traffic metering is desired, add the firewall filters to your public interface. For example, a sample configuration output (for public interface ge-0/0/3.0, public security zone untrust, and private security zone trust) is:

```
ge-0/0/3 {
    unit 0 {
        family inet {
            filter {
                input untrust;
                output trust;
            }
            address 172.25.0.252/16;
        }
    }
}
```

11. Make sure all Zone VLANs are brought to the private interface of the SRX.

12. CloudStack will be configured with the Juniper information after the Management Server is installed.

Management Server Load Balancing

CloudStack can use a load balancer to provide a virtual IP for multiple Management Servers. The administrator is responsible for creating the load balancer rules for the Management Servers. The application requires persistence or stickiness across multiple sessions. The following chart lists the ports that should be load balanced and whether or not persistence is required.

Even if persistence is not required, enabling it is permitted.

Source Port	Destination Port	Protocol	Persistence Required?
80 or 443	8080 (or 20400 with AJP)	HTTP (or AJP)	Yes
8250	8250	TCP	Yes
8096	8096	HTTP	No

External Guest Load Balancer Integration for F5 (optional)

Only for guests using virtual networking (Advanced Mode).

CloudStack can optionally use an F5 load balancer to provide load balancing services to guests. If this is not enabled, CloudStack will use the software load balancer in the virtual router.

If an external load balancer is used, an external firewall must also be used.

To install and enable your F5 for CloudStack management:

1. Set up your F5 BigIp appliance according to the vendor's directions.
2. Connect it to the public network and the private network.
3. Record the IP address, username, password, public interface name, and private interface name. The interface names will be something like "1.1" or "1.2".
4. Make sure that the Zone VLANs are trunked to the private network interface.
5. CloudStack will be configured with the F5 information after the Management Server is installed.

Direct Network Usage Integration for Traffic Sentinel

To collect usage data for a direct network, CloudStack needs to pull the data from an external network statistics collector installed on the network. Metering statistics for Direct Networking are available through CloudStack’s integration with inMon Traffic Sentinel™.

Traffic Sentinel is a network traffic usage data collection package. CloudStack can feed statistics from Traffic Sentinel into its own usage records, providing a basis for billing users of Direct Networked cloud infrastructure. Traffic Sentinel uses the traffic monitoring protocol sFlow®. Routers and switches generate sFlow records and provide them for collection by Traffic Sentinel, then CloudStack queries the Traffic Sentinel database to obtain this information.

To construct the query, CloudStack determines what guest IPs were in use during the current query interval. This includes both newly assigned IPs and IPs that were assigned in a previous time period and continued to be in use. CloudStack queries Traffic Sentinel for network statistics that apply to these IPs during the time period they remained allocated in CloudStack. The returned data is correlated with the customer account that owned each IP and the timestamps when IPs were assigned and released in order to create billable metering records in CloudStack. When the Usage Server runs, it collects this data. For more information about the Usage Server, see Installing the Usage Server (Optional) on page 124.

To set up the integration between CloudStack and Traffic Sentinel:

1. On your network infrastructure, install Traffic Sentinel and configure it to gather traffic data. For installation and configuration steps, see inMon documentation at http://inmon.com.
2. In the Traffic Sentinel UI, configure Traffic Sentinel to accept script querying from guest users. CloudStack will be the guest user performing the remote queries to gather network usage for one or more IP addresses.
   a. Click File – Users – Access Control – Reports Query, then select Guest from the dropdown list.
   b. Click File – Users – Access Control – Reports Script, then select Guest from the dropdown list.
3. On CloudStack, add the Traffic Sentinel host by calling the CloudStack API command addTrafficMonitor. Pass in the URL of the Traffic Sentinel as protocol + host + port (optional); for example, http://10.147.28.100:8080. For the command syntax, see the API Reference at http://download.cloud.com/releases/2.2.0/api_2.2.8/global_admin/addTrafficMonitor.html. For information about how to call the CloudStack API, see the Developer's Guide at http://docs.cloud.com/CloudStack_Documentation/Developer's_Guide%3A_CloudStack.
4. In the CloudStack admin UI, click Configuration – Global Settings. Set direct.network.stats.interval to the length of time you would like to pass between queries to Traffic Sentinel.

Topology Requirements

Security Requirements

- The public Internet must not be able to access port 8096 or port 8250 on the Management Server.

Runtime Internal Communications Requirements

- The Management Servers communicate with each other to coordinate tasks. This communication uses TCP on ports 8250 and 9090.
- The console proxy VMs connect to all hosts in the zone over the management traffic network. Therefore the management traffic network of any given pod in the zone must have connectivity to the management traffic network of all other pods in the zone.
- The secondary storage VMs and console proxy VMs connect to the Management Server on port 8250. If you are using multiple Management Servers, the load balanced IP address of the Management Servers on port 8250 must be reachable.

Storage Network Topology Requirements

- The secondary storage NFS export is mounted by the secondary storage VM. Secondary storage traffic goes over the management traffic network, even if there is a separate storage network. Primary storage traffic goes over the storage network, if available. If you choose to place secondary storage NFS servers on the storage network, you must make sure there is a route from the management traffic network to the storage network.

External Firewall Topology Requirements

- When external firewall integration is in place, the public IP VLAN must still be trunked to the Hosts. This is required to support the Secondary Storage VM and Console Proxy VM.

Advanced Zone Topology Requirements

- With Advanced Networking, separate subnets must be used for private and public networks.

XenServer Topology Requirements

- The Management Servers communicate with XenServer hosts on ports 22 (ssh), 80 (HTTP), and 443 (HTTPs).

VMware Topology Requirements

- The Management Server and secondary storage VMs must be able to access vCenter and all ESXi hosts in the zone. To allow the necessary access through the firewall, keep port 443 open.
- The Management Servers communicate with VMware vCenter servers on port 443 (HTTPs).
- The Management Servers communicate with the System VMs on port 3922 (ssh) on the management traffic network.

KVM Topology Requirements

- The Management Servers communicate with KVM hosts on port 22 (ssh).

Storage Setup

CloudStack is designed to work with a wide variety of commodity and enterprise-grade storage. Local disk may be used as well, if supported by the selected hypervisor. Storage type support for guest virtual disks differs based on hypervisor selection.

	XenServer	vSphere	OVM	KVM
NFS	Supported	Supported	Supported	Supported
iSCSI	Supported	Supported via VMFS	Supported via OCFS2	Supported via Clustered Filesystems
Fiber Channel	Supported via Pre-existing SR	Supported	Not Supported	Supported via Clustered Filesystems
Local Disk	Supported	Supported	Not Supported	Not Supported

Small-Scale Setup

In a small-scale setup, a single NFS server can function as both primary and secondary storage. The NFS server just needs to export two separate shares, one for primary storage and the other for secondary storage.

Secondary Storage

CloudStack is designed to work with any scalable secondary storage system. The only requirement is the secondary storage system supports the NFS protocol.

Example Configurations

In this section we go through a few examples of how to set up storage to work properly with CloudStack on a few types of NFS and iSCSI storage systems.

Linux NFS on Local Disks and DAS

This section describes how to configure an NFS export on a standard Linux installation. Instructions in this section specifically apply to RHEL/CentOS 5. Steps to setup other distributions may vary.

> The storage server should be a machine with a large number of disks. The disks should ideally be managed by a hardware RAID controller. Modern hardware RAID controllers support hot plug functionality independent of the operating system so you can replace faulty disks without impacting the running operating system.

1. Install the RHEL/CentOS distribution on the storage server.
2. If the root volume is more than 2 TB in size, create a smaller boot volume to install RHEL/CentOS. A root volume of 20 GB should be sufficient.
3. After the system is installed, create a directory called /export. This can each be a directory in the root partition itself or a mount point for a large disk volume.
4. If you have more than 16TB of storage on one host, create multiple EXT3 file systems and multiple NFS exports. Individual EXT3 file systems cannot exceed 16TB.
5. After /export directory is created, run the following command to configure it as an NFS export.

   ```
   # echo "/export <CIDR>(rw,async,no_root_squash)" >> /etc/exports
   ```

   Adjust the above command to suit your deployment needs.

   - **Limiting NFS export**. It is highly recommended that you limit the NFS export to a particular subnet by specifying a subnet mask (e.g.,”192.168.1.0/24”). By allowing access from only within the expected cluster, you avoid having non-pool member mount the storage. **The limit you place must include the private network(s) and the storage network(s).** If the two are the same network then one CIDR is sufficient. If you have a separate storage network you must provide separate CIDR’s for both or one CIDR that is broad enough to span both.

   The following is an example with separate CIDRs:

   ```
   /export 192.168.1.0/24(rw,async,no_root_squash) 10.50.1.0/24(rw,async,no_root_squash)
   ```

   - **Removing the async flag**. The async flag improves performance by allowing the NFS server to respond before writes are committed to the disk. Remove the async flag in your mission critical production deployment.
6. Run the following command to enable NFS service.

   ```
   # chkconfig nfs on
   ```

7. Edit the /etc/sysconfig/nfs file and uncomment the following lines.

   ```
   LOCKD_TCPPORT=32803
   LOCKD_UDPPORT=32769
   MOUNTD_PORT=892
   RQUOTAD_PORT=875
   STATD_PORT=662
   STATD_OUTGOING_PORT=2020
   ```

8. Edit the /etc/sysconfig/iptables file and add the following lines at the beginning of the INPUT chain.

   ```
   -A INPUT -m state --state NEW -p udp --dport 111 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 111 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 2049 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 32803 -j ACCEPT
   -A INPUT -m state --state NEW -p udp --dport 32769 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 892 -j ACCEPT
   -A INPUT -m state --state NEW -p udp --dport 892 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 875 -j ACCEPT
   -A INPUT -m state --state NEW -p udp --dport 875 -j ACCEPT
   -A INPUT -m state --state NEW -p tcp --dport 662 -j ACCEPT
   -A INPUT -m state --state NEW -p udp --dport 662 -j ACCEPT
   ```

9. Reboot the server.

An NFS share called /export is now set up.

Linux NFS on iSCSI

Use the following steps to set up a Linux NFS server export on an iSCSI volume. These steps apply to RHEL/CentOS 5 distributions.

1. Install iscsiadm.

   ```
   # yum install iscsi-initiator-utils
   # service iscsi start
   # chkconfig --add iscsi
   # chkconfig iscsi on
   ```

2. Discover the iSCSI target.

   ```
   # iscsiadm -m discovery -t st -p <iSCSI Server IP address>:3260
   ```

   For example:

   ```
   # iscsiadm -m discovery -t st -p 172.23.10.240:3260
   172.23.10.240:3260,1 iqn.2001-05.com.equallogic:0-8a0906-83bcb3401-16e0002fd0a46f3d-
   rhel5-test
   ```

3. Log in.

   ```
   # iscsiadm -m node -T <Complete Target Name> -l -p <Group IP>:3260
   ```

For example:

```
# iscsiadm -m node -l -T iqn.2001-05.com.equallogic:83bcb3401-16e0002fd0a46f3d-rhel5-
test -p 172.23.10.240:3260
```

4. Discover the SCSI disk. For example:

   ```
   # iscsiadm -m session -P3 | grep Attached
   Attached scsi disk sdb State: running
   ```

5. Format the disk as ext3 and mount the volume.

   ```
   # mkfs.ext3 /dev/sdb
   # mkdir -p /export
   # mount /dev/sdb /export
   ```

6. Add the disk to /etc/fstab to make sure it gets mounted on boot.

   ```
   /dev/sdb /export ext3 _netdev 0 0
   ```

Now you can set up /export as an NFS share.

- **Limiting NFS export**. In order to avoid data loss, it is highly recommended that you limit the NFS export to a particular subnet by specifying a subnet mask (e.g.,”192.168.1.0/24”). By allowing access from only within the expected cluster, you avoid having non-pool member mount the storage and inadvertently delete all its data. **The limit you place must include the private network(s) and the storage network(s).** If the two are the same network then one CIDR is sufficient. If you have a separate storage network you must provide separate CIDRs for both or one CIDR that is broad enough to span both.

  The following is an example with separate CIDRs:

  ```
  /export 192.168.1.0/24(rw,async,no_root_squash) 10.50.1.0/24(rw,async,no_root_squash)
  ```

- **Removing the async flag**. The async flag improves performance by allowing the NFS server to respond before writes are committed to the disk. Remove the async flag in your mission critical production deployment.

Additional Installation Options

The next few sections describe CloudStack features above and beyond the basic deployment options.

Edit the Global Configuration Settings (Optional)

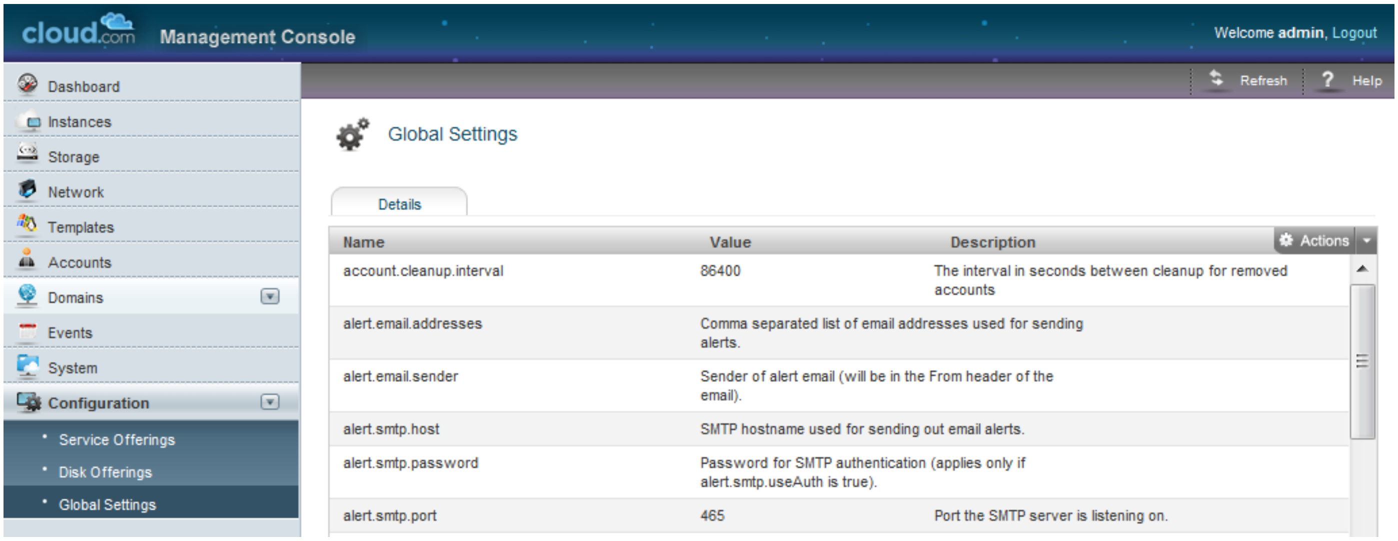
Once your Management Server is running, you might need to set some global configuration parameters.

1. Log in to the administrator web UI.

   ```
   http://management-server-ip-address:8080/client
   ```

   The default credentials are “admin” for user and “password” for password. The domain field should be left blank. A blank domain field is defaulted to the ROOT domain.

2. Using the left navigation tree, click on Configuration then Global Settings.



Name	Value	Description
account.cleanup.interval	86400	The interval in seconds between cleanup for removed accounts
alert.email.addresses		Comma separated list of email addresses used for sending alerts.
alert.email.sender		Sender of alert email (will be in the From header of the email).
alert.smtp.host		SMTP hostname used for sending out email alerts.
alert.smtp.password		Password for SMTP authentication (applies only if alert.smtp.useAuth is true).
alert.smtp.port	465	Port the SMTP server is listening on.

You **might** need to edit the following fields.

Field	Value
management.network.cidr	A CIDR that describes the network that the management CIDRs reside on. **This variable must be set for deployments that use vSphere.** It is recommended to be set for other deployments as well. Example: 192.168.3.0/24.

Field	Value
xen.public.network.device	For XenServer nodes, this is the device name with the name-label that was used for the public network. For example, “cloud-public”. **This is applicable only if you have a second NIC for the public network.** In this case, set the value to the name-label used for the separate public network created in Configuring Public Network with a Dedicated NIC for XenServer (optional) on page 58.
xen.private.network.device	For XenServer nodes, this is the device name with the name-label that was used for the private network. For example, “cloud-private”.
kvm.public.network.device	For KVM nodes, this is the device that was used for the public network. For example, “cloudbr0”.
kvm.private.network.device	For KVM nodes, this is the device that was used for the private network. For example, “cloudbr1”.
xen.setup.multipath	For XenServer nodes, this is a true/false variable that instructs CloudStack to enable iSCSI multipath on the XenServer Hosts when they are added. This defaults to false. Set it to true if you would like CloudStack to enable multipath. If this is true for a NFS-based deployment multipath will still be enabled on the XenServer host. However, this does not impact NFS operation and is harmless.
secstorage.allowed.internal.sites	This is used to protect your internal network from rogue attempts to download arbitrary files using the template download feature. This is a comma-separated list of CIDRs. If a requested URL matches any of these CIDRs the Secondary Storage VM will use the private network interface to fetch the URL. Other URLs will go through the public interface. We suggest you set this to 1 or 2 hardened internal machines where you keep your templates. For example, set it to 192.168.1.66/32.
use.local.storage	Determines whether CloudStack will use storage that is local to the Host for VHDs. By default CloudStack will not use this storage. You should change this to true if you want to use local storage and you understand the reliability and feature drawbacks to choosing local storage.
Host	This is the IP address of the Management Server. If you are using multiple Management Servers you should enter a load balanced IP address that is reachable via the private network.

Field	Value
default.page.size	Maximum number of items per page that can be returned by a CloudStack API command. The limit applies at the cloud level and can vary from cloud to cloud. You can override this with a lower value on a particular API call by using the page and pagesize API command parameters. For more information, see the Developer's Guide. Default: 500.
vmware.guest.vswitch	The name of the vSwitch in vCenter that you want to carry the guest-guest traffic. Default is vSwitch0.
vmware.private.vswitch	The name of the vSwitch in vCenter that you want to carry the management and storage traffic. Default is vSwitch0.
vmware.public.vswitch	The name of the vSwitch in vCenter that you want to carry the traffic to/from the public internet. Default is vSwitch0.

There are additional configuration parameters that you may want to set. These are discussed in the Administration Guide.

3. If you changed any of these values, restart the Management Server.

```
# service cloud-management restart
```

Edit Service Offerings (Optional)

The service offering defines CPU and RAM for the guests. CloudStack ships with several default service offerings. You can optionally use the following steps to edit these.

1. Go to the Configuration tab, and select the Service Offerings section.
2. Add or edit service offerings as needed. Select "Add Service Offering" to add one.
3. Provide the following information to define this service offering.
   - **Name**. Any desired name for the service offering.
   - **Display text**. A short description of the offering.
   - **Storage type**. The type of disk that should be allocated to the guest. Local allocates from storage attached to XenServer directly. Shared allocates from storage accessible via NFS.
   - **# of CPU cores**. The number of cores which should be allocated to an instance with this offering.
   - **CPU (in MHz)**. The CPU speed of the cores that the instance is allocated. For example, “2000” would provide for a 2 GHz clock.
   - **Memory (in MB)**. The amount of memory in megabytes that the instance should be allocated. For example, “2048” would provide for a 2 GB RAM allocation.
   - **Network Rate**. Allowed data transfer rate in MB per second.

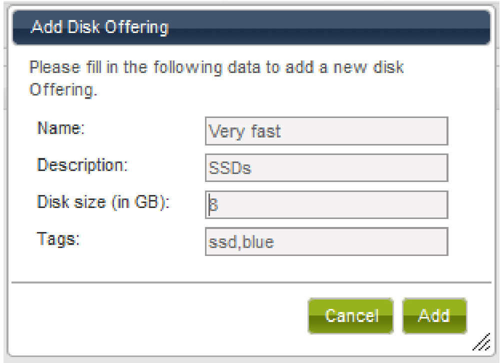
- **Offer HA**. If yes, the user will be able to choose a VM to be monitored and as highly available as possible.
- **Storage Tags (optional)**. The tags that should be associated with the primary storage for this disk.
- **Host Tags (optional)**. Any tags that you use to organize your hosts.
- **CPU cap**. Whether to cap users at their purchased level of CPU usage even if spare capacity is available.
- **Public**. Should the service offering be available all domains or only some domains. Choose Yes to make it available to all domains. Choose No to limit the scope to a subdomain; CloudStack will then prompt for the subdomain's name.

4. Click Add.

Edit Disk Offerings (Optional)

The disk offering defines the size and characteristics of data disks attached to the guests. CloudStack ships with several default disk offerings.

1. Go to the Configuration tab, and select the Disk Offerings section. This is located next to the Zones section.
2. Add or edit disk offerings as needed. Select "Add Disk Offering" to add one.



Add Disk Offering

Please fill in the following data to add a new disk Offering.

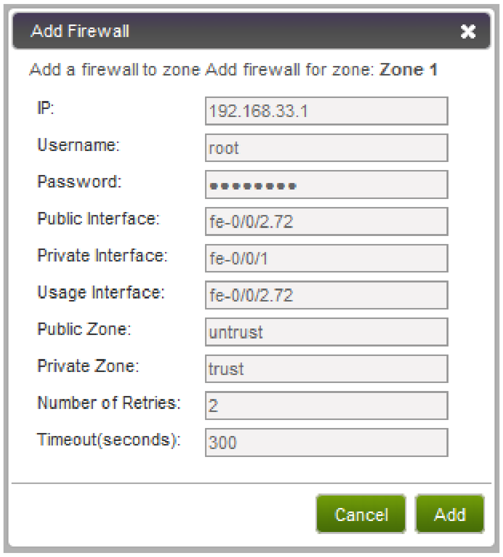
Name:	Very fast
Description:	SSDs
Disk size (in GB):	8
Tags:	ssd,blue

3. Provide the following information to define this disk offering.
   - **Name**. Name of the disk offering (E.g. extra large)
   - **Description**. A short description of the disk offering
   - **Disk size (in GB)**. The size of the disk offering in GB (E.g. 10 is a 10 GB offering)
   - **Tags**. Tags are a comma separated list of attributes of the storage. For example "ssd,blue". Tags are optional. They are also added on Primary Storage. CloudStack matches tags on a disk offering to tags on the storage. If a tag is present on a disk offering that tag (or tags) must also be present on Primary Storage for the volume to be provisioned. If no such primary storage exists allocation from the disk offering will fail.

Advanced Networking: Adding an External Firewall (Optional)

This step is required if you would like to add an external firewall device to CloudStack. The Juniper SRX series is the only supported such device. In this procedure it is assumed you have already installed and configured the firewall. Before you can use it, you must add it to the CloudStack deployment.

1. Log in to the Management Server UI.
2. In the left navigation tree, choose System - Physical Resources, choose your Zone, then choose Network.
3. Click Add Firewall.



Add a firewall to zone Add firewall for zone: **Zone 1**

IP:	192.168.33.1
Username:	root
Password:	••••••••
Public Interface:	fe-0/0/2.72
Private Interface:	fe-0/0/1
Usage Interface:	fe-0/0/2.72
Public Zone:	untrust
Private Zone:	trust
Number of Retries:	2
Timeout(seconds):	300

4. Input the following:
   - **IP.** The IP address of the SRX.
   - **Username.** The user name of the account on the SRX that CloudStack should use.
   - **Password.** The password of the account.
   - **Public Interface.** The name of the public interface on the SRX. For example, ge-0/0/2. A ".x" at the end of the interface indicates the VLAN that is in use.
   - **Private Interface.** The name of the private interface on the SRX. For example, ge-0/0/1.
   - **Usage Interface.** (Optional) Typically, the public interface is used to meter traffic. If you want to use a different interface, specify its name here.
   - **Public Zone.** The name of the public zone on the SRX. For example, trust.
   - **Private Zone.** The name of the private zone on the SRX. For example, untrust.
   - **Number of Retries.** The number of times to attempt a command on the SRX before failing. The default value is 2.
   - **Timeout (seconds).** The time to wait for a command on the SRX before considering it failed. Default is 300 seconds.

5. Click Add.

In the external networking case, every VM in a zone must have a unique guest IP address. There are two variables that you need to consider in determining how to configure CloudStack to support this: how many Zone VLANs do you expect to have and how many VMs do you expect to have running in the Zone at any one time.

Use the following table to determine how to configure CloudStack for your deployment.

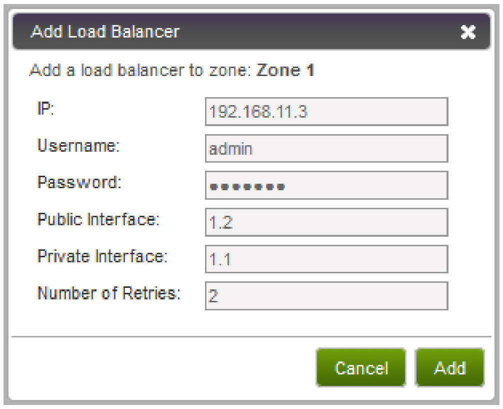
guest.vlan.bits	Maximum Running VMs per Zone	Maximum Zone VLANs
12	4096	4094
11	8192	2048
10	16384	1024
9	32768	512

Based on your deployment's needs, choose the appropriate value of guest.vlan.bits. Set that in the global configuration table as discussed in the first part of this section and restart the Management Server.

Advanced Networking: Adding an External Load Balancer (Optional)

This step is required if you would like to add an external load balancer to CloudStack. The BigIP F5 is the only supported external load balancing device. Add the BigIP via the UI.

1. Go to Physical Resources -> (Select Zone) -> Network -> Add Load Balancer button.



Add a load balancer to zone: **Zone 1**

IP:	192.168.11.3
Username:	admin
Password:	●●●●●●●
Public Interface:	1.2
Private Interface:	1.1
Number of Retries:	2

Parameters for this dialog include:

- **IP.** Enter the IP address of the F5.
- **User name**. Enter the user name of the account on the F5 that CloudStack should use.
- **Password**. Enter the password of the user name above.
- **Public Interface.** Enter the name of the public interface on the F5.
- **Private Interface.** Enter the name of the private interface on the F5.
- **Number of Retries.** Number of times to attempt a command on the Load Balancer before considering the operation failed. Default is 2.

If the F5 is in side-by-side mode, you are finished. You can proceed to add VMs and NAT/load balancing rules.

2. To add an F5 in inline mode, perform the following additional steps before the first user VM is started:
   a. Stop all management servers.
   b. Log into the CloudStack database.
   c. Run the following command:

```
update host_details set value=’true’ where name=’inline’
```

   d. Start all management servers.

Add VMs and NAT/LB rules just as you would in side-by-side mode. CloudStack will handle them as inline mode.

Installing the Usage Server (Optional)

You can optionally install the Usage Server once the Management Server is configured properly. The Usage Server takes data from the events in the system and enables usage-based billing for accounts.

When multiple Management Servers are present, the Usage Server may be installed on any number of them. The Usage Servers will coordinate usage processing. A site that is concerned about availability should install Usage Servers on at least two Management Servers.

Requirements for Installing the Usage Server

- The Management Server must be running when the Usage Server is installed.
- The Usage Server must be installed on the same server as a Management Server.

Steps to Install the Usage Server

1. Run ./install.sh.

```
# ./install.sh
```

2. Choose “S” to install the Usage Server.

```
    > S
```

3. Once installed, start the Usage Server with the following command.

```
# service cloud-usage start
```

The Administration Guide discusses further configuration of the Usage Server.

SSL (Optional)

CloudStack provides HTTP access in its default installation. There are a number of technologies and sites which choose to implement SSL. As a result, we have left CloudStack to expose HTTP under the assumption that a site will implement its typical practice.

CloudStack uses Tomcat as its servlet container. For sites that would like CloudStack to terminate the SSL session, Tomcat's SSL access may be enabled. Tomcat SSL configuration is described at http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html.

Database Replication (Optional)

CloudStack supports database replication from one MySQL node to another. This is achieved using standard MySQL replication. You may want to do this as insurance against MySQL server or storage loss. MySQL replication is implemented using a master/slave model. The master is the node that the Management Servers are configured to use. The slave is a standby node that receives all write operations from the master and applies them to a local, redundant copy of the database. The following steps are a guide to implementing MySQL replication.

> Creating a replica is not a backup solution. You should develop a backup procedure for the MySQL data that is distinct from replication.

1. Ensure that this is a fresh install with no data in the master.
2. Edit my.cnf on the master and add the following in the [mysqld] section below datadir.

```
log_bin=mysql-bin
server_id=1
```

   The server_id must be unique with respect to other servers. The recommended way to achieve this is to give the master an ID of 1 and each slave a sequential number greater than 1, so that the servers are numbered 1, 2, 3, etc. Restart the MySQL service:

```
# service mysqld restart
```

3. Create a replication account on the master and give it privileges. We will use the "cloud-repl" user with the password "password". This assumes that master and slave run on the 172.16.1.0/24 network.

```
# mysql -u root
mysql> create user 'cloud-repl'@'172.16.1.%' identified by 'password';
mysql> grant replication slave on *.* TO 'cloud-repl'@'172.16.1.%';
mysql> flush privileges;
mysql> flush tables with read lock;
```

4. Leave the current MySQL session running.

5. In a new shell start a second MySQL session.

6. Retrieve the current position of the database.

```
# mysql -u root
mysql> show master status;
+------------------+----------+--------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+--------------+------------------+
| mysql-bin.000001 |      412 |              |                  |
+------------------+----------+--------------+------------------+
```

7. Note the file and the position that are returned by your instance.

8. Exit from this session.

9. Complete the master setup. Returning to your first session on the master, release the locks and exit MySQL.

```
mysql> unlock tables;
```

10. Install and configure the slave. On the slave server, run the following commands.

```
# yum install mysql-server
# chkconfig mysqld on
```

11. Edit my.cnf and add the following lines in the [mysqld] section below datadir.

```
server_id=2
innodb_rollback_on_timeout=1
innodb_lock_wait_timeout=600
```

12. Restart MySQL.

```
# service mysqld restart
```

13. Instruct the slave to connect to and replicate from the master. Replace the IP address, password, log file, and position with the values you have used in the previous steps.

```
mysql> change master to
    -> master_host='172.16.1.217',
    -> master_user='cloud-repl',
    -> master_password='password',
    -> master_log_file='mysql-bin.000001',
    -> master_log_pos=412;
```

14. Then start replication on the slave.

```
mysql> start slave;
```

15. Optionally, open port 3306 on the slave as was done on the master earlier.

This is not required for replication to work. But if you choose not to do this, you will need to do it when failover to the replica occurs.

Failover

This will provide for a replicated database that can be used to implement manual failover for the Management Servers. CloudStack failover from one MySQL instance to another is performed by the administrator. In the event of a database failure you should:

1. Stop the Management Servers (via service cloud-management stop).
2. Change the replica's configuration to be a master and restart it.
3. Ensure that the replica's port 3306 is open to the Management Servers
4. Make a change so that the Management Server uses the new database. The simplest process here is to put the IP address of the new database server into each Management Server's /etc/cloud/management/db.properties.
5. Restart the Management Servers (via service cloud-management start)

Best Practices

Deploying a cloud is challenging. There are many different technology choices to make, and CloudStack is flexible enough in its configuration that there are many possible ways to combine and configure the chosen technology. This section contains suggestions and requirements about cloud deployments.

Required Practices

- For XenServer and vSphere, do not put more than 8 hosts in a cluster. For KVM, do not put more than 16 hosts in a cluster.
- For XenServer, install PV drivers / Xen tools on each template that you create. This will enable live migration and clean guest shutdown.
- For vSphere, install VMware Tools on each template that you create. This will enable console view to work properly.

Suggested Practices

These should be treated as suggestions and not absolutes. However, we do encourage anyone planning to build a cloud outside of these guidelines to discuss their needs with us.

- Use multiple Clusters per Pod if you need to achieve a certain switch density.
- Primary storage mountpoints or LUNs should not exceed 6 TB in size. It is better to have multiple smaller primary storage elements per Cluster than one large one.
- When exporting shares on primary storage, avoid data loss by restricting the range of IP addresses that can access the storage. See "Linux NFS on Local Disks and DAS" on page 115 or "Linux NFS on iSCSI" on page 116.
- NIC bonding is straightforward to implement and provides increased reliability.
- 10G networks are generally recommended for storage access when larger servers that can support relatively more VMs are used.
- Host capacity should generally be modeled in terms of RAM for the guests. Storage and CPU may be overprovisioned. RAM may not. RAM is usually the limiting factor in capacity designs.
- A staging system that models the production environment is strongly advised. It is critical if customizations have been applied to CloudStack.
- Monitor host disk space. Many host failures occur because the host's root disk fills up from logs that were not rotated adequately.
- Allow adequate time for installation, a beta, and learning the system. Installs with Basic Networking can be done in a day or two. Installs with Advanced Networking usually take several days for the first attempt, with complicated installations taking longer. Allow at least 4-8 weeks for a beta to work through all of the integration issues. It takes months to gain confidence with CloudStack and related technologies. You may want to contact our sales team about training sessions to help accelerate this.

Troubleshooting

Checking the Management Server Log

A quick step to look for errors in the management server log is this:

```
# grep -i -E 'exception|unable|fail|invalid|leak|invalid|warn'
/var/log/cloud/management/management-server.log
```

Troubleshooting the Secondary Storage VM

Many install problems relate to the secondary storage VM. Sample common problems:

- SSVM cannot reach the DNS server
- SSVM cannot reach the Management Server
- SSVM cannot reach the outside world to download templates. It contacts download.cloud.com via HTTP.
- The configured DNS server cannot resolve your internal hostnames. E.g., you entered private-nfs.lab.example.org for secondary storage NFS, but gave a DNS server that your customers use, and that server cannot resolve private-nfs.lab.example.org.

> To recover a failed SSVM after making changes that fix the root cause of the failure, you must stop the VM first and then start it. A restart merely reboots the VM without resending the configuration, which may have changed.

You can troubleshoot the secondary storage VM either by running a diagnostic script or by checking the log file. The following sections detail each of these methods.

If you have corrected the problem but the template hasn't started to download, restart the cloud service with "service cloud restart". This will restart the default CentOS template download.

Running a Diagnostic Script

You can log into the SSVM. To do this you have to find the Host running the SSVM, ssh into it, then ssh into the SSVM's private IP from that host. Once you are logged in, use the following steps to run a diagnostic script.

1. In the admin UI, go to System -> Virtual Resources -> System VMs. Select the target VM.
2. Note the name of the Host hosting the SSVM as shown in the Host row. Also note the private IP of the SSVM as shown in the Private IP row.
3. ssh into the private IP of the SSVM with the following.

   For XenServer or KVM:

   a. ssh into the Host using your known user and password.
   b. Run the following command:

```
# ssh -i /root/.ssh/id_rsa.cloud -p 3922 root@link-local-ip
```

For VMware:

c. ssh into the CloudStack Management Server using your known user and password.

d. Run the following command:

```
# ssh -i /var/lib/cloud/management/.ssh/id_rsa -p 3922 root@private-ip
```

4. Once into the SSVM, run the following diagnostic script:

```
# /usr/local/cloud/systemvm/ssvm-check.sh
```

This script will test various aspects of the SSVM and report warnings and errors.

Checking the Log File

You can check the log files in /var/log/cloud/ for any error messages.

VLAN Issues

A common installation issue is that your VLANs are not set up correctly. VLANs must be trunked into every host in the Zone.

- VLANs must be trunked into every host in the Zone.
- In Basic Networking, the netwohrk interface in the host that is connected to the VLAN must be named cloud-guest. For example, in XenServer, the network name-label must be “cloud-guest”.

Console Proxy VM Issues

Symptom

When you launch the Console Viewer, you see this error:

```
Access is denied for console session. Please close the window
```

Cause

This most likely means that the Console Proxy VM cannot connect from its private interface to port 8250 on the Management Server (or load balanced Management server pool).

Solution

Check the following:

- Load balancer has port 8250 open
- All Management Servers have port 8250 open
- There is a network path from the CIDR in the pod hosting the Console Proxy VM to the load balancer or Management Server

- The "host" global configuration parameter is set to the load balancer if in use

Troubleshooting Bare Metal Instances

Symptom

Creating an instance from a bare metal template results in an endless loop.

Cause

If PXE is first in the boot order, the host gets stuck in an infinite PXE boot loop when creating an instance from the CloudStack UI.

Solution

Make sure hard disk is the preferred boot option. See Enable PXE on the Bare Metal Host on page 93.

Binary Logging Error when Upgrading Database

Symptom

When attempting to upgrade the database, an error like the following:

```
Unable to upgrade the db due to java.sql.SQLException: Binary logging not possible.
```

Cause

Binary logging is not enabled.

Solution

Edit the MySQL configuration (/etc/my.cnf or /etc/mysql/my.cnf, depending on your OS) and set the log-bin and binlog-format variables in the [mysqld] section. For example:

```
log-bin=mysql-bin
binlog-format= 'ROW'
```

After editing my.cnf, restart the MySQL server.

```
# service mysqld restart
```

NOTE: The binlog-format variable is supported in MySQL versions 5.1 and greater. It is not supported in MySQL 5.0. In some versions of MySQL, an underscore character is used in place of the hyphen in the variable name. For the exact syntax and spelling of each variable, consult the documentation for your version of MySQL.

Can't Add Host

A Host must have a statically allocated IP address. Host addition will error and fail if a dynamically-assigned address is present.

Contacting Support

Open-source community

A variety of channels are available for getting help with CloudStack, from forums to IRC chat and more. For details, see http://cloudstack.org/discuss/.

Commercial customers

The CloudStack support team is available to help commercial customers plan and execute their installations. To contact the support team, log in to the support portal at https://na6.salesforce.com/sserv/login.jsp?orgId=00D80000000LWom using the account credentials you received when you purchased your support contract.