
openstack-xenserver Documentation

Release latest

Sep 27, 2017

Contents

1	1. Overview	3
1.1	Changelog	5
1.2	About the Author	6
2	2. Build Controller Host	7
3	3. Install core services on controller	13
4	4. Install Identity (keystone) on controller	15
5	5. Install Images (glance) on controller	21
6	6. Install Compute (nova) on controller	25
7	7. Build XenServer Host	27
8	8. Build XenServer Compute VM	43
9	9. Install Compute (nova) on XenServer compute VM	63
10	10. Install Networking (neutron) on controller	69
11	11. Install Networking (neutron) on compute VM	73
12	12. Install Dashboard (horizon) on controller	77
13	13. Build block1 storage node OS	83
14	14. Install Block Storage (cinder) on controller	89
15	15. Install Block Storage (cinder) on storage node	91
16	16. Fix cinder quotas for the demo project	93
17	17. Launch a test Boot-From-Volume instance from Horizon	95
18	18. Build KVM Host	101
19	19. Install Compute (nova) on KVM Host	109

20	20. Install Networking (neutron) on KVM Host	113
21	21. Update images for dual-hypervisor environment	115
22	22. Create Xen CentOS 7 Image	121
23	23. Launch test Xen CentOS 7 Instance	141
24	24. Create KVM CentOS 7 Image	147
25	25. Create test KVM CentOS 7 Instance	167

Contents:

CHAPTER 1

1. Overview

The OpenStack foundation has an excellent setup guide for their October 2015 release, “Liberty”, which can be found at <http://docs.openstack.org/liberty/install-guide-rdo/>. However, this guide only deals with the use of the “KVM” hypervisor, and does not cover the use of “XenServer” hypervisor.

There are many circumstances in which it may be desirable to build an OpenStack Liberty XenServer environment. However, in my efforts to do so, I have found the available online documentation regarding using XenServer with OpenStack to be inadequate, outdated or just plain incorrect. Specifically, during this project I experienced issues with:

- XenServer networking configuration
- Nova and Neutron configurations for XenServer networking
- iSCSI authentication issues with Cinder volumes
- Cinder volume mapping errors with XenServer instances
- Cinder quota errors
- ISO image support for XenServer
- Horizon bug affecting XenServer images
- Image metadata for dual hypervisor-type environments
- Neutron requirements for dual-hypervisor-type environments
- Neutron bug affecting the use of OpenvSwitch (Required for XenServer)
- VNC console connectivity

This guide is heavily based on the OpenStack foundation’s guide. It does not go into the same level of detail, but does highlight the differences when using XenServer instead of KVM. Their guide should be considered the superior one, and the “master” guide, and I recommend reading their guide if you have no familiarity with OpenStack at all.

Some elements of this guide are also based on the following blog post: <https://www.citrix.com/blogs/2015/11/30/integrating-xenserver-rdo-and-neutron/>

On each page, I have highlighted in **bold** any steps which differ from the original guide. These are typically XenServer-specific changes.

This guide is for a simple setup with “flat” networking. There are no provisions for private “virtual” networks, or any firewall functionality. The guide also does not yet cover “swift” object storage, although this shouldn’t differ from the OpenStack foundation’s guide. A future version of the guide may add these functions.

Later pages in this guide deal with adding a KVM hypervisor to the environment. These pages include changes which I found to be necessary in order to support a dual hypervisor-type environment (i.e the use of XenServer and KVM in the same OpenStack).

Finally, there are pages regarding the creation of CentOS 7 images for both hypervisors. These pages highlight some differences in the image-creation process for both hypervisors, including the package and partitioning requirements to support automatic disk resizing and injection of SSH keys for the root user.

Two networks are required, a “public” network (which instances will be connected to for their day-to-day traffic), and a “management” network, which our OpenStack servers will use for their connectivity. Any servers with connections to both will have eth0 connected to the “public” network, and eth1 connected to the “management” network.

Any IP addresses in the guide should, of course, be replaced with your own. You will also need to pre-generate the following variables which will be referred to throughout the guide:

Variable	Meaning
MYSQL_ROOT	Root password for MySQL.
KEYSTONE_DBPASS	Password for the keystone MySQL database.
ADMIN_TOKEN	A temporary token for initial connection to keystone.
RABBIT_PASS	Password for the openstack rabbitmq user.
GLANCE_DBPASS	Password for the glance MySQL database.
GLANCE_PASS	Password for the glance identity user.
NOVA_DBPASS	Password for the nova MySQL database.
NOVA_PASS	Password for the nova identity user.
NEUTRON_DBPASS	Password for the neutron MySQL database.
NEUTRON_PASS	Password for the neutron identity user.
NEUTRON_METADATA_SECRET	Random secret string for the metadata service.
CINDER_DBPASS	Password for the cinder MySQL database.
CINDER_PASS	Password for the cinder identity user.
XENSERVER_ROOT	Root password for XenServer.
XENSERVER_IP	IP address of XenServer.
CONTROLLER_ADDRESS	A DNS address for the controller server.
ADMIN_PASS	Password for the admin identity user.
DEMO_PASS	Password for the demo identity user.
XAPI_BRIDGE	The name of the ovs bridge to be used by instances.
SERVER_IP	The IP of the server you are currently working on.
VM_IP	The IP of the “compute” VM for that hypervisor.
HOST_NAME	The hostname of the physical hypervisor (e.g. XenServer).

- The *ADMIN_TOKEN* can be created by running:

```
# openssl rand -hex 10
```

- For *XENSERVER_ROOT*, do not use a password you’re not comfortable placing in plaintext in the nova configuration.
- For *CONTROLLER_ADDRESS*, ensure that this is an address which you can reach from your workstation.
- For *XAPI_BRIDGE*, this won’t be determined until later in the build process. You should write it down for later use once it is defined.
- Any instance of “*HOST_NAME*” refers to the hostname of the **physical hypervisor host**. For example, this would be “compute1.openstack.lab.mycompany.com”, and not “compute1-vm.openstack.lab.mycompany.com”.

One final note: I do disable SELINUX in this guide, for simplicity. This is a personal choice, but I know that some people do choose to run SELINUX on their systems. The guide does include the installation of SELINUX support for openstack, so you should be able to set this back to “ENFORCING”, even after performing the installation with this set to “PERMISSIVE”. I have not tested this.

Changelog

Mar 17 2016:

- Add patch for neutron bug to the “install neutron on compute VM” page.

Mar 16 2016:

- Add nova and neutron configuration fixes for whole-host migration.
- Replace unnecessary XenServer reboot with Toolstack restart.

Mar 15 2016:

- Add cinder configuration fix to allow volume migration.
- Correct screenshot ordering on XenServer host installation page.
- Add screenshot for primary disk selection to XenServer host installation page.

Mar 9 2016:

- Add note regarding case-sensitive udev rules file.

Mar 4 2016:

- Add fix to prevent installation of kernels from Xen repository on Storage node.

Feb 19 2016:

- Add fix to Horizon config for Identity v3.
- Fix changelog order.

Feb 17 2016:

- Add steps to enable auto power-on of the “compute” VM on the XenServer host.
- Add required steps to enable migration and live migration of instances between XenServer hosts.

Feb 12 2016:

- Create changelog.
- Various clarifications.
- Extended identity’s token expiration time.
- Correct syntax for neutron ovs configuration on controller.
- Correct syntax when populating neutron database.
- Add note regarding large storage requirements for cinder image-to-volume conversion.

About the Author

My name is Alex Oughton, and I work with OpenStack clouds, as well as dedicated hosting solutions. My work doesn't involve the actual deployment of OpenStack, and so this guide was developed during a self-learning exercise. If you have any feedback regarding this guide, including any suggestions or fixes, please do contact me on Twitter: <http://twitter.com/alexoughton>.

You can also directly contribute to this guide through its github: <https://github.com/alexoughton/rtd-openstack-xenserver>.

CHAPTER 2

2. Build Controller Host

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/environment-networking-controller.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-ntp-controller.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-packages.html>

1. In this guide, I am using a Virtual Machine running on a VMWare hypervisor as my control node. If you are doing the same, you must ensure that the vSwitches on the hypervisor have “promiscuous mode” enabled.
2. Boot the control node with the CentOS 7.2.1511 DVD.
3. Set your time zone and language.
4. For “Software Selection”, set this to “Infrastructure Server”.
5. Keep automatic partitioning. Allow to install only on first disk.
6. Set the controller’s IPv4 address and hostname. Disable IPv6. Give the connection the name “eth1”.

Editing eth1

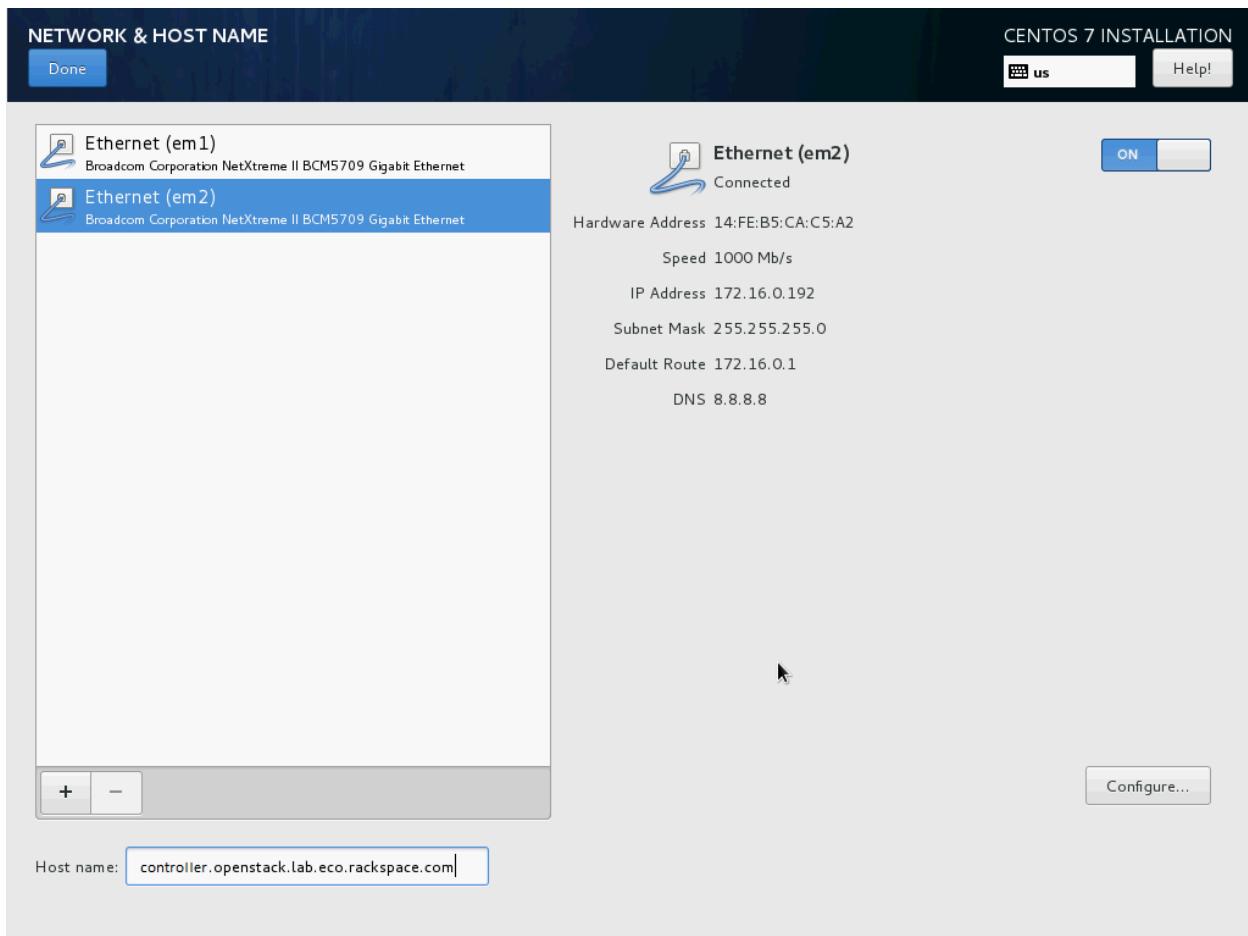
Connection name: **eth1**

General	Ethernet	802.1x Security	DCB	IPv4 Settings	IPv6 Settings					
Method: Manual										
Addresses										
<table border="1"> <thead> <tr> <th>Address</th> <th>Netmask</th> <th>Gateway</th> </tr> </thead> <tbody> <tr> <td>172.16.0.192</td> <td>255.255.255.0</td> <td>172.16.0.1</td> </tr> </tbody> </table>			Address	Netmask	Gateway	172.16.0.192	255.255.255.0	172.16.0.1	Add	Delete
Address	Netmask	Gateway								
172.16.0.192	255.255.255.0	172.16.0.1								
DNS servers: 8.8.8.8						Routes...				
Search domains:						Cancel	Save			
DHCP client ID:										
<input type="checkbox"/> Require IPv4 addressing for this connection to complete										

Editing eth1

Connection name: **eth1**

General	Ethernet	802.1x Security	DCB	IPv4 Settings	IPv6 Settings					
Method: Ignore										
Addresses										
<table border="1"> <thead> <tr> <th>Address</th> <th>Prefix</th> <th>Gateway</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>			Address	Prefix	Gateway				Add	Delete
Address	Prefix	Gateway								
DNS servers:						Routes...				
Search domains:						Cancel	Save			
IPv6 privacy extensions: Disabled										
<input type="checkbox"/> Require IPv6 addressing for this connection to complete										



7. Click on “Begin Installation”.
8. Set a good root password.
9. Once installation is complete, reboot the server, and remove the DVD/ISO from the server.
10. SSH in to server as root.
11. Stop and disable the firewalld service:

```
# systemctl disable firewalld.service
# systemctl stop firewalld.service
```

12. Disable SELINUX:

```
# setenforce 0
# vim /etc/sysconfig/selinux

SELINUX=permissive
```

13. Update all packages on the server:

```
# yum update
```

14. If running the control node on VMWare, install the VM tools:

```
# yum install open-vm-tools
```

15. We need persistent network interface names, so we'll configure udev to give us these. Replace `00:00:00:00:00:00` with the MAC addresses of your control node:

```
# vim /etc/udev/rules.d/90-persistent-net.rules

SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="eno*", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="eno*", NAME="eth1"
```

- Note: This file is case-sensitive, and the MAC addresses should be lower-case.

16. Rename the network interface configuration files to `eth0` and `eth1`. Replace `eno00000001` and `eno00000002` with the names of your control node's interfaces:

```
# cd /etc/sysconfig/network-scripts
# mv ifcfg-eno00000001 ifcfg-eth0
# mv ifcfg-eno00000002 ifcfg-eth1
```

17. Modify the interface configuration files, replacing any instances of `eno00000001` and `eno00000002` (or whatever your interface names are) with `eth0` and `eth1` respectively:

```
# vim ifcfg-eth0

NAME=eth0
DEVICE=eth0

# vim ifcfg-eth1

NAME=eth1
DEVICE=eth1
```

18. Reboot the control node:

```
# systemctl reboot
```

19. SSH back in as root after the reboot.

20. Check that ifconfig now shows `eth0` and `eth1`:

```
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 00:0c:29:d9:36:46 txqueuelen 1000  (Ethernet)
      RX packets 172313 bytes 34438137 (32.8 MiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 7298 bytes 1552292 (1.4 MiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 172.16.0.192 netmask 255.255.255.0 broadcast 172.16.0.255
      inet6 fe80::20c:29ff:fed9:3650 prefixlen 64 scopeid 0x20<link>
      ether 00:0c:29:d9:36:50 txqueuelen 1000  (Ethernet)
      RX packets 1487929 bytes 210511596 (200.7 MiB)
      RX errors 0 dropped 11 overruns 0 frame 0
      TX packets 781276 bytes 4320203416 (4.0 GiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 0 (Local Loopback)
        RX packets 2462286 bytes 3417529317 (3.1 GiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 2462286 bytes 3417529317 (3.1 GiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

21. Update the system hosts file with entries for all nodes:

```
# vim /etc/hosts

172.16.0.192 controller controller.openstack.lab.eco.rackspace.com
172.16.0.203 compute1 compute1.openstack.lab.eco.rackspace.com
172.16.0.204 compute1-vm compute1-vm.openstack.lab.eco.rackspace.com
172.16.0.195 compute2 compute2.openstack.lab.eco.rackspace.com
172.16.0.196 block1 block1.openstack.lab.eco.rackspace.com
172.16.0.197 object1 object1.openstack.lab.eco.rackspace.com
172.16.0.198 object2 object2.openstack.lab.eco.rackspace.com
```

22. Update the “Chrony” (NTP Server) configuration to allow connections from our other nodes:

```
# vim /etc/chrony.conf

Allow 172.16.0.0/24
```

23. Restart the Chrony service:

```
# systemctl restart chronyd.service
```

24. Enable the OpenStack-Liberty yum repository:

```
# yum install centos-release-openstack-liberty
```

25. Install the OpenStack client and SELINUX support:

```
# yum install python-openstackclient openstack-selinux
```


CHAPTER 3

3. Install core services on controller

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/environment-sql-database.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-nosql-database.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-messaging.html>

1. Install MariaDB:

```
# yum install mariadb mariadb-server MySQL-python
```

2. Set some needed MariaDB configuration parameters:

```
# vim /etc/my.cnf

bind-address = 172.16.0.192
default-storage-engine = innodb
innodb_file_per_table
collation-server = utf8_general_ci
init-connect = 'SET NAMES utf8'
character-set-server = utf8
```

3. Enable and start the MariaDB service:

```
# systemctl enable mariadb.service
# systemctl start mariadb.service
```

4. Initialize MariaDB security. Say ‘yes’ to all prompts, and set a good root password:

```
# mysql_secure_installation
```

5. Set up the MySQL client configuration. Replace *MYSQL_ROOT* with your own:

```
# vim /root/.my.cnf
```

```
[client]
user=root
password=*MYSQL_ROOT*
```

6. Confirm that you are able to connect to MySQL:

```
# mysql
> quit
```

7. Install RabbitMQ:

```
# yum install rabbitmq-server
```

8. Enable and start the RabbitMQ service:

```
# systemctl enable rabbitmq-server.service
# systemctl start rabbitmq-server.service
```

9. Create the “openstack” RabbitMQ user:

```
# rabbitmqctl add_user openstack *RABBIT_PASS*
# rabbitmqctl set_permissions openstack ".*" ".*" ".*"
```

CHAPTER 4

4. Install Identity (keystone) on controller

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/keystone-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/keystone-services.html>

<http://docs.openstack.org/liberty/install-guide-rdo/keystone-users.html>

<http://docs.openstack.org/liberty/install-guide-rdo/keystone-verify.html>

<http://docs.openstack.org/liberty/install-guide-rdo/keystone-openrc.html>

1. Open the MySQL client and create the “keystone” database. Replace *KEYSTONE_DBPASS* with your own:

```
# mysql
  > create database keystone;
  > grant all privileges on keystone.* to 'keystone'@'localhost' identified by
  ↵'*KEYSTONE_DBPASS*';
  > grant all privileges on keystone.* to 'keystone'@'%' identified by '*KEYSTONE_
  ↵DBPASS*';
  > quit
```

2. Install the keystone packages:

```
# yum install openstack-keystone httpd mod_wsgi memcached python-memcached
```

3. Enable and start the memcached service:

```
# systemctl enable memcached.service
# systemctl start memcached.service
```

4. Configure keystone. Replace *ADMIN_TOKEN* and *KEYSTONE_DBPASS* with your own:

```
# vim /etc/keystone/keystone.conf

[DEFAULT]
admin_token = *ADMIN_TOKEN*
```

```
[database]
connection = mysql://keystone:*KEYSTONE_DBPASS*@controller/keystone

[memcache]
servers = localhost:11211

[token]
provider = uuid
driver = memcache
expiration = 86400

[revoke]
driver = sql
```

- Note: I have extended token expiration to 24-hours, due to issues I experienced with large images timing-out during the saving process. You may wish to use a shorter expiration, depending on your security requirements.

5. Populate the keystone database:

```
# su -s /bin/sh -c "keystone-manage db_sync" keystone
```

6. Set the Apache server name:

```
# vim /etc/httpd/conf/httpd.conf

ServerName controller
```

7. Configure wsgi:

```
# vim /etc/httpd/conf.d/wsgi-keystone.conf

Listen 5000
Listen 35357

<VirtualHost *:5000>
    WSGIDaemonProcess keystone-public processes=5 threads=1 user=keystone
    ↪group=keystone display-name=%{GROUP}
        WSGIProcessGroup keystone-public
        WSGIScriptAlias / /usr/bin/keystone-wsgi-public
        WSGIAccessLog /var/log/httpd/keystone-access.log
        WSGIPassAuthorization On
        <IfVersion >= 2.4>
            ErrorLogFormat "%{cu}t %M"
        </IfVersion>
        ErrorLog /var/log/httpd/keystone-error.log
        CustomLog /var/log/httpd/keystone-access.log combined

        <Directory /usr/bin>
            <IfVersion >= 2.4>
                Require all granted
            </IfVersion>
            <IfVersion < 2.4>
                Order allow,deny
                Allow from all
            </IfVersion>
        </Directory>
    </VirtualHost>
```

```
<VirtualHost *:35357>
    WSGIDaemonProcess keystone-admin processes=5 threads=1 user=keystone
    <!--group=keystone display-name=%{GROUP}-->
    WSGIProcessGroup keystone-admin
    WSGIScriptAlias / /usr/bin/keystone-wsgi-admin
    WSGIApplicationGroup %{GLOBAL}
    WSGIPassAuthorization On
    <IfVersion >= 2.4>
        ErrorLogFormat "%{cu}t %M"
    </IfVersion>
    ErrorLog /var/log/httpd/keystone-error.log
    CustomLog /var/log/httpd/keystone-access.log combined

    <Directory /usr/bin>
        <IfVersion >= 2.4>
            Require all granted
        </IfVersion>
        <IfVersion < 2.4>
            Order allow,deny
            Allow from all
        </IfVersion>
    </Directory>
</VirtualHost>
```

8. Enable and start the Apache service:

```
# systemctl enable httpd.service
# systemctl start httpd.service
```

9. Set up temporary connection parameters. Replace *ADMIN_TOKEN* with your own:

```
# export OS_TOKEN=*ADMIN_TOKEN*
# export OS_URL=http://controller:35357/v3
# export OS_IDENTITY_API_VERSION=3
```

10. Create keystone service and endpoints:

```
# openstack service create --name keystone --description "OpenStack Identity"
    <!--identity-->
# openstack endpoint create --region RegionOne identity public http://
    <!--controller:5000/v2.0-->
# openstack endpoint create --region RegionOne identity internal http://
    <!--controller:5000/v2.0-->
# openstack endpoint create --region RegionOne identity admin http://
    <!--controller:35357/v2.0-->
```

11. Create the “admin” project, user and role. Provide your *ADMIN_PASS* twice when prompted:

```
# openstack project create --domain default --description "Admin Project" admin
# openstack user create --domain default --password-prompt admin
# openstack role create admin
# openstack role add --project admin --user admin admin
```

12. Create the “service” project:

```
# openstack project create --domain default --description "Service Project"
    <!--service-->
```

13. Create the “demo” project, user and role. Provide your *DEMO_PASS* twice when prompted:

```
# openstack project create --domain default --description "Demo Project" demo
# openstack user create --domain default --password-prompt demo
# openstack role create user
# openstack role add --project demo --user demo user
```

14. Disable authentication with the admin token:

```
# vim /usr/share/keystone/keystone-dist-paste.ini
```

- Remove `admin_token_auth` from `[pipeline:public_api]`, `[pipeline:admin_api]` and `[pipeline:api_v3]`

15. Disable the temporary connection parameters:

```
# unset OS_TOKEN OS_URL
```

16. Test authentication for the “admin” user. Provide *ADMIN_PASS* when prompted:

```
# openstack --os-auth-url http://controller:35357/v3 --os-project-domain-id_
˓→default --os-user-domain-id default --os-project-name admin --os-username admin_
˓→--os-auth-type password token issue
```

- If this is working, various values will be returned (yours will be different):

Field	Value
expires	2016-02-05T22:55:18.580385Z
id	9bd8b09e4fdd43cealf32ca6d62c946b
project_id	76f8c8fd7b1e407d97c4604eb2a408b3
user_id	31766cbe74d541088c6ba2fd24654034

17. Test authentication for the “demo” user. Provide *DEMO_PASS* when prompted:

```
# openstack --os-auth-url http://controller:5000/v3 --os-project-domain-id_
˓→default --os-user-domain-id default --os-project-name demo --os-username demo --
˓→--os-auth-type password token issue
```

- Again, if this is working, various values will be returned.

18. Create permanent client authentication file for the “admin” user. Replace *ADMIN_PASS* with your own:

```
# vim /root/admin-openrc.sh

export OS_PROJECT_DOMAIN_ID=default
export OS_USER_DOMAIN_ID=default
export OS_PROJECT_NAME=admin
export OS_TENANT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=*ADMIN_PASS*
export OS_AUTH_URL=http://controller:35357/v3
export OS_IDENTITY_API_VERSION=3
```

19. Create permanent client authentication file for the “demo” user. Replace *DEMO_PASS* with your own:

```
# vim /root/demo-openrc.sh

export OS_PROJECT_DOMAIN_ID=default
export OS_USER_DOMAIN_ID=default
export OS_PROJECT_NAME=demo
export OS_TENANT_NAME=demo
export OS_USERNAME=demo
export OS_PASSWORD=*DEMO_PASS*
export OS_AUTH_URL=http://controller:5000/v3
export OS_IDENTITY_API_VERSION=3
```

20. Test authentication with the permanent settings:

```
# source admin-openrc.sh
# openstack token issue
```

- Once more, if this works, various values will be returned.

CHAPTER 5

5. Install Images (glance) on controller

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/glance-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/glance-verify.html>

Step 9 has specific changes for the use of XenServer.

1. Open the MySQL client and create the “glance” database. Replace *GLANCE_DBPASS* with your own:

```
# mysql
  > create database glance;
  > grant all privileges on glance.* to 'glance'@'localhost' identified by
  ↵'*GLANCE_DBPASS*';
  > grant all privileges on glance.* to 'glance'@'%' identified by '*GLANCE_
  ↵DBPASS*';
  > quit
```

2. Create the “glance” user, role, service and endpoints. Provide *GLANCE_PASS* when prompted:

```
# source admin-openrc.sh
# openstack user create --domain default --password-prompt glance
# openstack role add --project service --user glance admin
# openstack service create --name glance --description "OpenStack Image service" ↵
  ↵image
# openstack endpoint create --region RegionOne image public http://controller:9292
# openstack endpoint create --region RegionOne image internal http://
  ↵controller:9292
# openstack endpoint create --region RegionOne image admin http://controller:9292
```

3. Install glance packages:

```
# yum install openstack-glance python-glance python-glanceclient
```

4. Configure glance-api. Replace *GLANCE_DBPASS* and *GLANCE_PASS* with your own:

```
# vim /etc/glance/glance-api.conf

[database]
connection = mysql://glance:*GLANCE_DBPASS*@controller/glance

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = glance
password = *GLANCE_PASS*

[paste_deploy]
flavor = keystone

[glance_store]
default_store = file
filesystem_store_datadir = /var/lib/glance/images/

[DEFAULT]
notification_driver = noop
```

5. Configure glance-registry. Replace ***GLANCE_DBPASS*** and ***GLANCE_PASS*** with your own:

```
# vim /etc/glance/glance-registry.conf

[database]
connection = mysql://glance:*GLANCE_DBPASS*@controller/glance

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = glance
password = *GLANCE_PASS*

[paste_deploy]
flavor=keystone

[DEFAULT]
notification_driver = noop
```

6. Populate the glance database:

```
# su -s /bin/sh -c "glance-manage db_sync" glance
```

- Note: “No handlers could be found for logger” warnings are normal, and can be ignored.

7. Enable and start the glance service:

```
# systemctl enable openstack-glance-api.service openstack-glance-registry.service
# systemctl start openstack-glance-api.service openstack-glance-registry.service
```

8. Add glance API version settings to the client authentication files:

```
# echo "export OS_IMAGE_API_VERSION=2" | tee -a admin-openrc.sh demo-openrc.sh
```

9. Upload a sample image to the glance service:

```
# source admin-openrc.sh
# wget http://ca.downloads.xensource.com/OpenStack/cirros-0.3.4-x86_64-disk.vhd.
˓→tgz
# glance image-create --name "cirros-xen" --container-format ovf --disk-format
˓→vhd --property vm_mode=xen --visibility public --file cirros-0.3.4-x86_64-disk.
˓→vhd.tgz
```

10. Confirm that the image has been uploaded:

```
# glance image-list

+-----+-----+
| ID          | Name      |
+-----+-----+
| 1e710e0c-0fb6-4425-b196-4b66bfac495e | cirros-xen |
+-----+-----+
```


CHAPTER 6

6. Install Compute (nova) on controller

This page is based on the following OpenStack Installation Guide page:

<http://docs.openstack.org/liberty/install-guide-rdo/nova-controller-install.html>

1. Open the MySQL client and create the “nova” database. Replace *NOVA_DBPASS* with your own:

```
# mysql  
  
> create database nova;  
> grant all privileges on nova.* to 'nova'@'localhost' identified by '*NOVA_  
-DBPASS*';  
> grant all privileges on nova.* to 'nova'@'%' identified by '*NOVA_DBPASS*';  
> quit
```

2. Create the “nova” user, role, service and endpoints. Provide *NOVA_PASS* when prompted:

```
# source admin-openrc.sh  
# openstack user create --domain default --password-prompt nova  
# openstack role add --project service --user nova admin  
# openstack service create --name nova --description "OpenStack Compute" compute  
# openstack endpoint create --region RegionOne compute public http://  
-controller:8774/v2/%\{(tenant_id\}\s  
# openstack endpoint create --region RegionOne compute internal http://  
-controller:8774/v2/%\{(tenant_id\}\s  
# openstack endpoint create --region RegionOne compute admin http://  
-controller:8774/v2/%\{(tenant_id\}\s
```

3. Install nova packages:

```
# yum install openstack-nova-api openstack-nova-cert openstack-nova-conductor  
-openstack-nova-console openstack-nova-novncproxy openstack-nova-scheduler  
-python-novaclient
```

4. Configure nova. Replace *NOVA_DBPASS*, *NOVA_PASS*, *SERVER_IP* and *RABIT_PASS* with your own:

```
# vim /etc/nova/nova.conf

[database]
connection = mysql://nova:*NOVA_DBPASS*@controller/nova

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
my_ip = *SERVER_IP*
network_api_class = nova.network.neutronv2.api.API
security_group_api = neutron
linuxnet_interface_driver = nova.network.linux_net.
                             NeutronLinuxBridgeInterfaceDriver
firewall_driver = nova.virt.firewall.NoopFirewallDriver
enabled_apis = osapi_compute,metadata

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = nova
password = *NOVA_PASS*

[vnc]
vncserver_listen = $my_ip
vncserver_proxyclient_address = $my_ip

[glance]
host = controller

[oslo_concurrency]
lock_path = /var/lib/nova/tmp
```

5. Populate the nova database:

```
# su -s /bin/sh -c "nova-manage db sync" nova
```

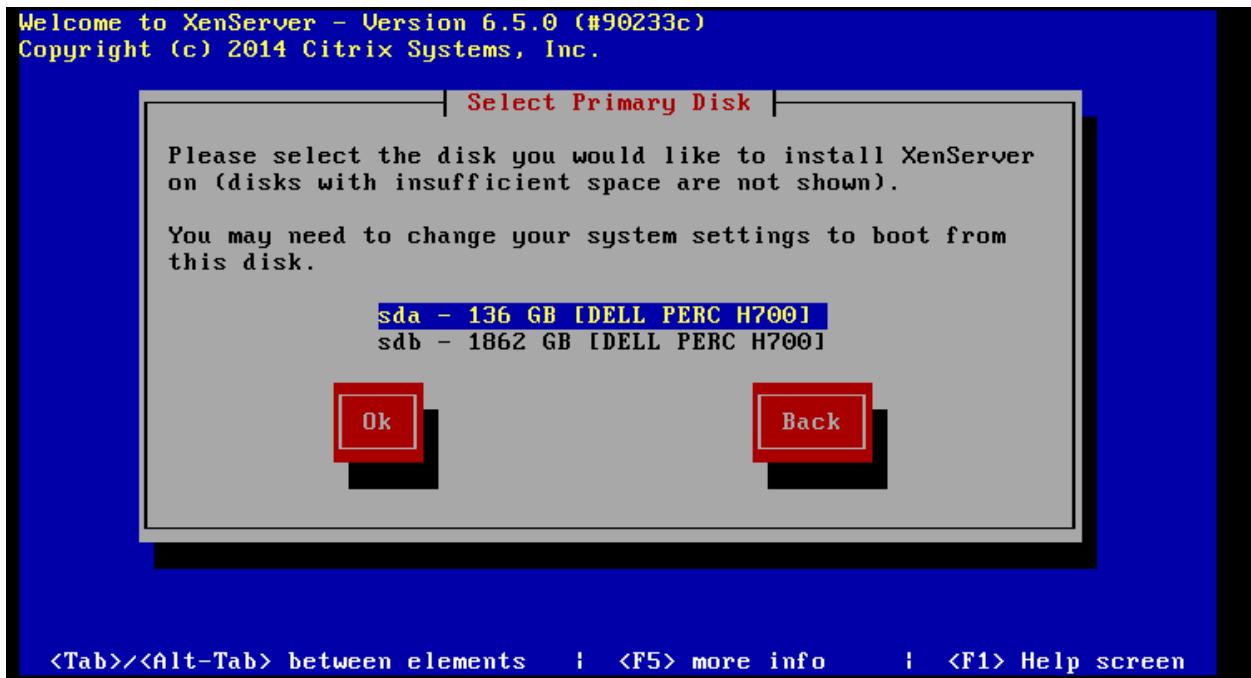
6. Enable and start the nova service:

```
# systemctl enable openstack-nova-api.service openstack-nova-cert.service
  ↪openstack-nova-consoleauth.service openstack-nova-scheduler.service openstack-
  ↪nova-conductor.service openstack-nova-novncproxy.service
# systemctl start openstack-nova-api.service openstack-nova-cert.service
  ↪openstack-nova-consoleauth.service openstack-nova-scheduler.service openstack-
  ↪nova-conductor.service openstack-nova-novncproxy.service
```

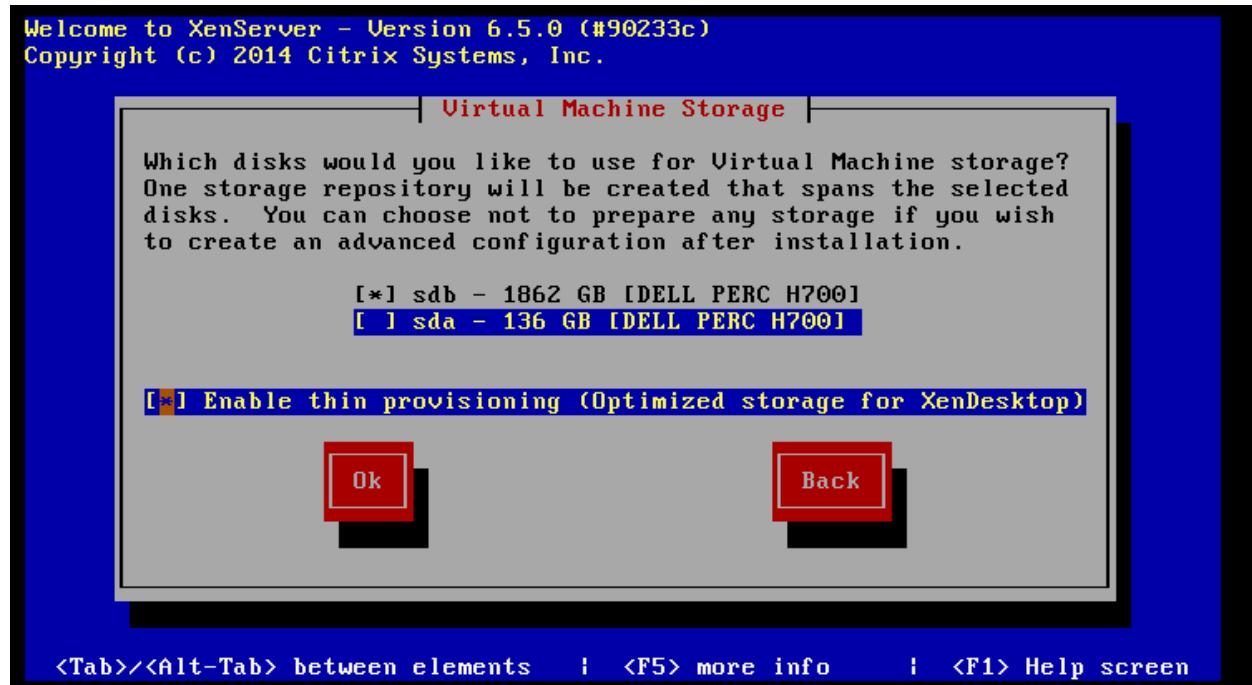
7. Build XenServer Host

This page is not based on the OpenStack Installation Guide.

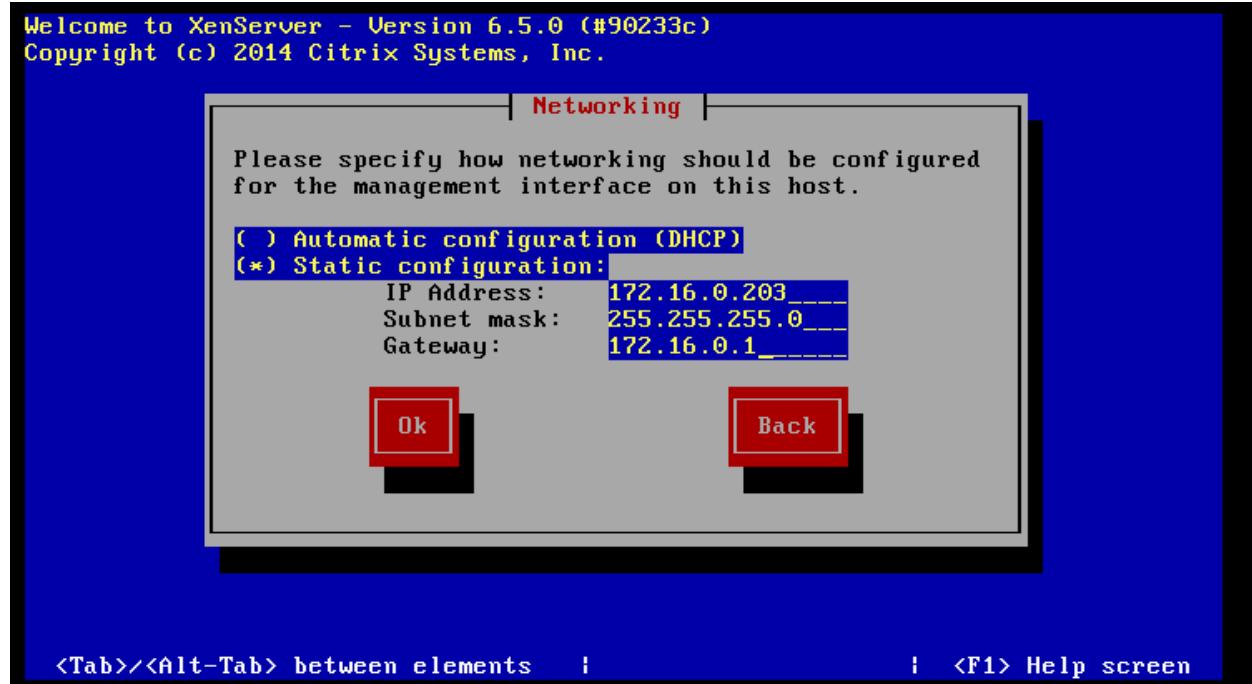
1. In this guide I am using a server with a small RAID-1 for the OS, and a large RAID-10 for the VMs.
2. Boot with XenServer 6.5 DVD.
3. Set keyboard, agree to terms, etc.
4. Set the installation destination to sda.

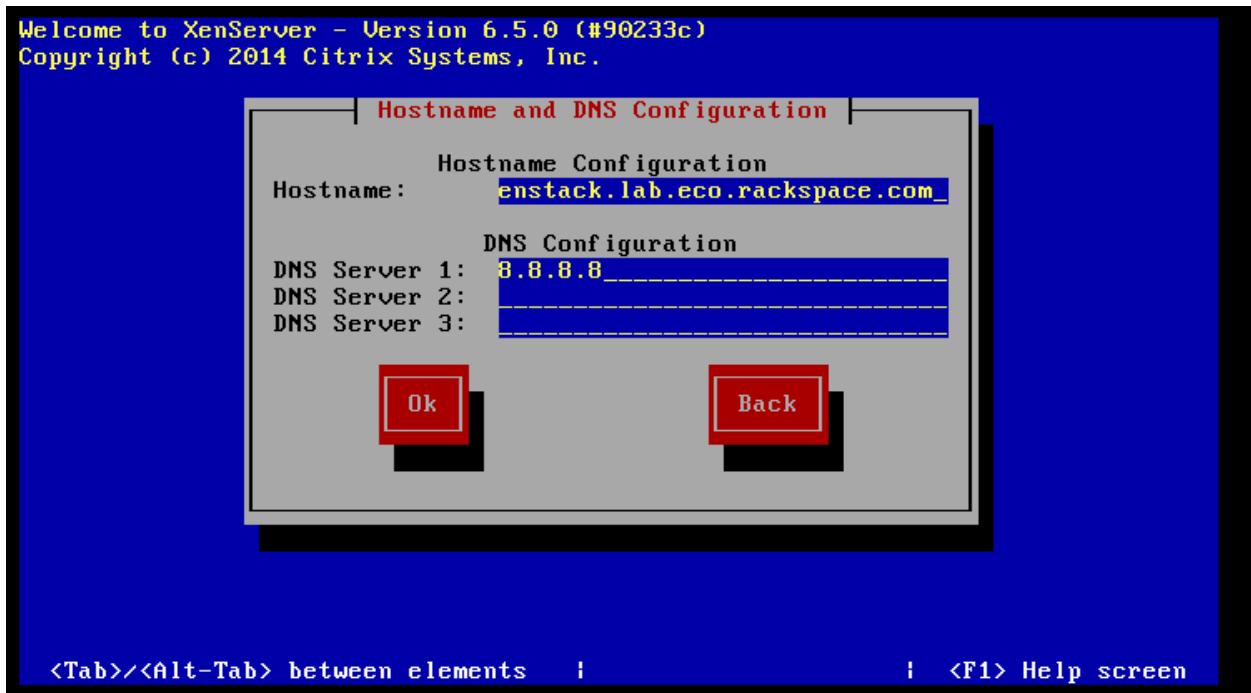


5. Set VM storage to only sdb, and enable thin provisioning:

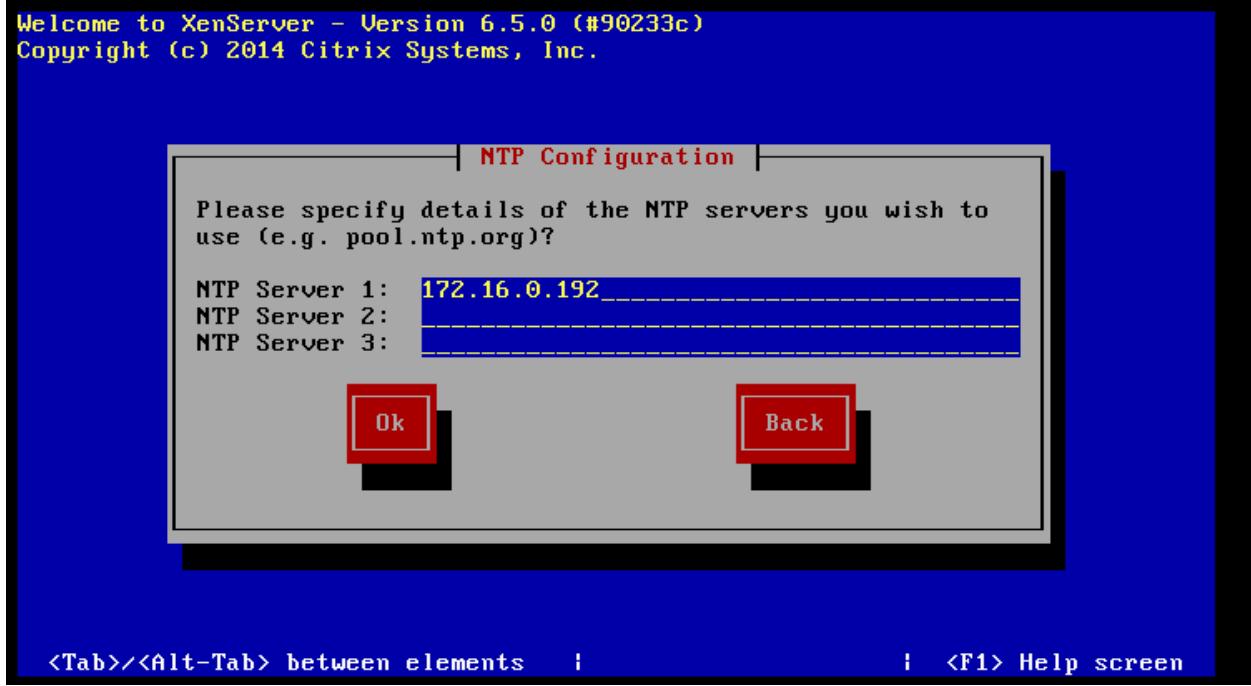


6. Select local media as the installation source.
7. Do not install any supplemental packs.
8. Skip verification of the installation media.
9. Set a good *XENSERVER_ROOT* password. Use a password which you don't mind being plain-text readable to anyone who has root access to this system.
10. Set the management network interface to use eth1 and configure the IPv4 addresses:



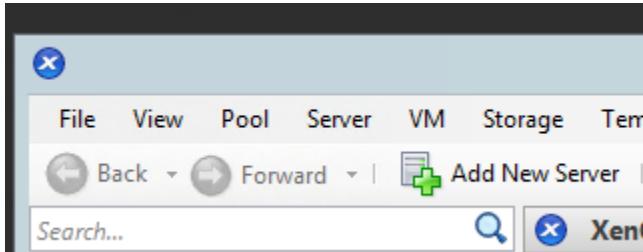


11. Set an appropriate timezone.
12. Configure the server to use NTP, and set the server address as the controller's IP:

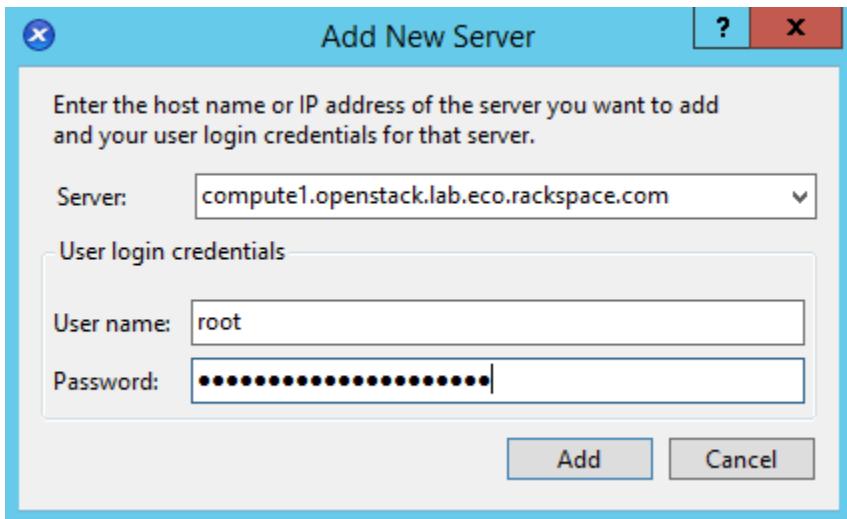


13. Start the installation.
14. Reboot the server to start XenServer. The first boot will take a very long time. It will appear to hang a couple of times, but wait for it to reach the user interface.
15. On a Windows workstation, go to <http://xenserver.org/open-source-virtualization-download.html>
16. Download XenCenter Windows Management Console, and install it.

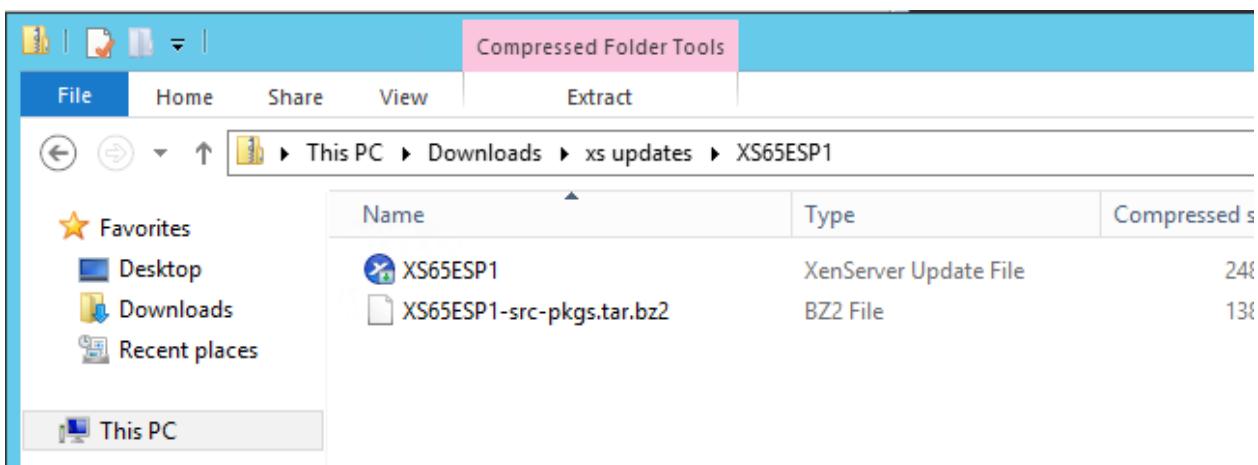
17. Download XenServer 6.5 SP1 (under Service Packs), and keep it safe in a directory.
18. Download all of the public hotfixes for XenServer 6.5 SP1, and also keep them safe in a directory.
19. Launch XenCenter, and click add new server:



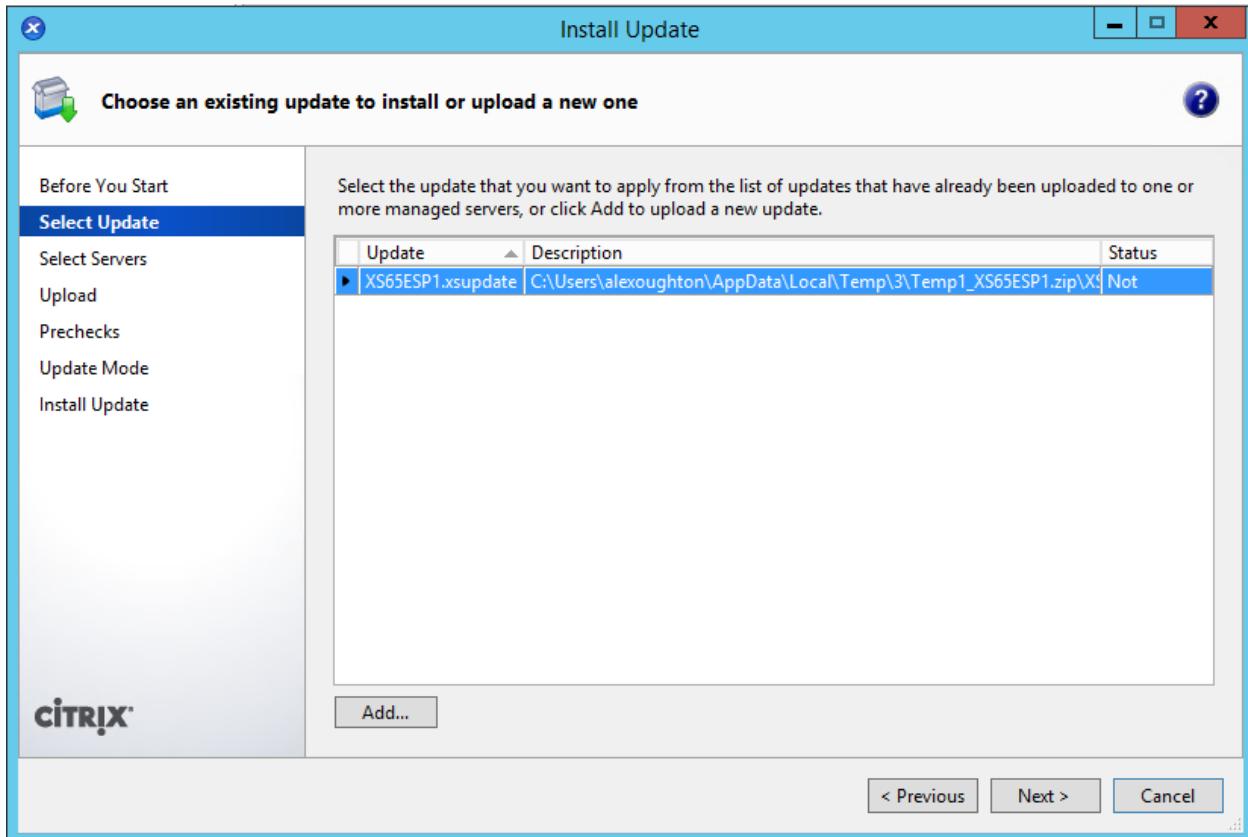
20. Enter the address and credentials of the XenServer:



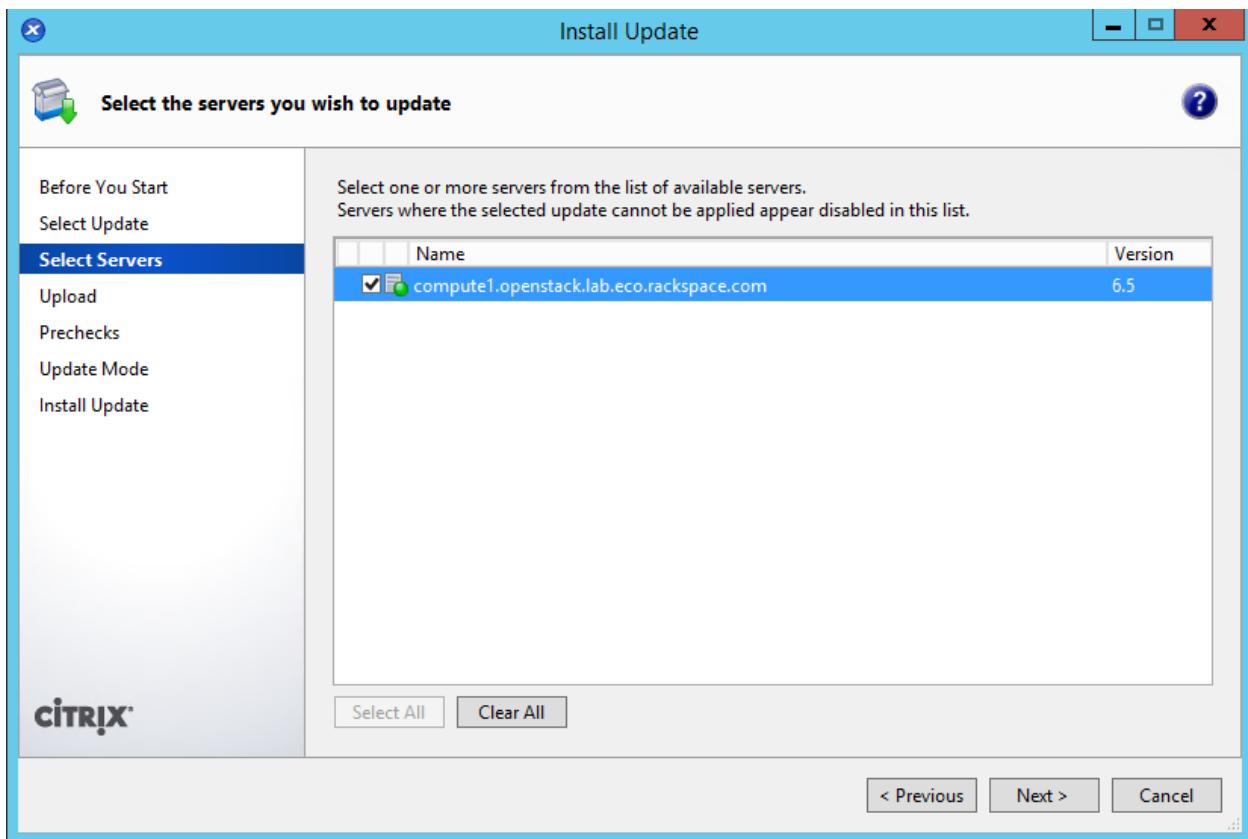
21. Enable the option to remember the connection, and click OK.
22. Open up the SP1 zip file you downloaded, and double-click the XenServer Update File inside:



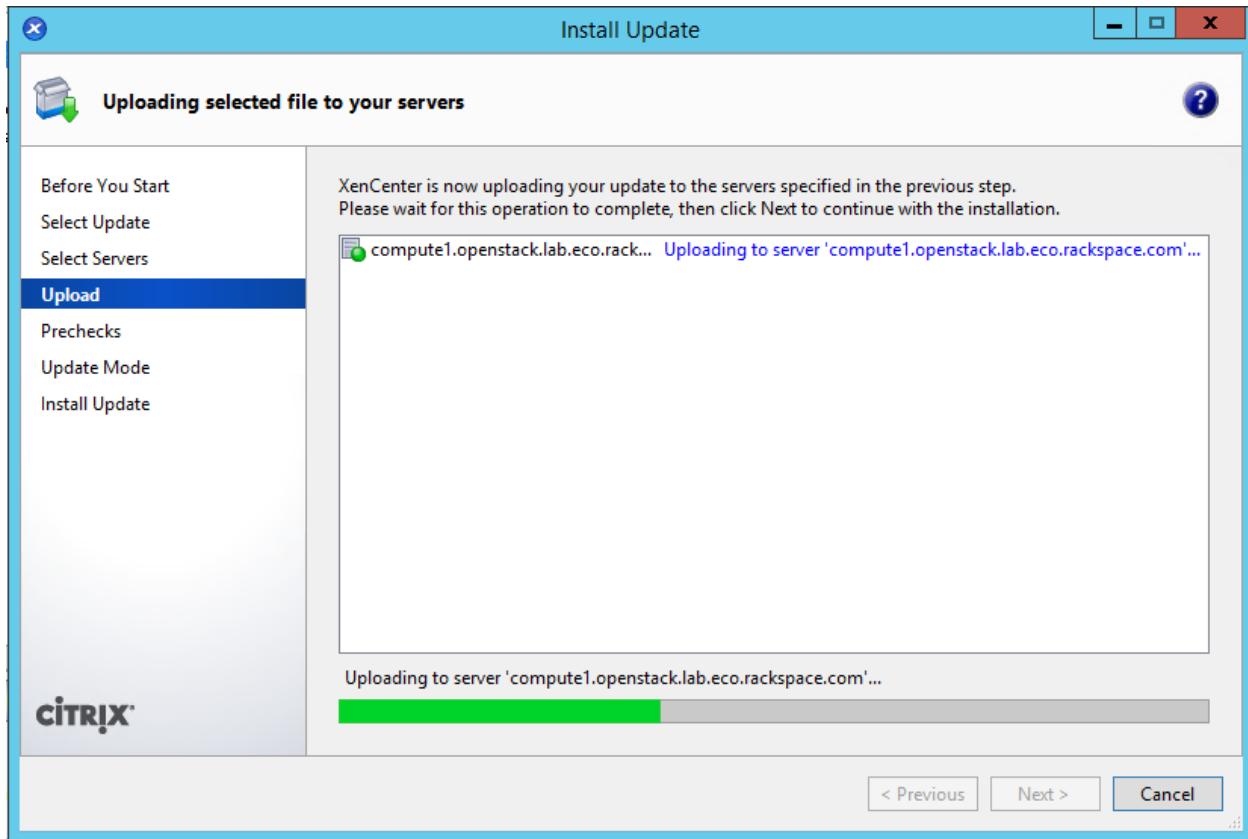
23. This will open the Install Update wizard. Click Next:



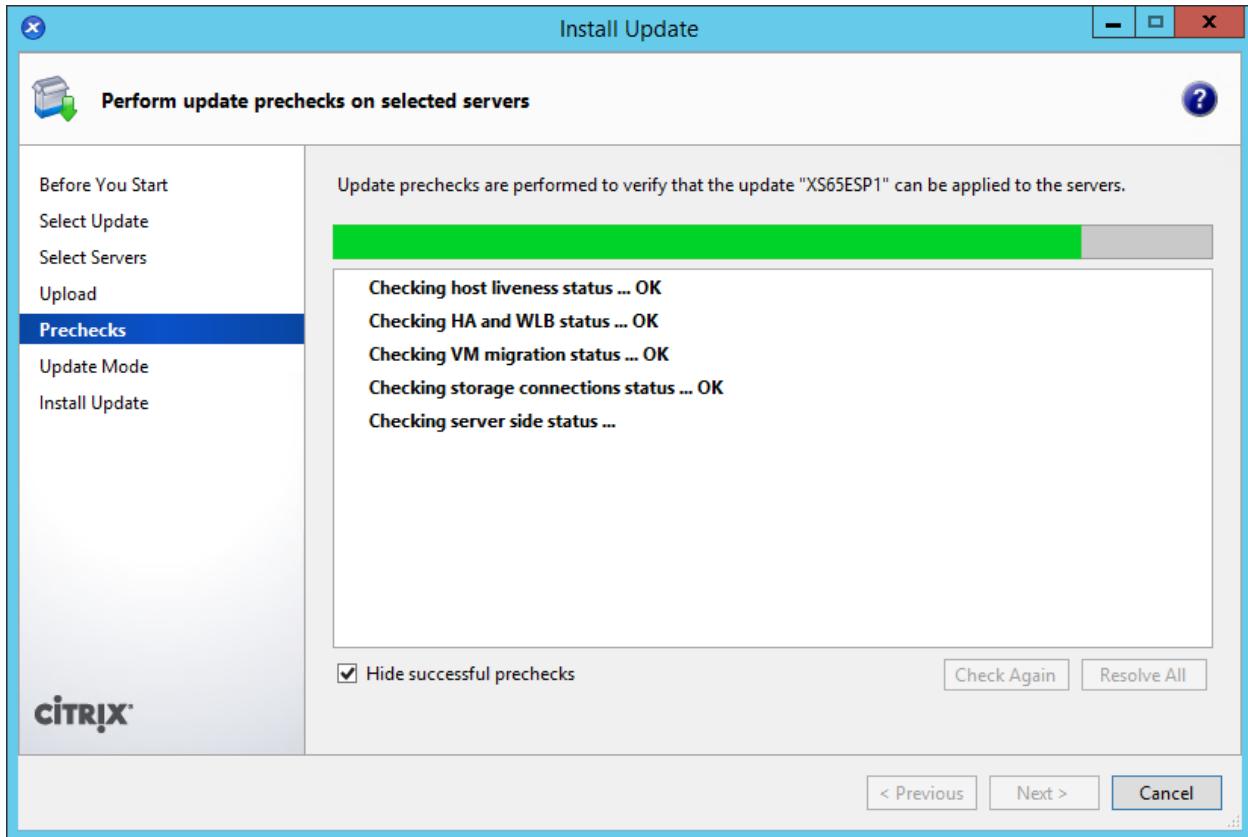
24. Select our one server, and click next:



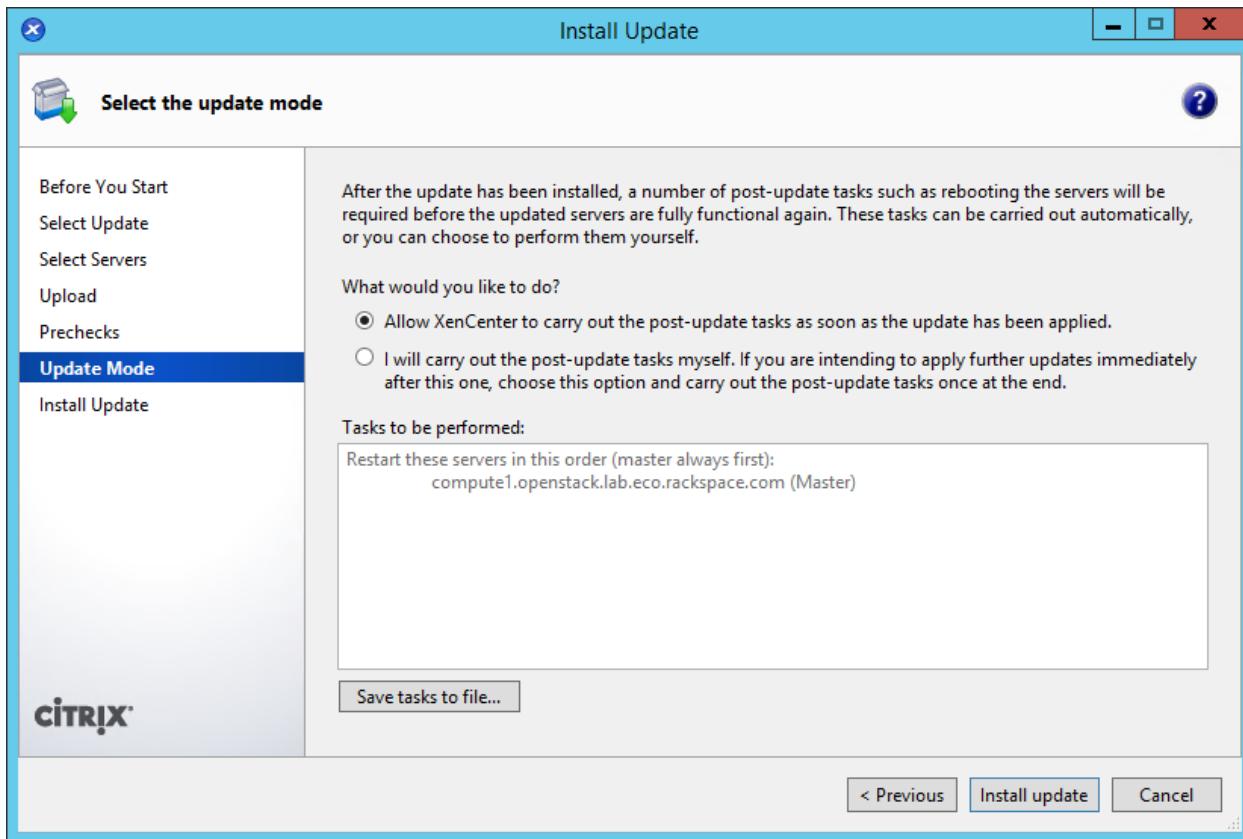
25. XenCenter will upload the update to the server. Click next when done:



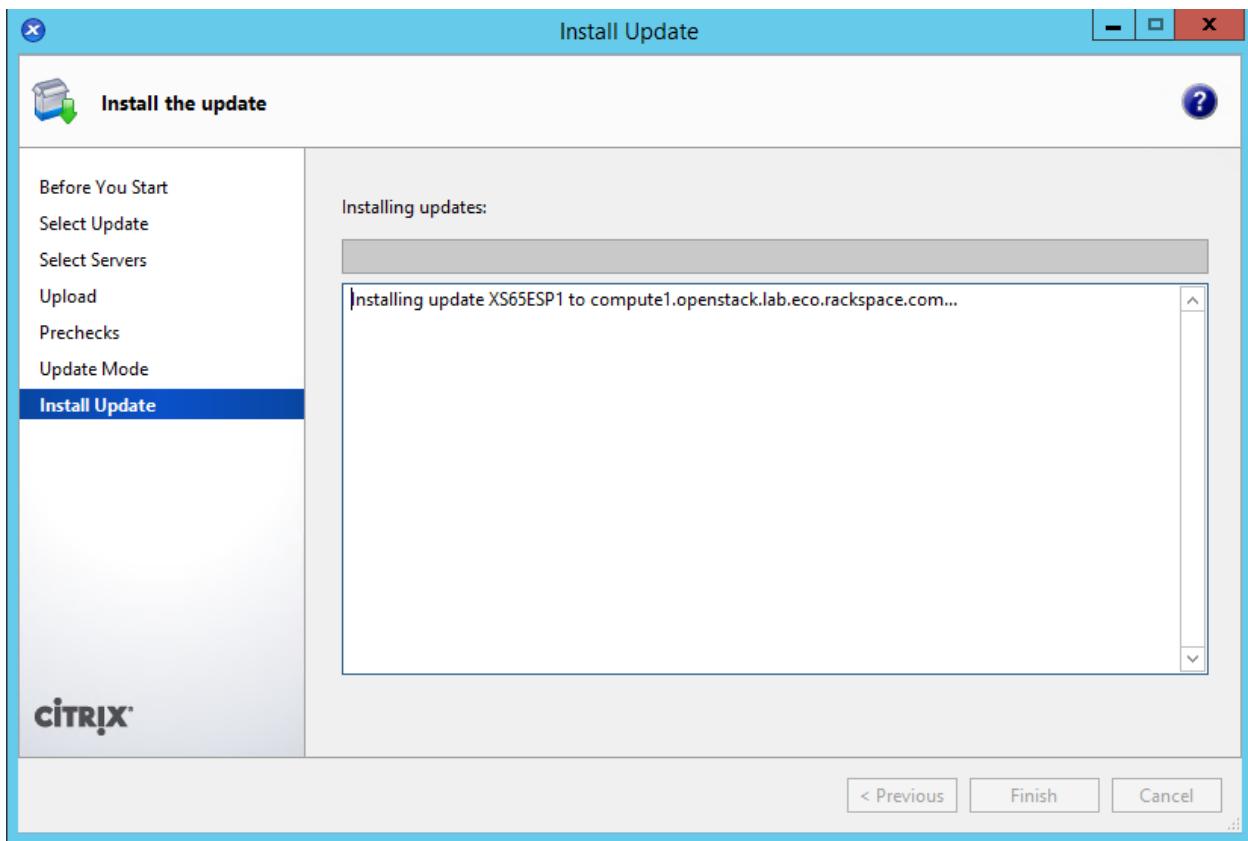
26. XenCenter will run some checks. Click next when done:



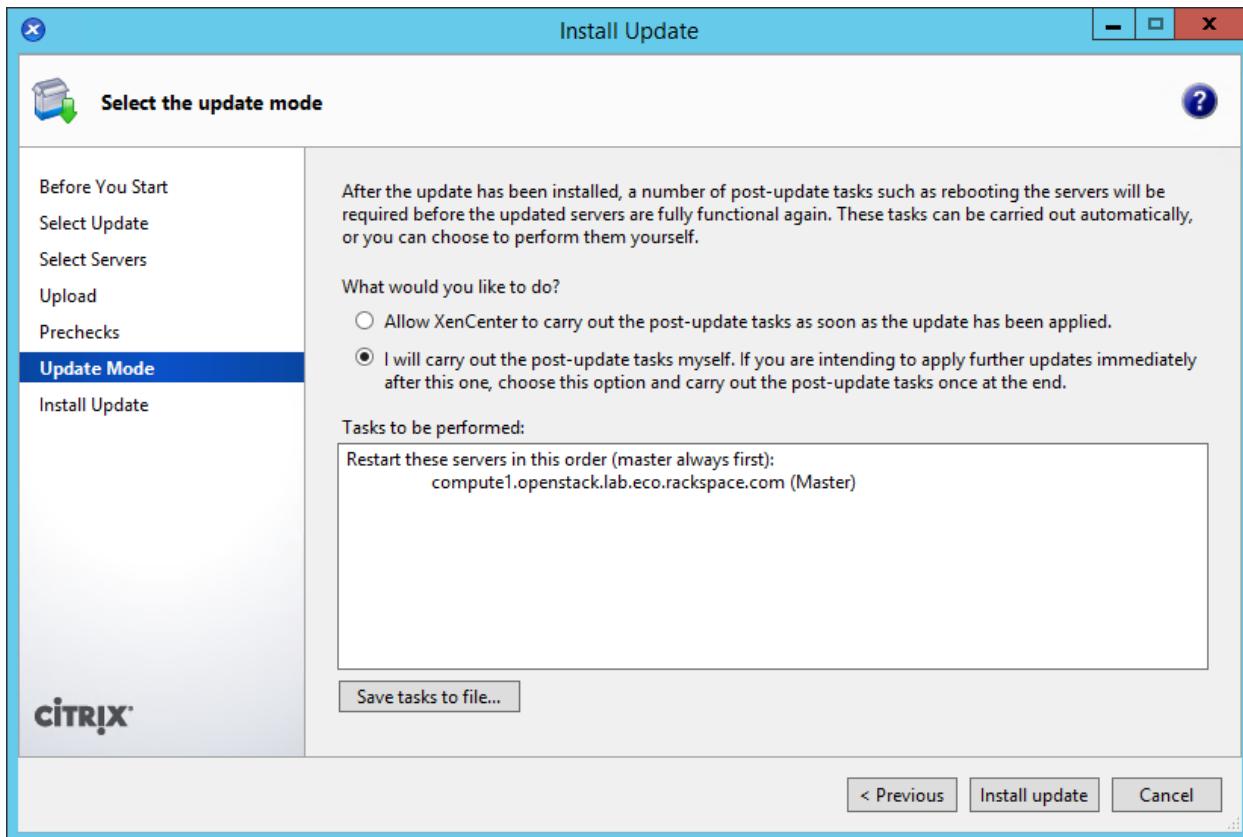
26. Select “Allow XenCenter to carry out the post-update tasks”, and then click on “Install Update”:



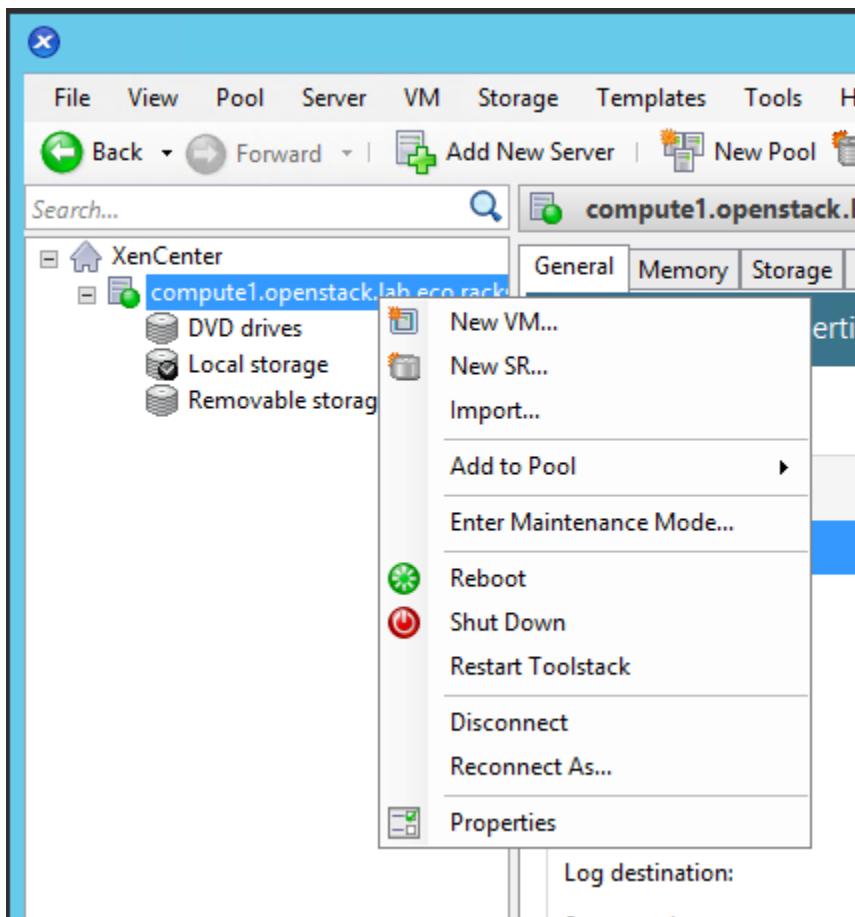
27. XenCenter will perform the installation, and reboot the server. This will take a while to complete. Click Finish when done:



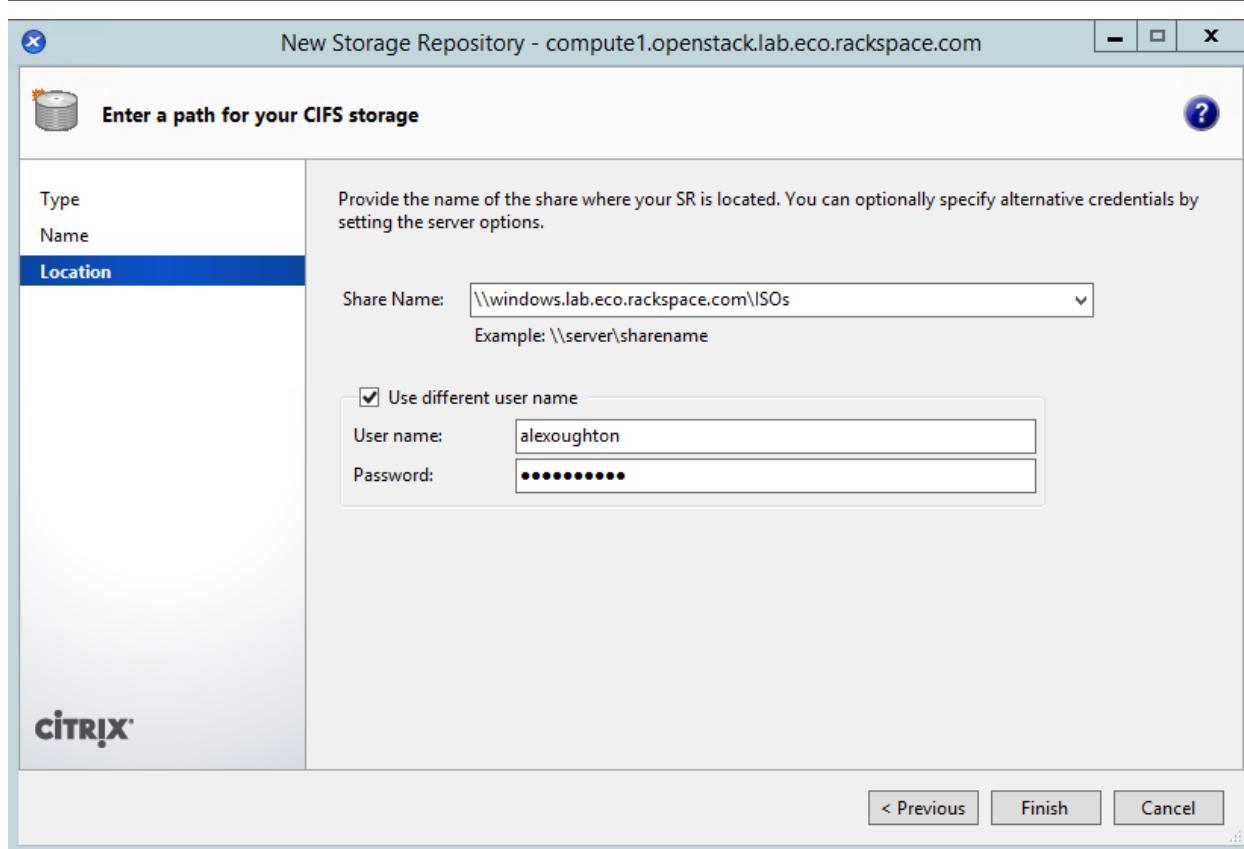
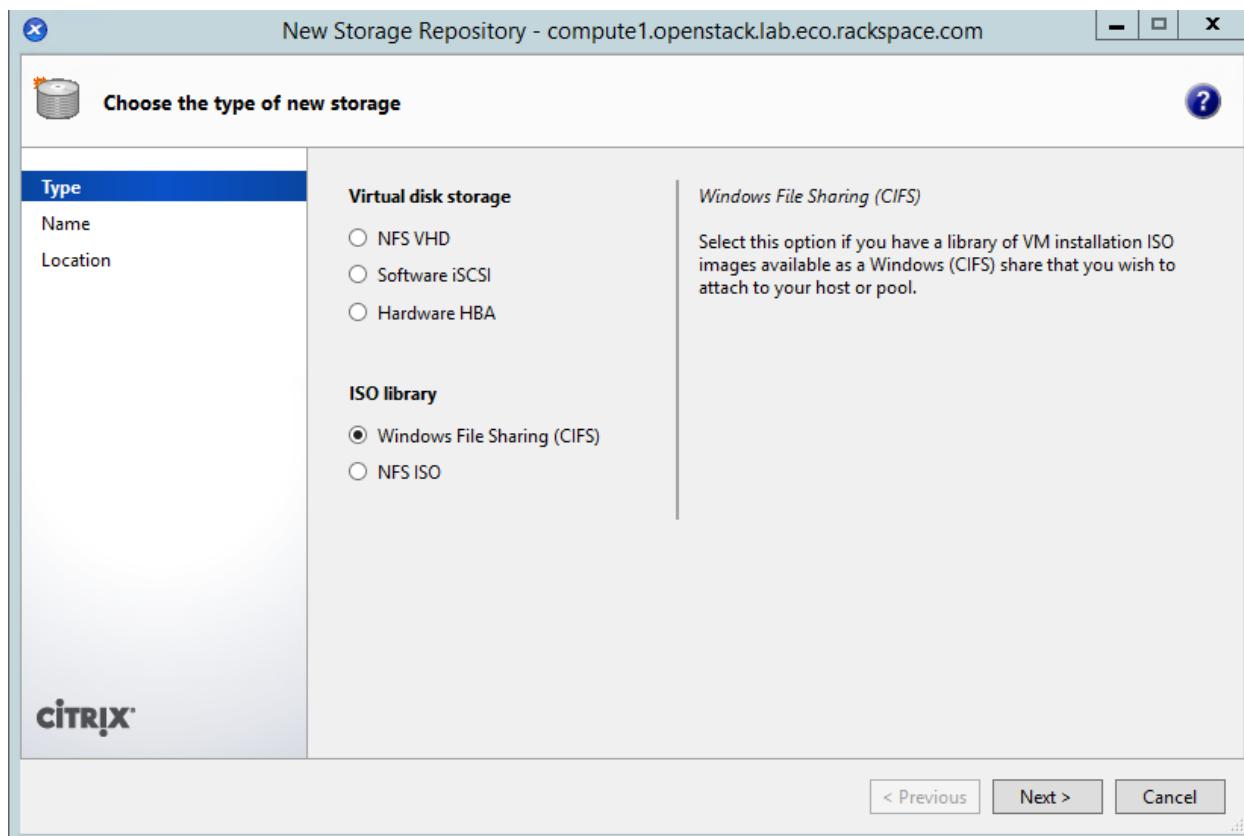
28. Repeat steps 22-27 for all of the hotfixes you downloaded. Except in step 26, select “I will carry out the post-update checks myself” for ALL of the hotfixes:



29. Reboot the XenServer by right-clicking it in XenCenter, and clicking on “Reboot”:



30. Once the server is back online, right-click it and select “New SR...”
31. Create an ISO library somewhere where you will have read/write access. In my case I am using a Windows share, but you can use NFS:



32. SSH to the XenServer as root.

33. Create the OpenStack Integration Bridge network:

```
# xe network-create name-label=openstack-int-network
```

34. Obtain the bridge name of the new network. Write this down as *XAPI_BRIDGE*, as this will be needed later:

```
# xe network-list name-label=openstack-int-network params=bridge  
bridge ( RO) : xapi0
```

35. Find the UUID of the ISO library created earlier:

```
# xe sr-list  
  
uuid ( RO) : ef0adc0a-3b56-5e9d-4824-0821f4be7ed4  
    name-label ( RW): Removable storage  
    name-description ( RW):  
        host ( RO): compute1.openstack.lab.eco.rackspace.com  
        type ( RO): udev  
    content-type ( RO): disk  
  
uuid ( RO) : 6658e157-a534-a450-c4db-2ca6dd6296cf  
    name-label ( RW): Local storage  
    name-description ( RW):  
        host ( RO): compute1.openstack.lab.eco.rackspace.com  
        type ( RO): ext  
    content-type ( RO): user  
  
uuid ( RO) : f04950c1-ee7b-2ccb-e3e2-127a5bffc5a6  
    name-label ( RW): CIFS ISO library  
    name-description ( RW): CIFS ISO Library [\\windows.lab.eco.rackspace.  
com\ISOs]  
        host ( RO): compute1.openstack.lab.eco.rackspace.com  
        type ( RO): iso  
    content-type ( RO): iso  
  
uuid ( RO) : 7a549ca7-d1af-cf72-fd7e-2f48448354e8  
    name-label ( RW): DVD drives  
    name-description ( RW): Physical DVD drives  
        host ( RO): compute1.openstack.lab.eco.rackspace.com  
        type ( RO): udev  
    content-type ( RO): iso  
  
uuid ( RO) : 9a4f8404-7745-b582-484f-108917bf1488  
    name-label ( RW): XenServer Tools  
    name-description ( RW): XenServer Tools ISOs  
        host ( RO): compute1.openstack.lab.eco.rackspace.com  
        type ( RO): iso  
    content-type ( RO): iso
```

- In my example, the UUID is f04950c1-ee7b-2ccb-e3e2-127a5bffc5a6.

36. Set a parameter on the ISO library. Replace *UUID* with the UUID found above:

```
# xe sr-param-set uuid=*UUID* other-config:i18n-key=local-storage-iso
```

37. Update the system hosts file with entries for all nodes:

```
# vi /etc/hosts

172.16.0.192 controller controller.openstack.lab.eco.rackspace.com
172.16.0.203 compute1 compute1.openstack.lab.eco.rackspace.com
172.16.0.204 compute1-vm compute1-vm.openstack.lab.eco.rackspace.com
172.16.0.195 compute2 compute2.openstack.lab.eco.rackspace.com
172.16.0.196 block1 block1.openstack.lab.eco.rackspace.com
172.16.0.197 object1 object1.openstack.lab.eco.rackspace.com
172.16.0.198 object2 object2.openstack.lab.eco.rackspace.com
```

38. Relax XSM SR checks. Needed for migration of instances with Cinder volumes:

```
# vi /etc/xapi.conf

relax-xsm-sr-check = true
```

39. Symlink a directory of the SR to /images. Needed for instance migration:

```
# LOCAL_SR=$(xe sr-list name-label="Local storage" --minimal)
# IMG_DIR="/var/run/sr-mount/$LOCAL_SR/images"
# mkdir -p "$IMG_DIR"
# ln -s "$IMG_DIR" /images
```

40. Set up SSH key authentication for the root user. Needed for instance migration. Press ENTER to give default response to all prompts:

```
# ssh-keygen

Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.

# cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
```

- Note: If you are building an additional XenServer host, you will instead copy the contents of /root/.ssh from your first XenServer host to your additional hosts.

41. Restart the XenServer Toolstack:

```
# xe-toolstack-restart
```


CHAPTER 8

8. Build XenServer Compute VM

This page is based on the following OpenStack Installation Guide pages:

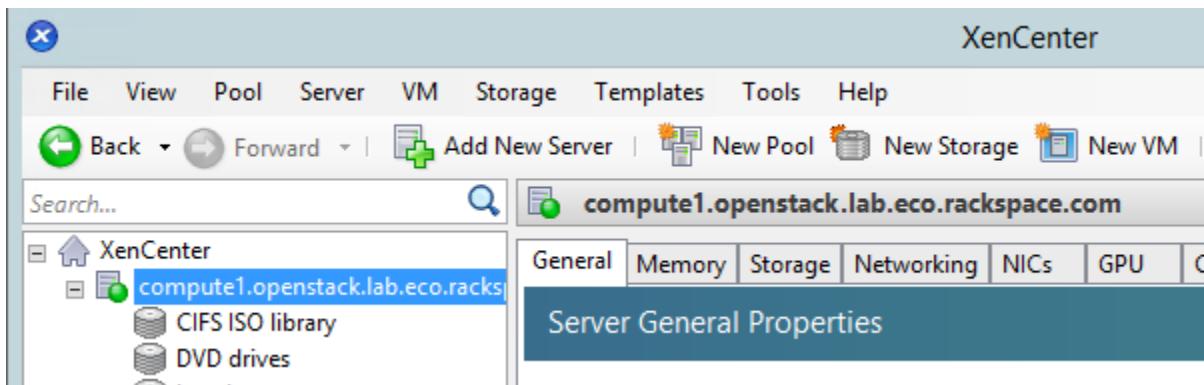
<http://docs.openstack.org/liberty/install-guide-rdo/environment-networking-compute.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-ntp-other.html>

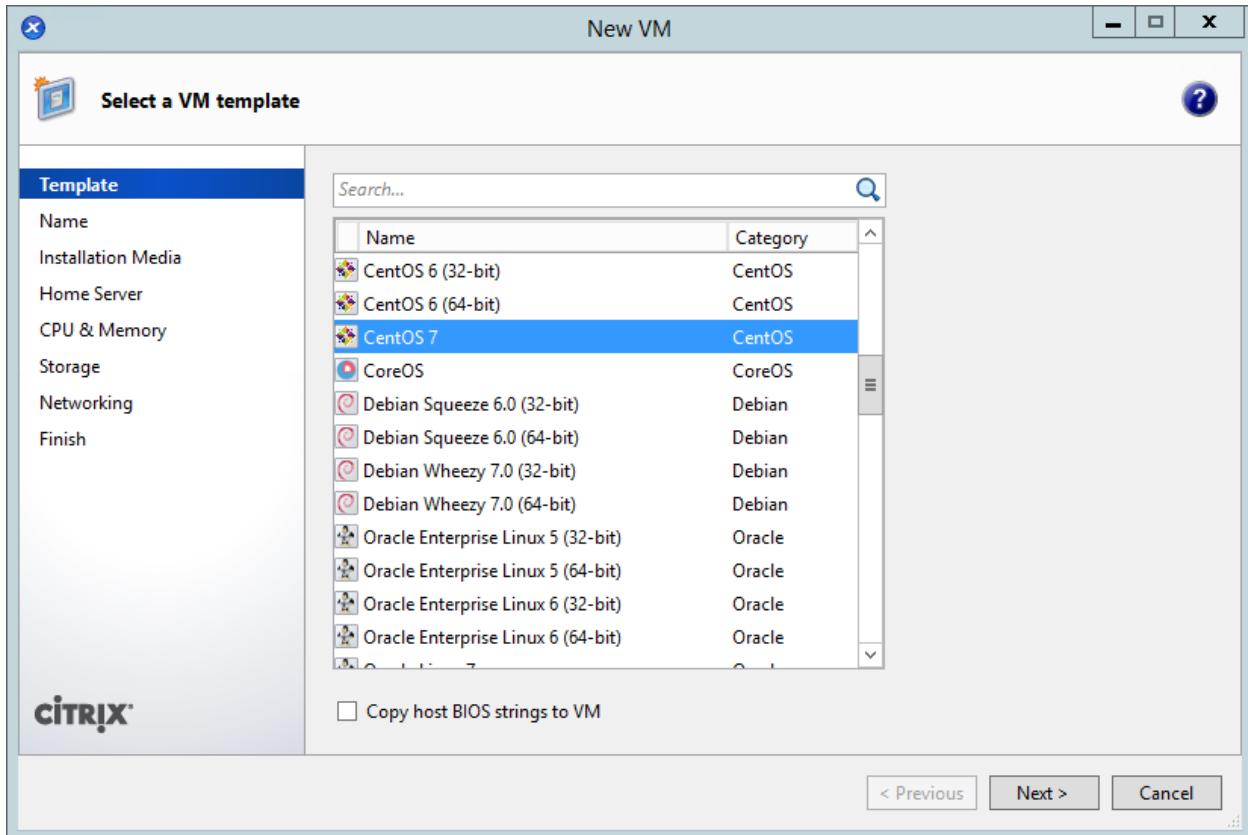
<http://docs.openstack.org/liberty/install-guide-rdo/environment-packages.html>

There are many additional steps here specific to XenServer.

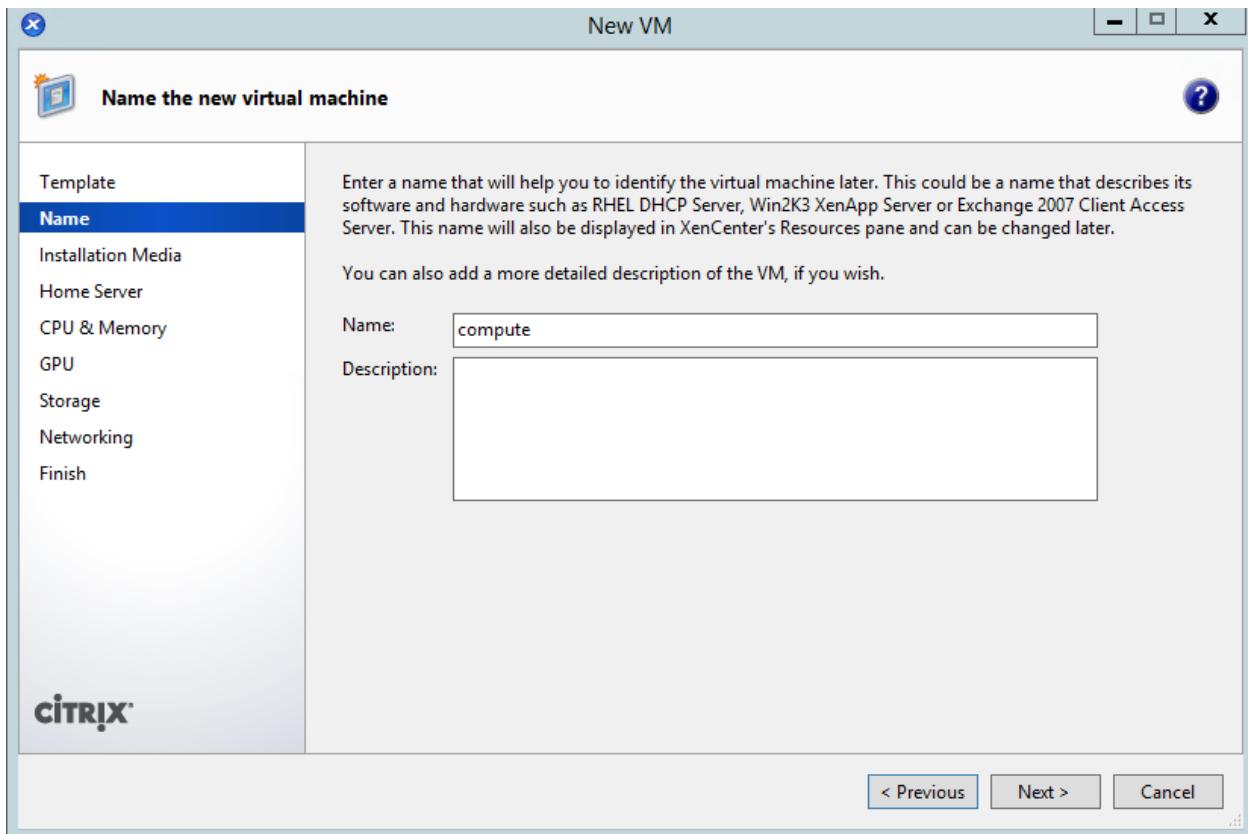
1. In XenCenter, create a new VM:



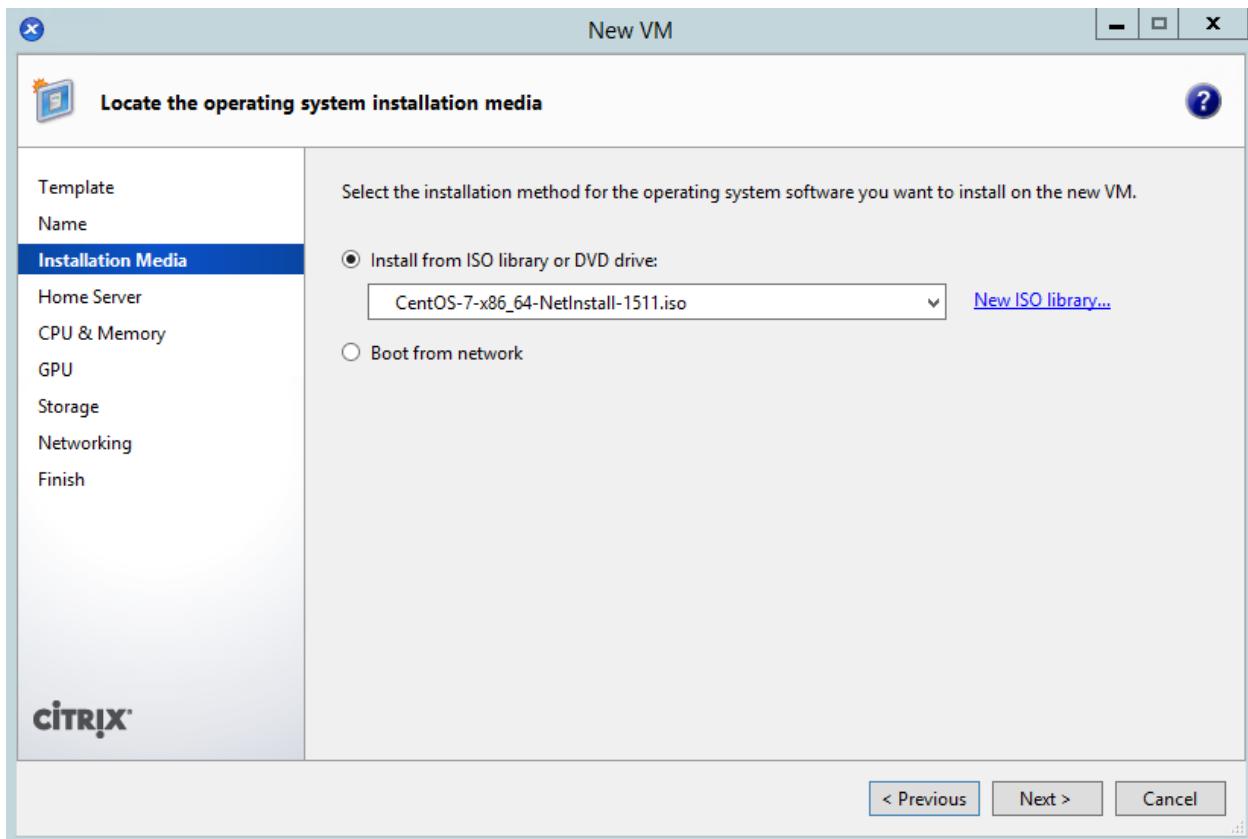
2. Select the CentOS 7 template:



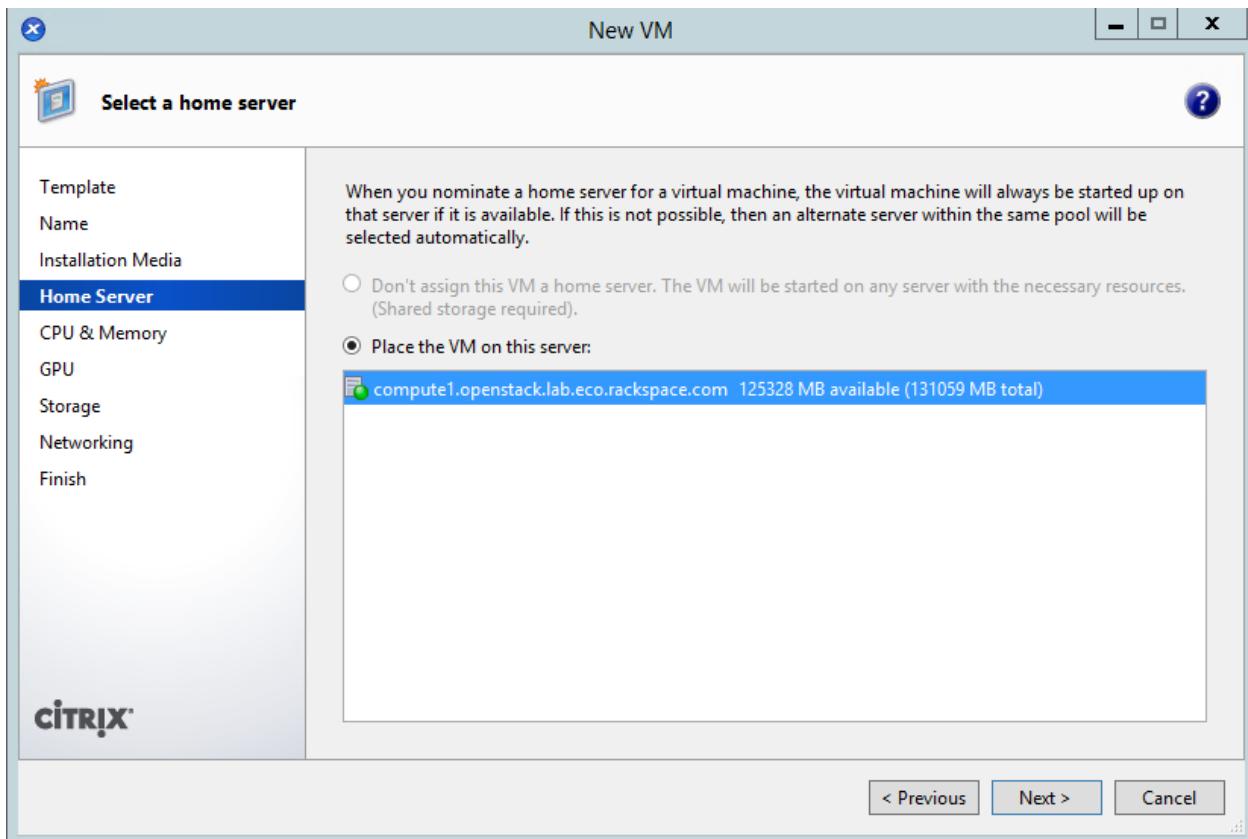
3. Name the VM "compute":



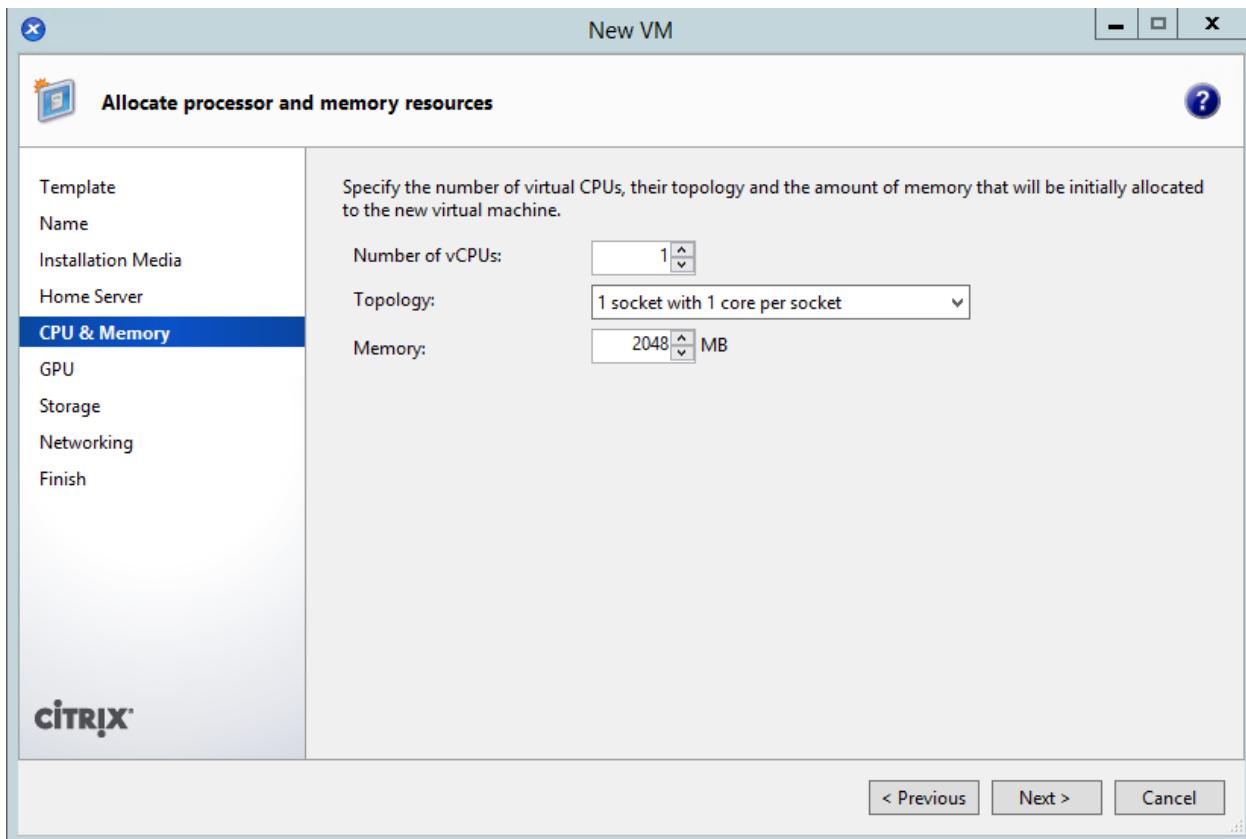
4. Choose the CentOS 7 ISO (which you should have previously uploaded to the ISO library):



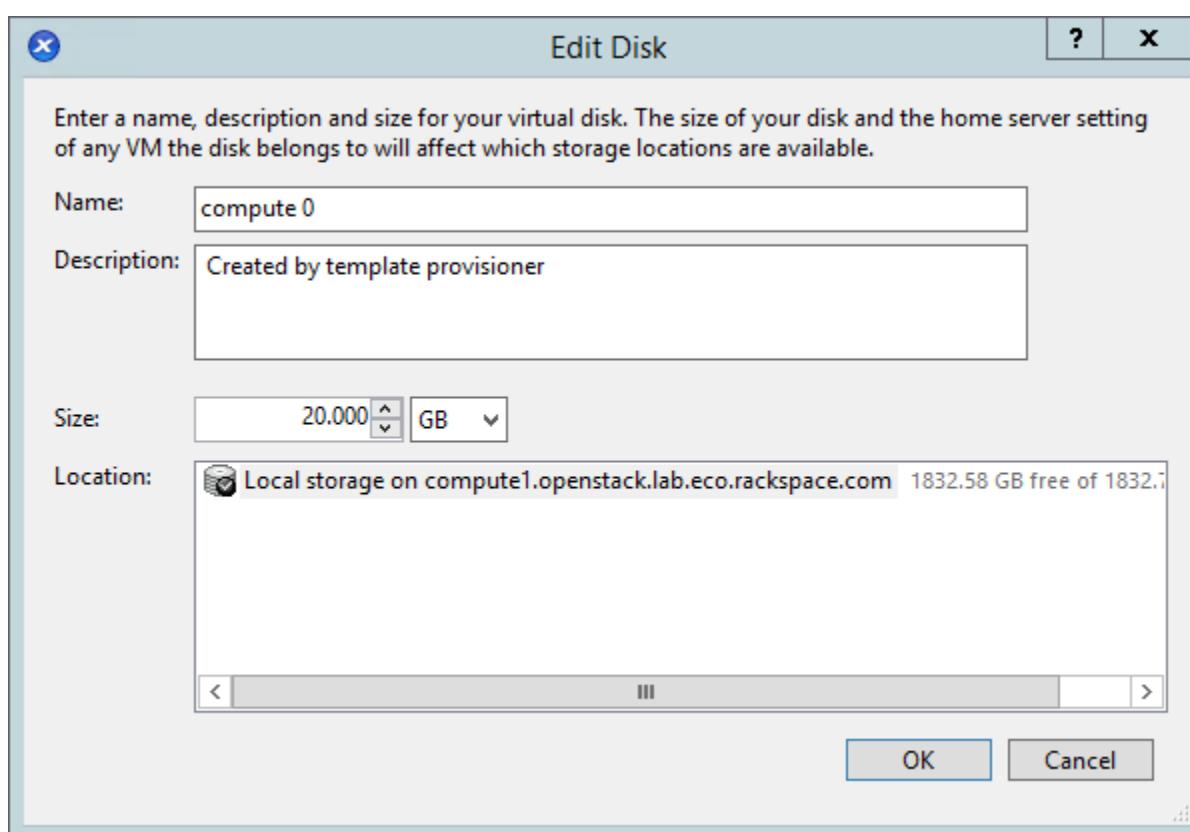
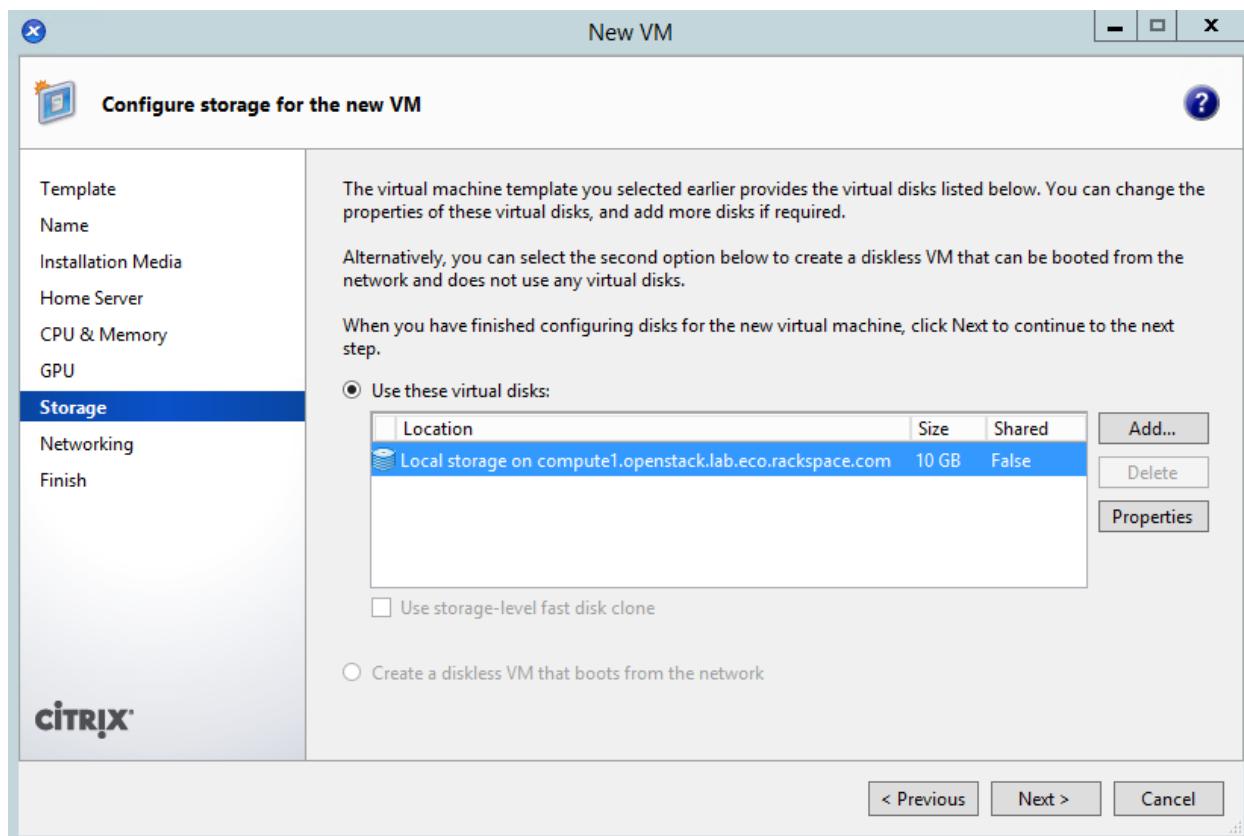
5. Place the VM on the only server available:



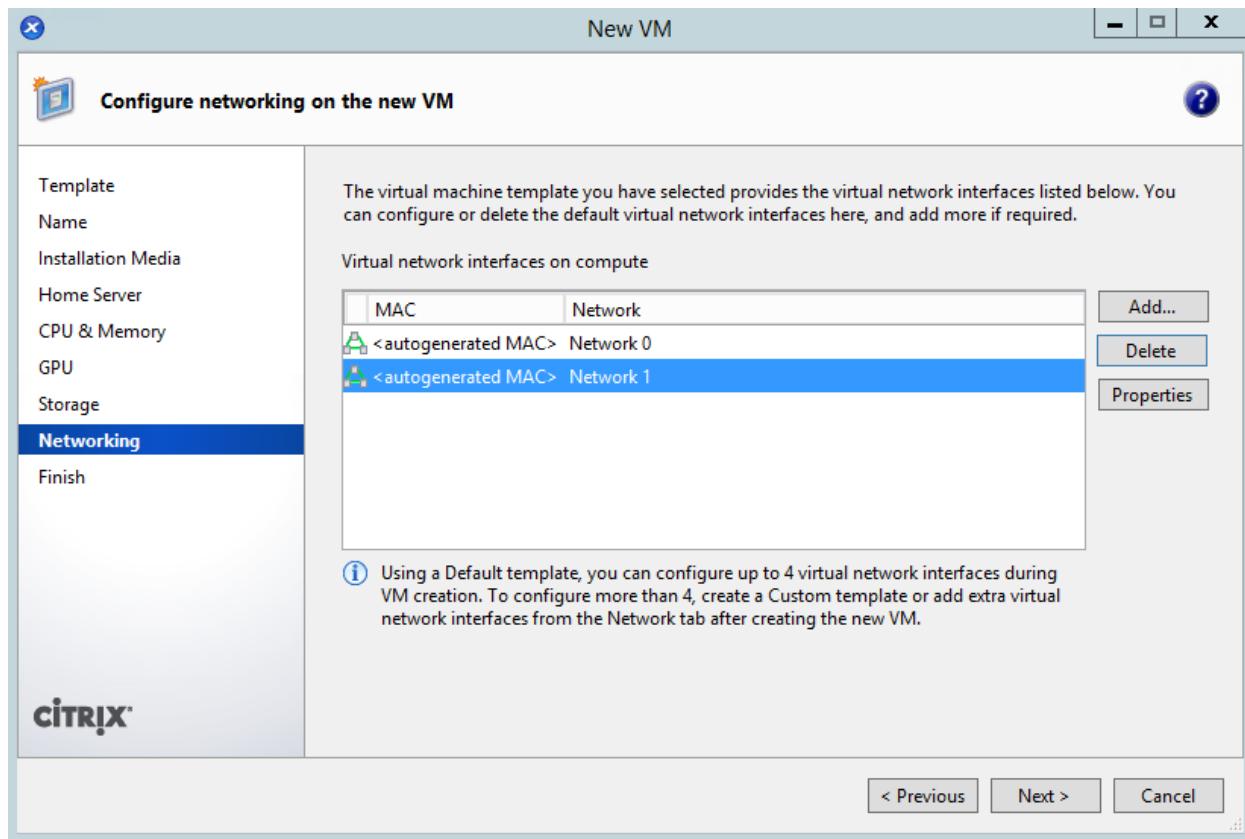
6. Give it one CPU and 2GB:



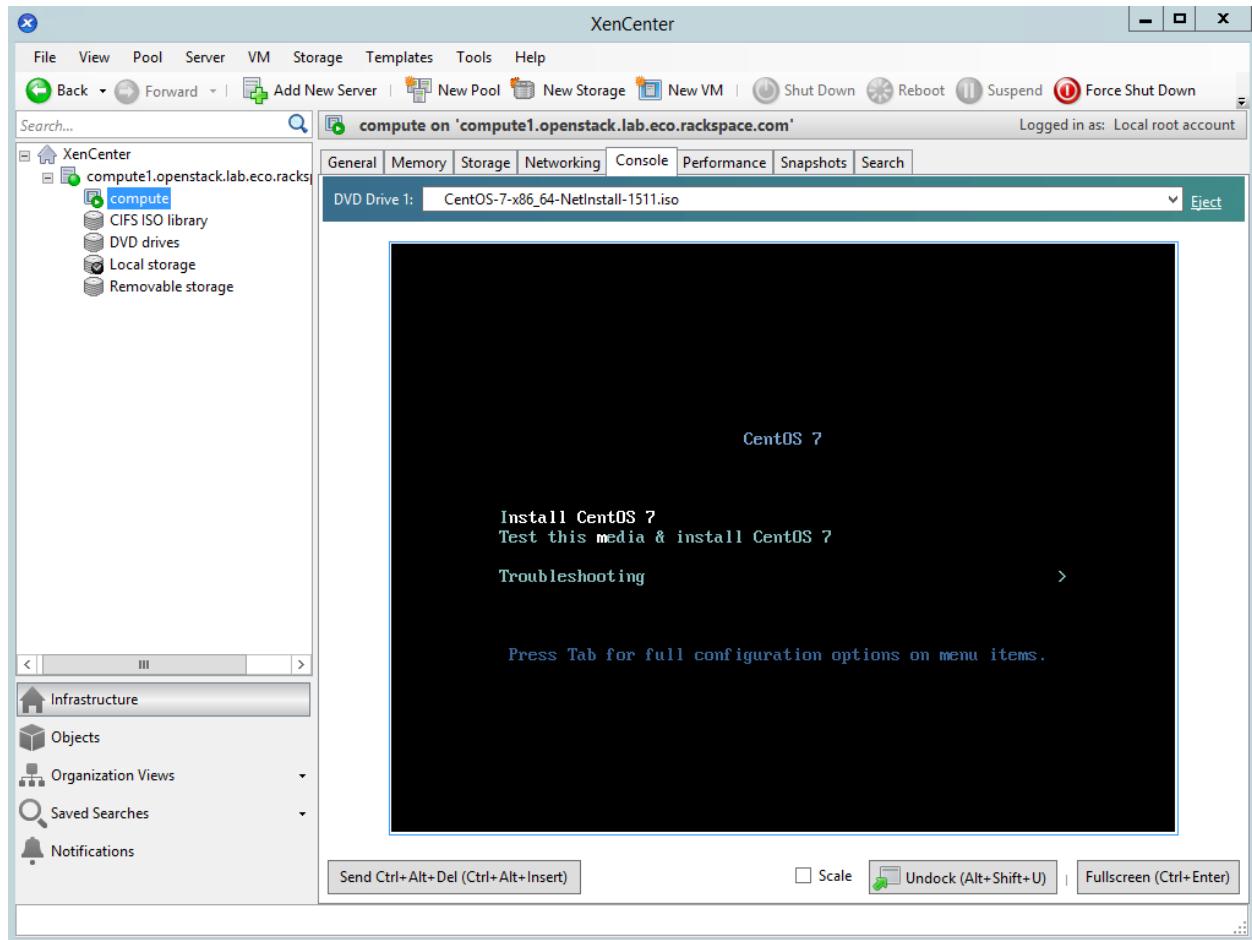
7. Change the disk to 20GB by clicking on properties:



- Give the VM connections to your management and public networks:



9. Complete the wizard, which will start the VM.
10. Go to the “compute” VM’s console, which should be displaying the CentOS installer’s boot screen:



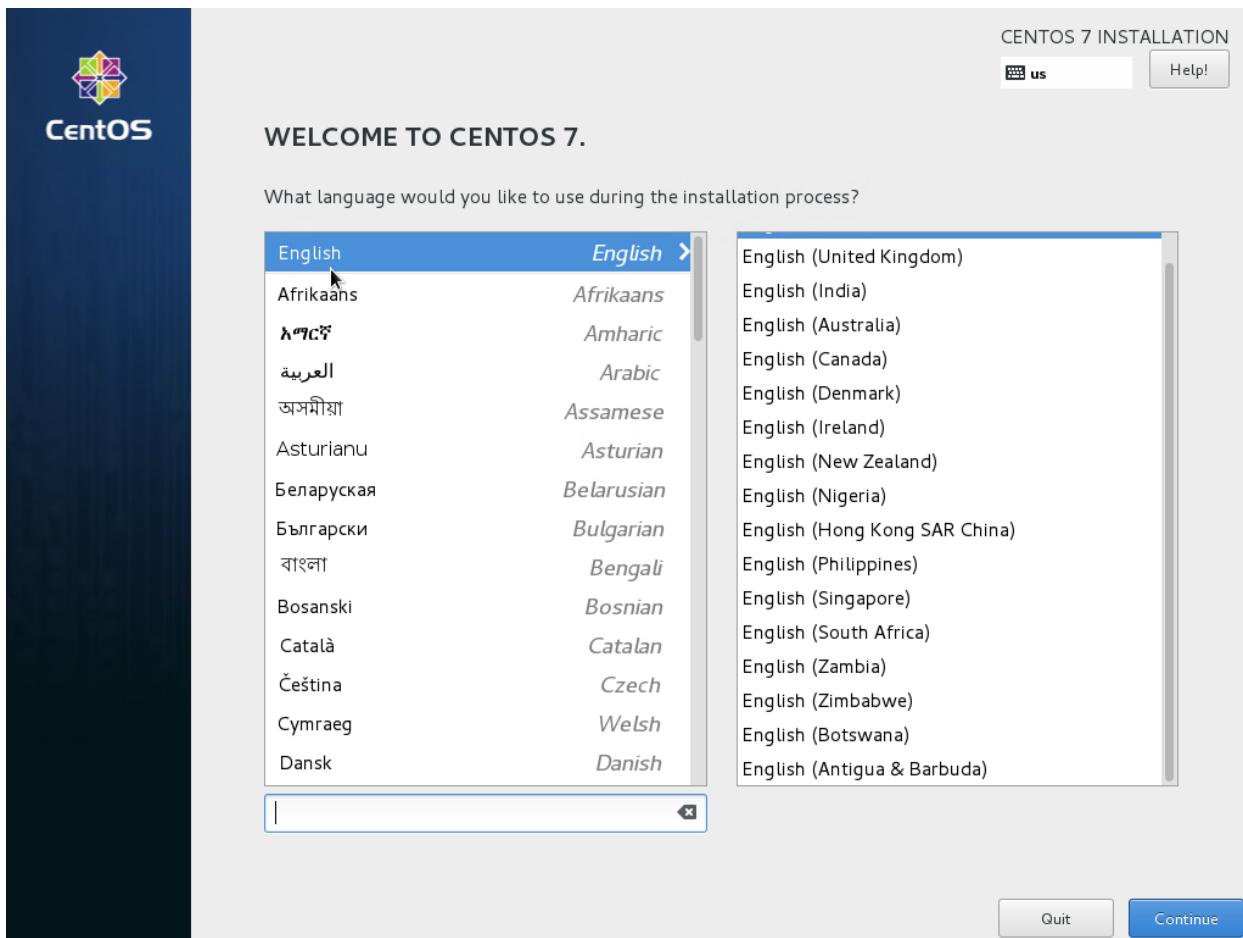
11. Highlight “Install CentOS 7”, and press Enter:

```
Stopping dracut initqueue hook...
Stopping Open-iSCSI...
[ OK ] Stopped udev Coldplug all Devices...
Stopping udev Coldplug all Devices...
[ OK ] Stopped dracut pre-trigger hook...
Stopping dracut pre-trigger hook...
[ OK ] Stopped udev Kernel Device Manager...
[ OK ] Stopped Open-iSCSI...
Stopping Device-Mapper Multipath Device Controller...
[ OK ] Stopped dracut pre-udev hook...
Stopping dracut pre-udev hook...
[ OK ] Stopped dracut cndline hook...
Stopping dracut cndline hook...
[ OK ] Stopped Create Static Device Nodes in /dev...
Stopping Create Static Device Nodes in /dev...
[ OK ] Stopped Create list of required static device nodes for the current kernel...
Stopping Create list of required static device nodes for the current kernel...
[ OK ] Closed udev Kernel Socket...
[ OK ] Closed udev Control Socket...
Starting Cleanup udevd DB...
[ OK ] Stopped Device-Mapper Multipath Device Controller...
[ OK ] Started Plymouth switch root service...
[ OK ] Started Cleanup udevd DB...
[ OK ] Reached target Switch Root...
Starting Switch Root...

Welcome to CentOS Linux 7 (Core)!

[ OK ] Stopped Switch Root.
[ OK ] Listening on LVM2 poll daemon socket.
[ OK ] Listening on LVM2 metadata daemon socket...
Mounting Temporary Directory...
Starting Create list of required static device nodes for the current kernel...
[ OK ] Listening on udev Kernel Socket...
Mounting Debug File System...
[ OK ] Stopped target Switch Root.
[ OK ] Stopped target Initrd File Systems...
Starting Device-Mapper Multipath Device Controller...
[ OK ] Reached target Paths.
[ OK ] Created slice User and Session Slice.
[ OK ] Stopped targetrunlevel file systems...
Starting Device-Mapper Multipath Device Controller...
[ OK ] Reached target Paths.
[ OK ] Created slice User and Session Slice.
[ OK ] Listening on udev Control Socket...
[ OK ] Created slice system-anaconda\x2dshe11.slice...
[ OK ] Reached target Swap...
Mounting POSIX Message Queue File System...
```

12. If the console appears to “hang”, with only a cursor showing (and no other activity), then quit XenCenter, relaunch it, and go back to the console. This should show the graphical installer is now running:



13. Set language and timezone.
14. Click on “Network & Hostname”. Click on the “eth1” interface, and click on “configure”.
15. Set the IPv4 address as appropriate:

Editing eth1

Connection name: **eth1**

General	Ethernet	802.1x Security	DCB	IPv4 Settings	IPv6 Settings
---------	----------	-----------------	-----	----------------------	---------------

Method: **Manual**

Addresses

Address	Netmask	Gateway	
172.16.0.192	255.255.255.0	172.16.0.1	Add
			Delete

DNS servers: **8.8.8.8**

Search domains:

DHCP client ID:

Require IPv4 addressing for this connection to complete

Routes...

Cancel **Save**

16. Disable IPv6, and click on “save”:

Editing eth1

Connection name: **eth1**

General	Ethernet	802.1x Security	DCB	IPv4 Settings	IPv6 Settings
---------	----------	-----------------	-----	----------------------	----------------------

Method: **Ignore**

Addresses

Address	Prefix	Gateway	
			Add
			Delete

DNS servers:

Search domains:

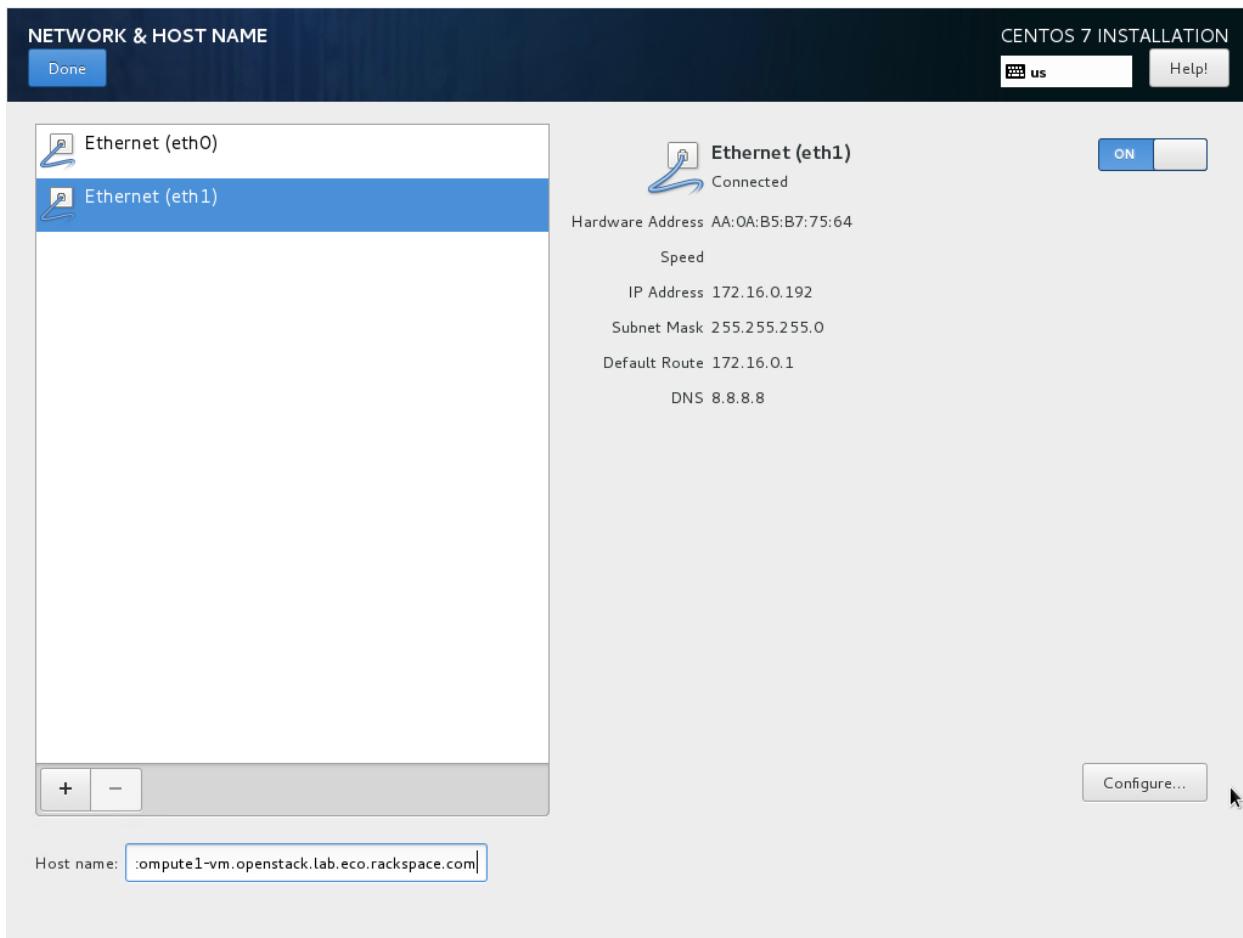
IPv6 privacy extensions: **Disabled**

Require IPv6 addressing for this connection to complete

Routes...

Cancel **Save**

17. Set an appropriate hostname, and then enable the “eth1” interface by setting the switch to “on”:



18. If using the NetInstall image, click on “Installation source”. Set the source to network, and then define a known-good mirror. You can use http://mirror.rackspace.com/CentOS/7.2.1511/os/x86_64/.
19. Click on “Installation Destination”. Select “I will configure partitioning” and click on “Done”:

INSTALLATION DESTINATION

CENTOS 7 INSTALLATION

Done

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

20 GiB

xvda / 20 GiB free

Disks left unselected here will not be touched.

Specialized & Network Disks

Add a disk...

Disks left unselected here will not be touched.

Other Storage Options

Partitioning

Automatically configure partitioning. I will configure partitioning.

I would like to make additional space available.

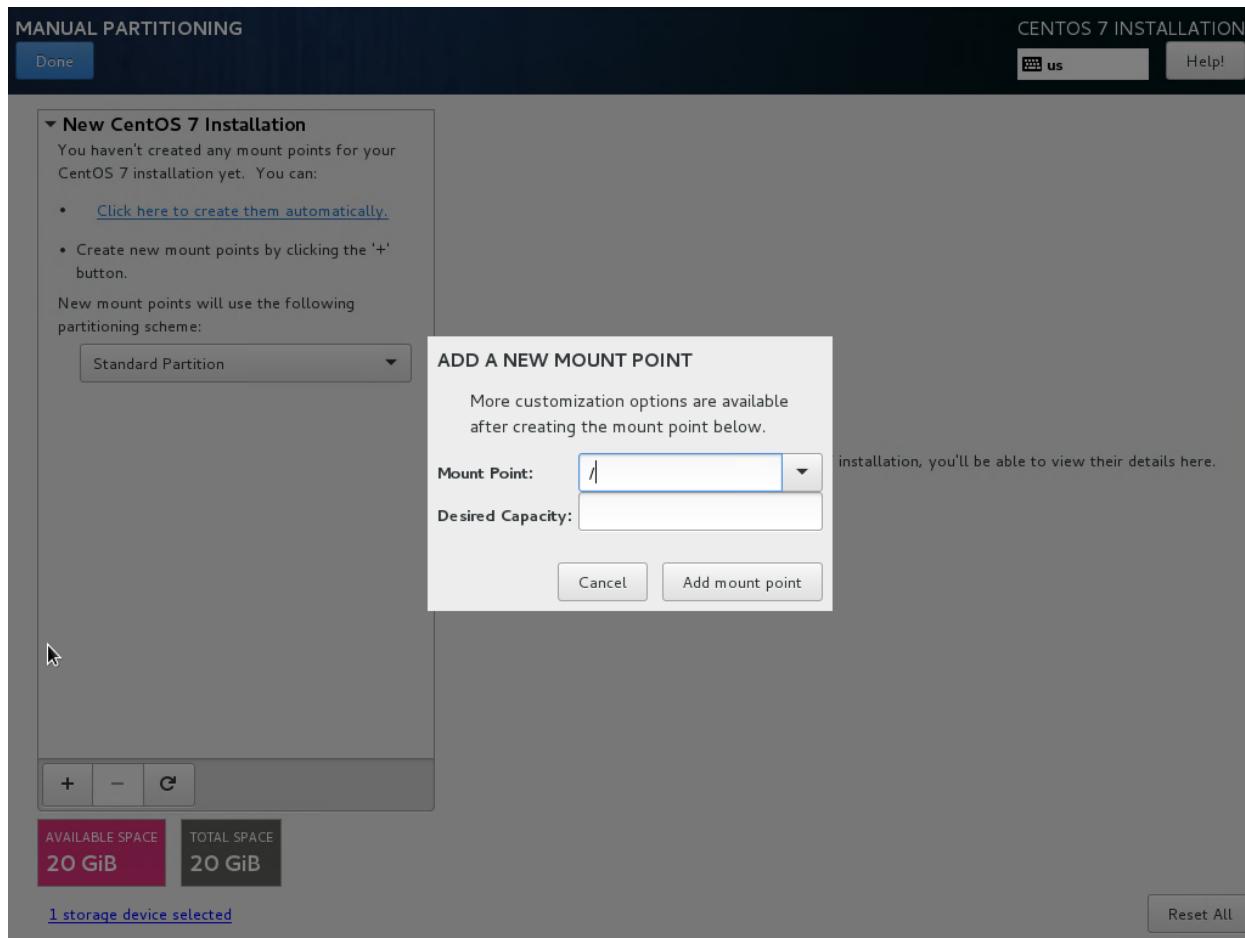
Encryption

Encrypt my data. You'll set a passphrase next.

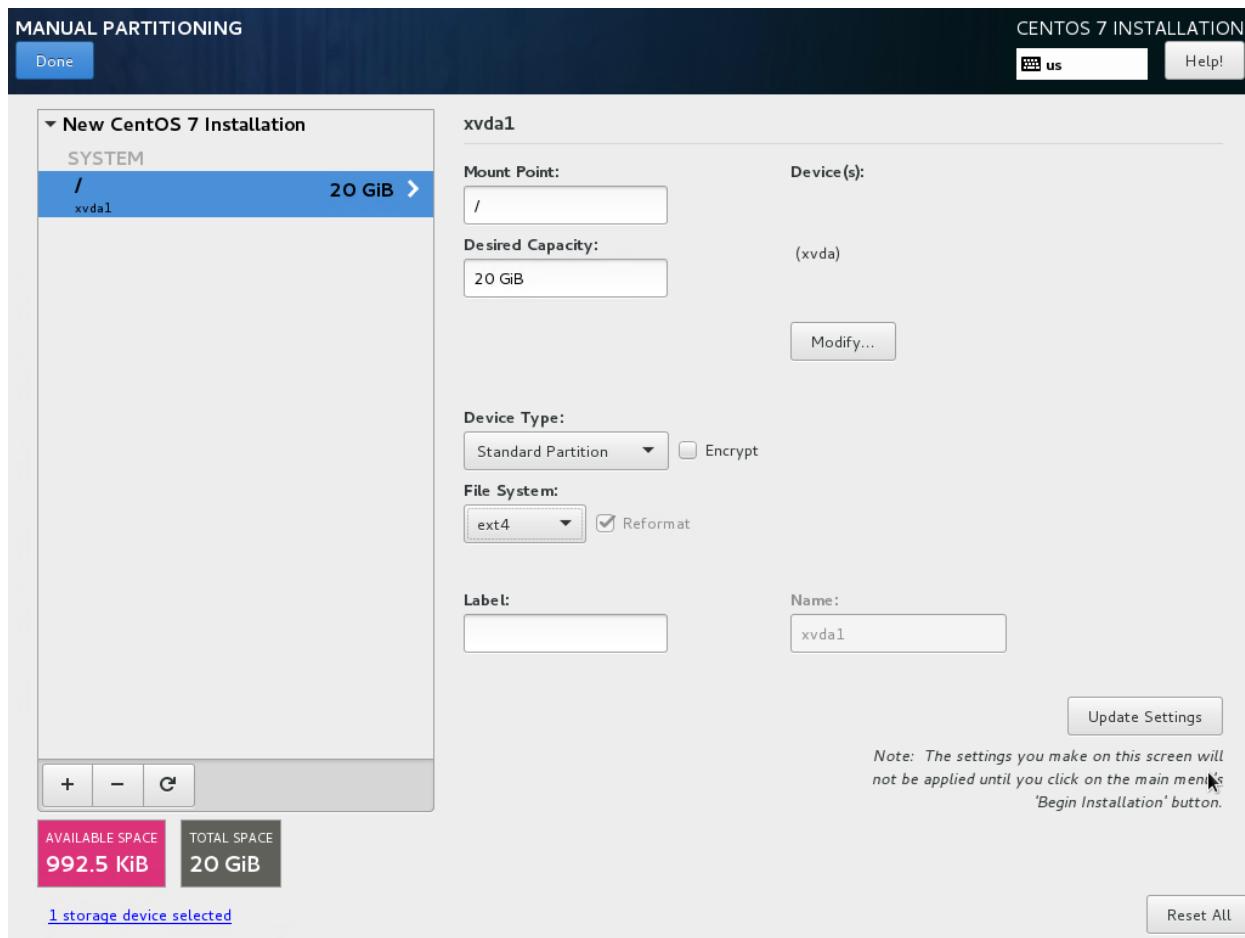
[Full disk summary and boot loader...](#)

1 disk selected; 20 GiB capacity; 20 GiB free

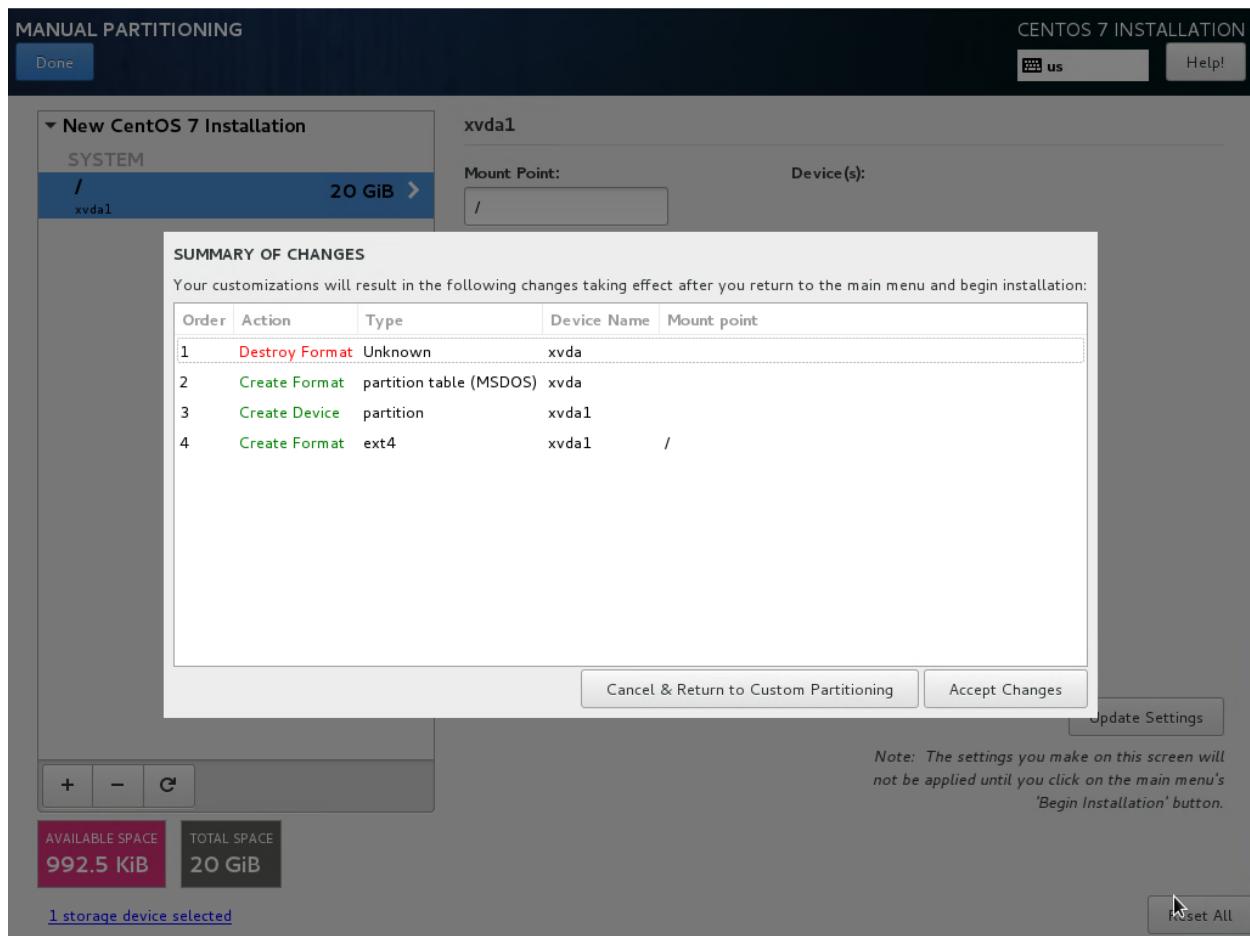
20. Under “New mount points will use the following partition scheme”, select “Standard Partition”.
21. Click on the + button. Set the mount point to / and click “Add mount point”:



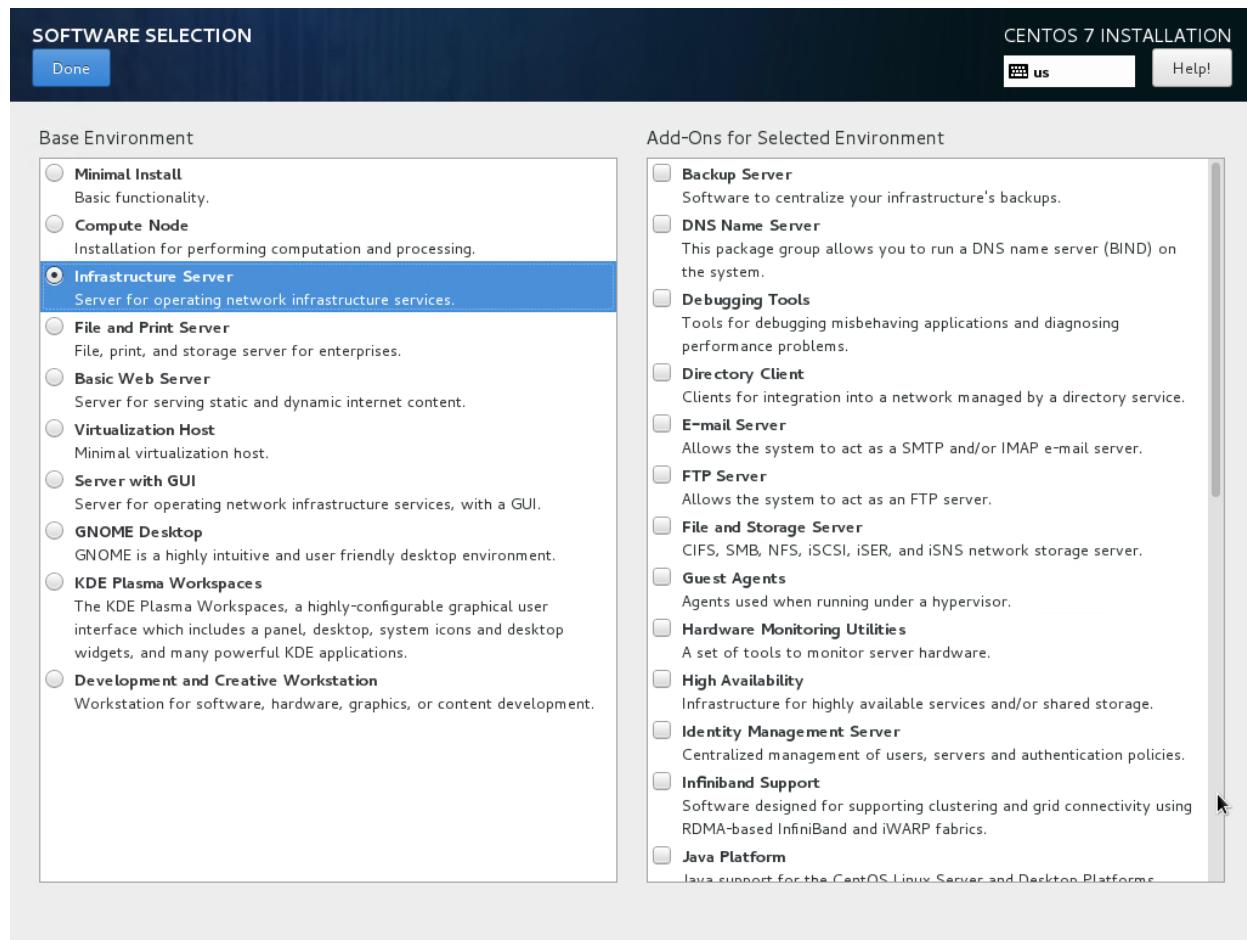
22. Set “File System” to “ext4”, and then click “Done”.



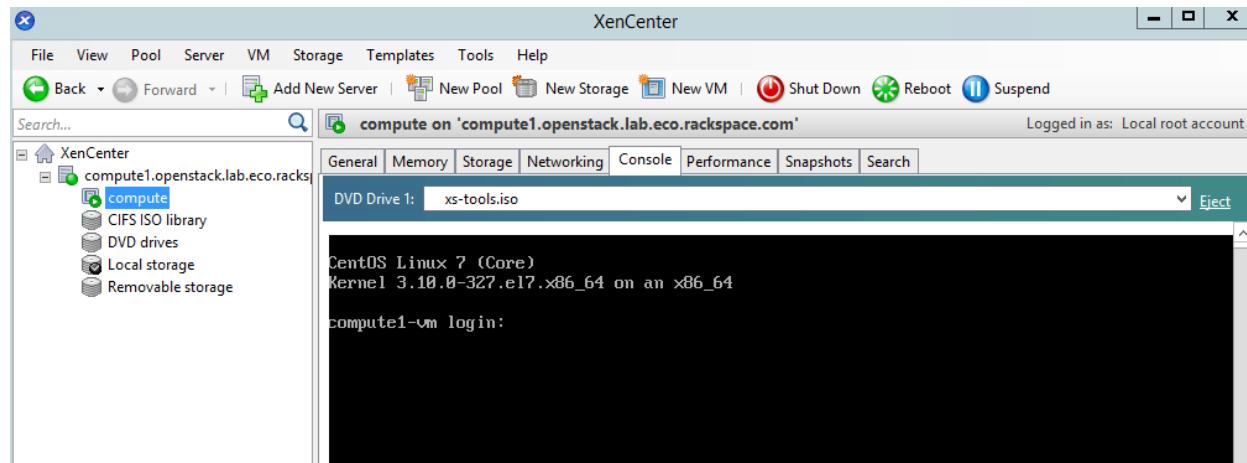
23. A yellow warning bar will appear. Click “Done” again, and then click on “Accept Changes”.



24. Click on “Software Selection”. Select “Infrastructure Server”, and click “Done”.



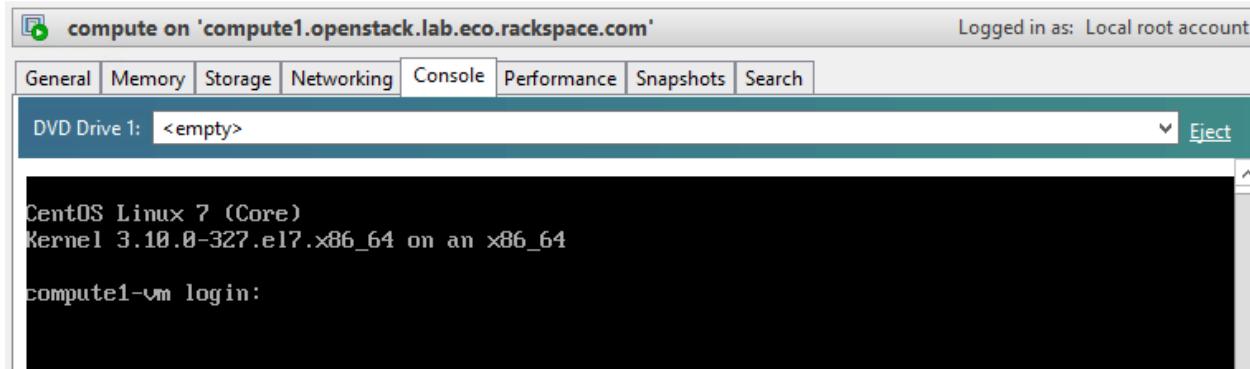
25. Click “Begin Installation”. Click on “Root Password” and set a good password.
26. Once installation is complete, click “Reboot”.
27. SSH as root to the new VM.
28. In XenCenter, change the DVD drive to xs-tools.iso:



29. Mount the tools ISO and install the tools:

```
# mkdir /mnt/cdrom
# mount /dev/cdrom /mnt/cdrom
# cd /mnt/cdrom/Linux
# rpm -Uvh xe-guest-utilities-6.5.0-1427.x86_64.rpm xe-guest-utilities-xenstore-6.
# rpm -Uvh xe-guest-utilities-6.5.0-1427.x86_64.rpm
# cd ~
# umount /mnt/cdrom
```

30. In XenCenter, eject the DVD drive:



31. Stop and disable the firewalld service:

```
# systemctl disable firewalld.service
# systemctl stop firewalld.service
```

32. Disable SELINUX:

```
# setenforce 0
# vim /etc/sysconfig/selinux

SELINUX=permissive
```

33. Update all packages on the VM:

```
# yum update
```

34. Reboot the VM:

```
# systemctl reboot
```

35. Wait for the VM to complete the reboot, and SSH back in as root.

36. Update the system hosts file with entries for all nodes:

```
# vim /etc/hosts

172.16.0.192 controller controller.openstack.lab.eco.rackspace.com
172.16.0.203 compute1 compute1.openstack.lab.eco.rackspace.com
172.16.0.204 compute1-vm compute1-vm.openstack.lab.eco.rackspace.com
172.16.0.195 compute2 compute2.openstack.lab.eco.rackspace.com
172.16.0.196 block1 block1.openstack.lab.eco.rackspace.com
172.16.0.197 object1 object1.openstack.lab.eco.rackspace.com
172.16.0.198 object2 object2.openstack.lab.eco.rackspace.com
```

37. Update the chrony configuration to use the controller as a time source:

```
# vim /etc/chrony.conf  
  
server controller iburst
```

- Remove any other servers listed, leaving only “controller”.

38. Restart the chrony service, and confirm that “controller” is listed as a source:

```
# systemctl restart chronyd.service  
# chronyc sources  
 210 Number of sources = 1  
    MS Name/IP address          Stratum Poll Reach LastRx Last sample  
=====  
    ^* controller                3      6     17      6 -3374ns [+2000ns] +/- 6895us
```

39. Enable the OpenStack-Liberty yum repository:

```
# yum install centos-release-openstack-liberty
```

40. Install the OpenStack client and SELINUX support:

```
# yum install python-openstackclient openstack-selinux
```

41. SSH to the XenServer as root.

42. Obtain the UUID of the XenServer pool:

```
# xe pool-list  
  
uuid ( RO) : f824b628-1696-9ebe-5a5a-d1f9cf117158  
  name-label ( RW):  
  name-description ( RW):  
    master ( RO): b11f5aab-d1a5-42fb-8335-3a6451cec4c7  
    default-SR ( RW): 271e0f43-8b03-50c5-a08a-9c7312741378
```

- Note: In my case, the UUID is f824b628-1696-9ebe-5a5a-d1f9cf117158.

43. Enable auto power-on for the XenServer pool. Replace *POOL_UUID* with your own:

```
# xe pool-param-set uuid=*POOL_UUID* other-config:auto_poweron=true
```

44. Obtain the UUID of the “compute VM”:

```
# xe vm-list name-label='compute'  
  
uuid ( RO) : 706ba8eb-fe5f-8da2-9090-3a5b009ce1c4  
  name-label ( RW): compute  
  power-state ( RO): running
```

- Note: In my case, the UUID is 706ba8eb-fe5f-8da2-9090-3a5b009ce1c4.

45. Enable auto power-on for the “compute” VM. Replace *VM_UUID* with your own:

```
# xe vm-param-set uuid=*VM_UUID* other-config:auto_poweron=true
```

CHAPTER 9

9. Install Compute (nova) on XenServer compute VM

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/nova-compute-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/nova-verify.html>

<http://docs.openstack.org/liberty/install-guide-rdo/cinder-storage-install.html>

It is also based on some steps from the following guide:

<https://www.citrix.com/blogs/2015/11/30/integrating-xenserver-rdo-and-neutron/>

All steps have modifications for XenServer.

1. Download and install pip, and xenapi:

```
# wget https://bootstrap.pypa.io/get-pip.py
# python get-pip.py
# pip install xenapi
```

2. Install nova packages:

```
# yum install openstack-nova-compute sysfsutils
```

3. Configure nova. Replace *HOST_NAME*, *XENSERVER_ROOT*, *CONTROLLER_ADDRESS*, *XAPI_BRIDGE*, *VM_IP*, *NOVA_PASS*, *XENSERVER_IP* and *RABIT_PASS* with your own:

```
# vim /etc/nova/nova.conf

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
my_ip = *VM_IP*
network_api_class = nova.network.neutronv2.api.API
security_group_api = neutron
linuxnet_interface_driver = nova.network.linux_net.
˓→NeutronLinuxBridgeInterfaceDriver
```

```
firewall_driver = nova.virt.firewall.NoopFirewallDriver
compute_driver = xenapi.XenAPIDriver
host = *HOST_NAME*
live_migration_retry_count=600

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = nova
password = *NOVA_PASS*

[vnc]
enabled = True
vncserver_listen = 0.0.0.0
vncserver_proxyclient_address = *XENSERVER_IP*
novncproxy_base_url = http://*CONTROLLER_ADDRESS*:6080/vnc_auto.html

[glance]
host = controller

[oslo_concurrency]
lock_path = /var/lib/nova/tmp

[xenserver]
connection_url=http://compute1
connection_username=root
connection_password=*XENSERVER_ROOT*
vif_driver=nova.virt.xenapi.vif.XenAPIOpenVswitchDriver
ovs_int_bridge=*XAPI_BRIDGE*
ovs_integration_bridge=*XAPI_BRIDGE*
```

4. Download and modify a helper script for installing the dom0 plugins:

```
# wget --no-check-certificate https://raw.githubusercontent.com/Annie-XIE/summary-
→os/master/rdo_xenserver_helper.sh
# sed -i 's/dom0_ip=169.254.0.1/dom0_ip=compute1/g' rdo_xenserver_helper.sh
```

5. Use the script to install the dom0 nova plugins:

```
# source rdo_xenserver_helper.sh
# install_dom0_plugins
```

- Answer yes to the RSA key prompt
- Enter the XenServer root password when prompted (twice)
- Ignore the errors related to the neutron plugins

6. Update the LVM configuration to prevent scanning of instances' contents:

```
# vim /etc/lvm/lvm.conf

devices {
    ...
    filter = ["r/.*/"]
```

- Note: Do not replace the entire “devices” section, only the “filter” line.

7. Enable and start the nova services:

```
# systemctl enable openstack-nova-compute.service
# systemctl start openstack-nova-compute.service
```

8. Log on to the controller node as root.

9. Load the “admin” credential file:

```
# source admin-openrc.sh
```

10. Check the nova service list:

```
# nova service-list

+-----+-----+-----+-----+
| Id | Binary          | Host                               | Zone |
+-----+-----+-----+-----+
| 1  | nova-consoleauth | controller.openstack.lab.eco.rackspace.com | -   |
| 2  | nova-scheduler   | controller.openstack.lab.eco.rackspace.com | -   |
| 3  | nova-conductor   | controller.openstack.lab.eco.rackspace.com | -   |
| 4  | nova-cert         | controller.openstack.lab.eco.rackspace.com | -   |
| 5  | nova-compute      | compute1-vm.openstack.lab.eco.rackspace.com | nova |
+-----+-----+-----+-----+
```

- The list should include compute1-vm running nova-compute.

11. Check the nova endpoints list:

```
# nova endpoints

WARNING: nova has no endpoint in ! Available endpoints for this service:
+-----+-----+
| nova    | Value           |
+-----+-----+
| id      | 1c07bba299254336abd0cbe27c64be83 |
| interface | internal |
| region   | RegionOne |
| region_id | RegionOne |
| url     | http://controller:8774/v2/76f8c8fd7b1e407d97c4604eb2a408b3 |
+-----+-----+
+-----+-----+
```

```
| nova      | Value
+-----+
| id       | 221f3238f2da46fb8fc6897e6c2c4de1
| interface | public
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:8774/v2/76f8c8fd7b1e407d97c4604eb2a408b3
+-----+
+-----+
| nova      | Value
+-----+
| id       | fdbd2fe1dda5460aaa486b5d142f99aa
| interface | admin
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:8774/v2/76f8c8fd7b1e407d97c4604eb2a408b3
+-----+
+-----+
WARNING: keystone has no endpoint in ! Available endpoints for this service:
+-----+
| keystone | Value
+-----+
| id       | 33c74602793e454ea1d9ae9ab6ca5dcc
| interface | public
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:5000/v2.0
+-----+
+-----+
| keystone | Value
+-----+
| id       | 688939b258ea4f1d956cb85dfc75e0c0
| interface | internal
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:5000/v2.0
+-----+
+-----+
| keystone | Value
+-----+
| id       | 7c7652f07b2f4a2c8bf805ff49b6a4eb
| interface | admin
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:35357/v2.0
+-----+
+-----+
WARNING: glance has no endpoint in ! Available endpoints for this service:
+-----+
| glance   | Value
+-----+
| id       | 0d49d35fc21d4faa8c72ff3578198513
| interface | internal
| region   | RegionOne
| region_id | RegionOne
| url      | http://controller:9292
+-----+
+-----+
| glance   | Value
+-----+
```

id	54f519365b8e4f7f81b750fdbf55be2f
interface	public
region	RegionOne
region_id	RegionOne
url	http://controller:9292
<hr/>	
<hr/>	
glance	Value
<hr/>	
id	d5e7d60a0eba46b9ac7b992214809fe0
interface	admin
region	RegionOne
region_id	RegionOne
url	http://controller:9292
<hr/>	

- The list should include endpoints for nova, keystone, and glance. Ignore any warnings.

12. Check the nova image list:

nova image-list
<hr/>
ID
1e710e0c-0fb6-4425-b196-4b66bfac495e cirros-xen
<hr/>
<hr/>

- The list should include the cirros-xen image previously uploaded.

CHAPTER 10

10. Install Networking (neutron) on controller

This page is based on the following OpenStack Installation Guide page:

<http://docs.openstack.org/liberty/install-guide-rdo/neutron-controller-install.html>

Steps 3, 5, 6, 7, 9, 12, 13 and 15 have specific changes for the use of XenServer.

1. Open the MySQL client and create the “glance” database. Replace *NEUTRON_DBPASS* with your own:

```
# mysql
> create database neutron;
> grant all privileges on neutron.* to 'neutron'@'localhost' identified by
  ↵'*NEUTRON_DBPASS*';
> grant all privileges on neutron.* to 'neutron'@'%' identified by '*NEUTRON_
  ↵DBPASS*';
> quit
```

2. Create the “neutron” user, role, service and endpoints. Provide *NEUTRON_PASS* when prompted:

```
# source admin-openrc.sh
# openstack user create --domain default --password-prompt neutron
# openstack role add --project service --user neutron admin
# openstack service create --name neutron --description "OpenStack Networking" ↵
  ↵network
# openstack endpoint create --region RegionOne network public http://
  ↵controller:9696
# openstack endpoint create --region RegionOne network internal http://
  ↵controller:9696
# openstack endpoint create --region RegionOne network admin http://
  ↵controller:9696
```

3. **Install the neutron and ovs packages:**

```
# yum install openstack-neutron openstack-neutron-ml2 openstack-neutron-
  ↵openvswitch python-neutronclient ebtables ipset
```

4. Configure neutron. Note that the default file already has lines for keystone_auth_token. These must be deleted. Replace *NEUTRON_DBPASS*, *NEUTRON_PASS*, *RABBIT_PASS* and *NOVA_PASS* with your own:

```
# vim /etc/neutron/neutron.conf

[database]
connection = mysql://neutron:*NEUTRON_DBPASS*@controller/neutron
rpc_backend = rabbit

[DEFAULT]
core_plugin = ml2
service_plugins =
auth_strategy = keystone
notify_nova_on_port_status_changes = True
notify_nova_on_port_data_changes = True
nova_url = http://controller:8774/v2

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = neutron
password = *NEUTRON_PASS*

[nova]
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
region_name = RegionOne
project_name = service
username = nova
password = *NOVA_PASS*

[oslo_concurrency]
lock_path = /var/lib/neutron/tmp
```

- Note: The service_plugins value is intentionally left blank, and is used to disable these plugins.

5. Configure the ml2 plugin:

```
# vim /etc/neutron/plugins/ml2/ml2_conf.ini

[ml2]
type_drivers = flat,vlan
tenant_network_types =
mechanism_drivers = openvswitch
extension_drivers = port_security

[ml2_type_flat]
```

```
flat_networks = public

[securitygroup]
enable_ipset = True
```

- Note: The tenant_network_types value is also intentionally left blank.

6. Configure ml2's ovs plugin. Replace *XAPI_BRIDGE* with your own:

```
# vim /etc/neutron/plugins/ml2/openvswitch_agent.ini

[ovs]
integration_bridge = *XAPI_BRIDGE*
bridge_mappings = public:br-eth0

[securitygroup]
firewall_driver = neutron.agent.firewall.NoopFirewallDriver
```

7. Configure the DHCP Agent. Replace *XAPI_BRIDGE* with your own:

```
# vim /etc/neutron/dhcp_agent.ini

[DEFAULT]
interface_driver = neutron.agent.linux.interface.OVSIInterfaceDriver
ovs_integration_bridge = *XAPI_BRIDGE*
dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq
enable_isolated_metadata= True
```

8. Configure the metadata agent. Note that the default file already has some lines in [DEFAULT]. These need to be commented-out or deleted. Replace *NEUTRON_PASS* and *NEUTRON_METADATA_SECRET* with your own:

```
# vim /etc/neutron/metadata_agent.ini

[DEFAULT]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_region = RegionOne
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = neutron
password = *NEUTRON_PASS*
nova_metadata_ip = controller
metadata_proxy_shared_secret = *NEUTRON_METADATA_SECRET*
```

9. Reconfigure nova to use neutron. Replace *NEUTRON_PASS* , *NEUTRON_METADATA_SECRET* and *XAPI_BRIDGE* with your own:

```
# vim /etc/nova/nova.conf

[neutron]
url = http://controller:9696
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
```

```
region_name = RegionOne
project_name = service
username = neutron
password = *NEUTRON_PASS*
service_metadata_proxy = True
metadata_proxy_shared_secret = *NEUTRON_METADATA_SECRET*
ovs_bridge = *XAPI_BRIDGE*
```

10. Symlink the ml2 configuration file to neutron's plugin.ini file:

```
# ln -s /etc/neutron/plugins/ml2/ml2_conf.ini /etc/neutron/plugin.ini
```

11. Populate the neutron database:

```
# su -s /bin/sh -c "neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_conf.ini upgrade head" neutron
```

12. **Enable and start the ovs service:**

```
# systemctl enable openvswitch.service
# systemctl start openvswitch.service
```

13. **Set up the ovs bridge to the public network:**

```
# ovs-vsctl add-br br-eth0
# ovs-vsctl add-port br-eth0 eth0
```

14. Restart the nova service:

```
# systemctl restart openstack-nova-api.service
```

15. **Enable and start the neutron services:**

```
# systemctl enable neutron-server.service neutron-openvswitch-agent.service
# systemctl start neutron-server.service neutron-openvswitch-agent.service
```

CHAPTER 11

11. Install Networking (neutron) on compute VM

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/neutron-compute-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/launch-instance.html>

<http://docs.openstack.org/liberty/install-guide-rdo/launch-instance-networks-public.html>

It is also based on some steps from the following guide:

<https://www.citrix.com/blogs/2015/11/30/integrating-xenserver-rdo-and-neutron/>

Steps 1, 3, 4, 6, 8, 11, 14 and 15 have specific changes for the use of XenServer.

1. **Install the neutron and ovs packages:**

```
# yum install openstack-neutron openstack-neutron-openvswitch ebtables ipset
˓→openvswitch
```

2. **Configure neutron.** Replace *HOST_NAME*, *RABBIT_PASS* and *NEUTRON_PASS* with your own:

```
# vim /etc/neutron/neutron.conf

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
host = *HOST_NAME*

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
```

```
project_domain_id = default
user_domain_id = default
project_name = service
username = neutron
password = *NEUTRON_PASS*

[oslo_concurrency]
lock_path = /var/lib/neutron/tmp
```

- Make sure that any connection options under [database] are deleted or commented-out.
- Delete or comment-out any pre-existing lines in the [keystone_auth_token] section.

3. Configure the neutron ovs agent. Replace *XAPI_BRIDGE* and *XENSERVER_ROOT* with your own:

```
# vim /etc/neutron/plugins/ml2/openvswitch_agent.ini

[ovs]
integration_bridge = *XAPI_BRIDGE*
bridge_mappings = public:xenbr0

[agent]
root_helper = neutron-rootwrap-xen-dom0 /etc/neutron/rootwrap.conf
root_helper_daemon =
minimize_polling = False

[securitygroup]
firewall_driver = neutron.agent.firewall.NoopFirewallDriver
```

4. Configure neutron rootwrap to connect to XenServer. Replace *XENSERVER_ROOT* with your own:

```
# vim /etc/neutron/rootwrap.conf

[xenapi]
xenapi_connection_url=http://compute1
xenapi_connection_username=root
xenapi_connection_password=*XENSERVER_ROOT*
```

- There are other lines already present in this file. These should be left as-is.

5. Reconfigure nova to use neutron. Replace *NEUTRON_PASS* with your own:

```
# vim /etc/nova/nova.conf

[neutron]
url = http://controller:9696
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
region_name = RegionOne
project_name = service
username = neutron
password = *NEUTRON_PASS*
```

6. Use the helper script to install the dom0 neutron plugins:

```
# source rdo_xenserver_helper.sh
# install_dom0_plugins
```

- Enter the XenServer root password when prompted (twice).
 - If you are prompted whether or not to overwrite a file under /tmp, answer y.
7. Restart the nova service:

```
# systemctl restart openstack-nova-compute.service
```

8. Enable and start the neutron service:

```
# systemctl enable neutron-openvswitch-agent.service
# systemctl start neutron-openvswitch-agent.service
```

9. Log on to the controller node as root.

10. Load the “admin” credential file:

```
# source admin-openrc.sh
```

11. Check the neutron agent list:

```
# neutron agent-list

+-----+-----+-----+
| id | agent_type | host |
+-----+-----+-----+
| 57c49643-3e48-4252-9665-2f22e3b93b0e | Open vSwitch agent | compute1-vm.
| openstack.lab.eco.rackspace.com | :-) | True | neutron-openvswitch-
| agent |
| 977ff9ae-96e5-4ef9-93d5-65a8541d7d25 | Metadata agent | controller.
| openstack.lab.eco.rackspace.com | :-) | True | neutron-metadata-
| agent |
| ca0fb18a-b3aa-4cd1-bc5f-ba4700b4d9ce | Open vSwitch agent | controller.
| openstack.lab.eco.rackspace.com | :-) | True | neutron-openvswitch-
| agent |
| d42db23f-3738-48b3-8f83-279ee29e84ef | DHCP agent | controller.
| openstack.lab.eco.rackspace.com | :-) | True | neutron-dhcp-agent
| |
+-----+-----+-----+
```

- The list should include the ovs agent running on controller and compute1-vm.

12. Create the default security group:

```
# nova secgroup-add-rule default icmp -1 -1 0.0.0.0/0
# nova secgroup-add-rule default tcp 1 65535 0.0.0.0/0
```

13. Create the public network. Replace *PUBLIC_NETWORK_CIDR*, *START_IP_ADDRESS*, *END_IP_ADDRESS* *DNS_RESOLVER* and *PUBLIC_NETWORK_GATEWAY* with your own:

```
# neutron net-create public --shared --provider:physical_network public --
| provider:network_type flat
# neutron subnet-create public *PUBLIC_NETWORK_CIDR* --name public --allocation-
| pool start=*START_IP_ADDRESS*,end=*END_IP_ADDRESS* --dns-nameserver *DNS_
| RESOLVER* --gateway *PUBLIC_NETWORK_GATEWAY*
```

14. There is a bug regarding the network's segmentation ID which needs to be fixed. This should be resolved in openstack-neutron-7.0.1, but if you are running an older version:

- (a) Update the *segmentation_id* field in the *neutron* database:

```
# mysql neutron
> update ml2_network_segments set segmentation_id=0;
> quit
```

- (b) Update the *segmentation_id* for the DHCP agent's ovs port:

```
# ovs-vsctl set Port $(ovs-vsctl show | grep Port | grep tap | awk -F \" ' {_
˓print $2 } ') other_config:segmentation_id=0
```

15. There is a bug in Neutron which is causing available XenAPI sessions to be exhausted. I have reported this and submitted a patch in <https://bugs.launchpad.net/neutron/+bug/1558721>. Until the bug is fixed upstream, here is the manual patch to fix the problem:

1. Open the *neutron-rootwrap-xen-dom0* file:

```
# vim /usr/bin/neutron-rootwrap-xen-dom0
```

2. Locate the following lines (should start at line 117):

```
result = session.xenapi.host.call_plugin(
    host, 'netwrap', 'run_command',
    {'cmd': json.dumps(user_args), 'cmd_input': json.dumps(cmd_input)})
return json.loads(result)
```

3. Add the following before the 'return' line. It should have the same indentation as the 'return' line:

```
session.xenapi.session.logout()
```

4. The whole section should now read:

```
result = session.xenapi.host.call_plugin(
    host, 'netwrap', 'run_command',
    {'cmd': json.dumps(user_args), 'cmd_input': json.dumps(cmd_input)})
session.xenapi.session.logout()
return json.loads(result)
```

CHAPTER 12

12. Install Dashboard (horizon) on controller

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/horizon-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/horizon-verify.html>

Step 3 has specific changes for the use of XenServer.

1. Install horizon packages:

```
# yum install openstack-dashboard
```

2. Configure horizon. Replace *TIME_ZONE* with your own (for example “America/Chicago”):

```
# vim /etc/openstack-dashboard/local_settings

OPENSTACK_CONTROLLER = "controller"
ALLOWED_HOSTS = ['*', ]
CACHES = {
    'default': {
        'BACKEND': 'django.core.cache.backends.memcached.MemcachedCache',
        'LOCATION': '127.0.0.1:11211',
    }
}
OPENSTACK_KEYSTONE_DEFAULT_ROLE = "user"
OPENSTACK_NEUTRON_NETWORK = {
    'enable_router': False,
    'enable_quotas': False,
    'enable_distributed_router': False,
    'enable_ha_router': False,
    'enable_lb': False,
    'enable_firewall': False,
    'enable_vpnservice': False,
    'enable_fip_topology_check': False,
}
TIME_ZONE = "*TIME_ZONE*"
```

```
OPENSTACK_API_VERSIONS = {
    "data-processing": 1.1,
    "identity": 3,
    "volume": 2,
}
```

- Note 1: There are many options already present in the file. These should be left as-is.
 - Note 2: For the openstack_neutron_network block, modify the settings listed above, rather than replacing the entire block.
3. There is a bug in Horizon which is breaking image metadata when editing XenServer images. This has been reported in <https://bugs.launchpad.net/horizon/+bug/1539722>. Until the bug is fixed, here is a quick and dirty patch to avoid the problem:

- (a) Open the forms.py file:

```
# vim /usr/share/openstack-dashboard/openstack_dashboard/dashboards/project/
  ↳images/images/forms.py
```

- (b) Locate the following lines (should be lines 60 and 61):

```
else:
    container_format = 'bare'
```

- (c) Add the following two lines above those lines:

```
elif disk_format == 'vhd':
    container_format = 'ovf'
```

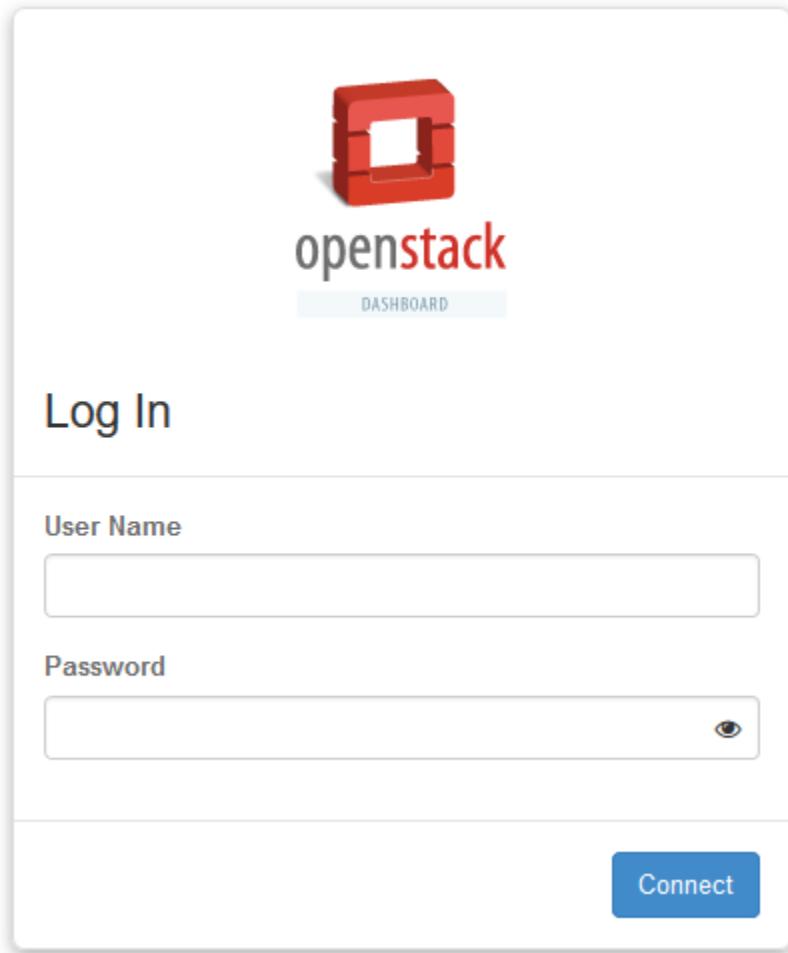
- (d) The whole section should now read:

```
elif disk_format == 'vhd':
    container_format = 'ovf'
else:
    container_format = 'bare'
```

4. Enable and restart the Apache and memcached services:

```
# systemctl enable httpd.service memcached.service
# systemctl restart httpd.service memcached.service
```

5. From a web browser, access http://*CONTROLLER_ADDRESS*/dashboard:



6. Log in using the admin credentials.
7. In the left-hand menu, under “Admin” and then “System”, click on “System Information”. This will display a list of compute services and network agents:

The screenshot shows the OpenStack Horizon dashboard with the following details:

- Header:** openstack logo, admin dropdown, admin dropdown.
- Left Sidebar (Project Admin):**
 - Project dropdown (selected)
 - Admin dropdown (selected)
 - System dropdown (selected)
 - Overview
 - Hypervisors
 - Host Aggregates
 - Instances
 - Flavors
 - Images
 - Networks
 - Defaults
 - Metadata Definitions
 - System Information (highlighted in red)
 - Identity dropdown
- Content Area:**
 - System Information Page:** Services tab selected, Compute Services tab, Network Agents tab. Filter search bar.
 - Table:** Displays system services with the following data:

Name	Host	Zone	Status	State	Last Updated
nova-consoleauth	controller.openstack.lab.eco.rackspace.com	internal	Enabled	Up	0 minutes
nova-scheduler	controller.openstack.lab.eco.rackspace.com	internal	Enabled	Up	0 minutes
nova-conductor	controller.openstack.lab.eco.rackspace.com	internal	Enabled	Up	0 minutes
nova-cert	controller.openstack.lab.eco.rackspace.com	internal	Enabled	Up	0 minutes
nova-compute	compute1-vm.openstack.lab.eco.rackspace.com	nova	Enabled	Up	0 minutes
 - Message:** Displaying 5 items
- Footer:** Version: 8.0.0

The screenshot shows the 'System Information' page of the openstack-xenserver interface. The left sidebar is collapsed, showing a list of system components: Overview, Hypervisors, Host Aggregates, Instances, Flavors, Images, Networks, Defaults, and Metadata Definitions. The 'System Information' section is selected, indicated by a red border.

The main content area is titled 'System Information' and contains a table of network agents. The table has columns for Type, Name, Host, Status, State, and Last Updated. There are four items displayed:

Type	Name	Host	Status	State	Last Updated
Open vSwitch agent	neutron-openvswitch-agent	compute1-vm.openstack.lab.eco.rackspace.com	Enabled	Up	0 minutes
Metadata agent	neutron-metadata-agent	controller.openstack.lab.eco.rackspace.com	Enabled	Up	0 minutes
Open vSwitch agent	neutron-openvswitch-agent	controller.openstack.lab.eco.rackspace.com	Enabled	Up	0 minutes
DHCP agent	neutron-dhcp-agent	controller.openstack.lab.eco.rackspace.com	Enabled	Up	0 minutes

Below the table, a message says 'Displaying 4 items'. At the bottom right of the main content area, the text 'Version: 8.0.0' is visible.

CHAPTER 13

13. Build block1 storage node OS

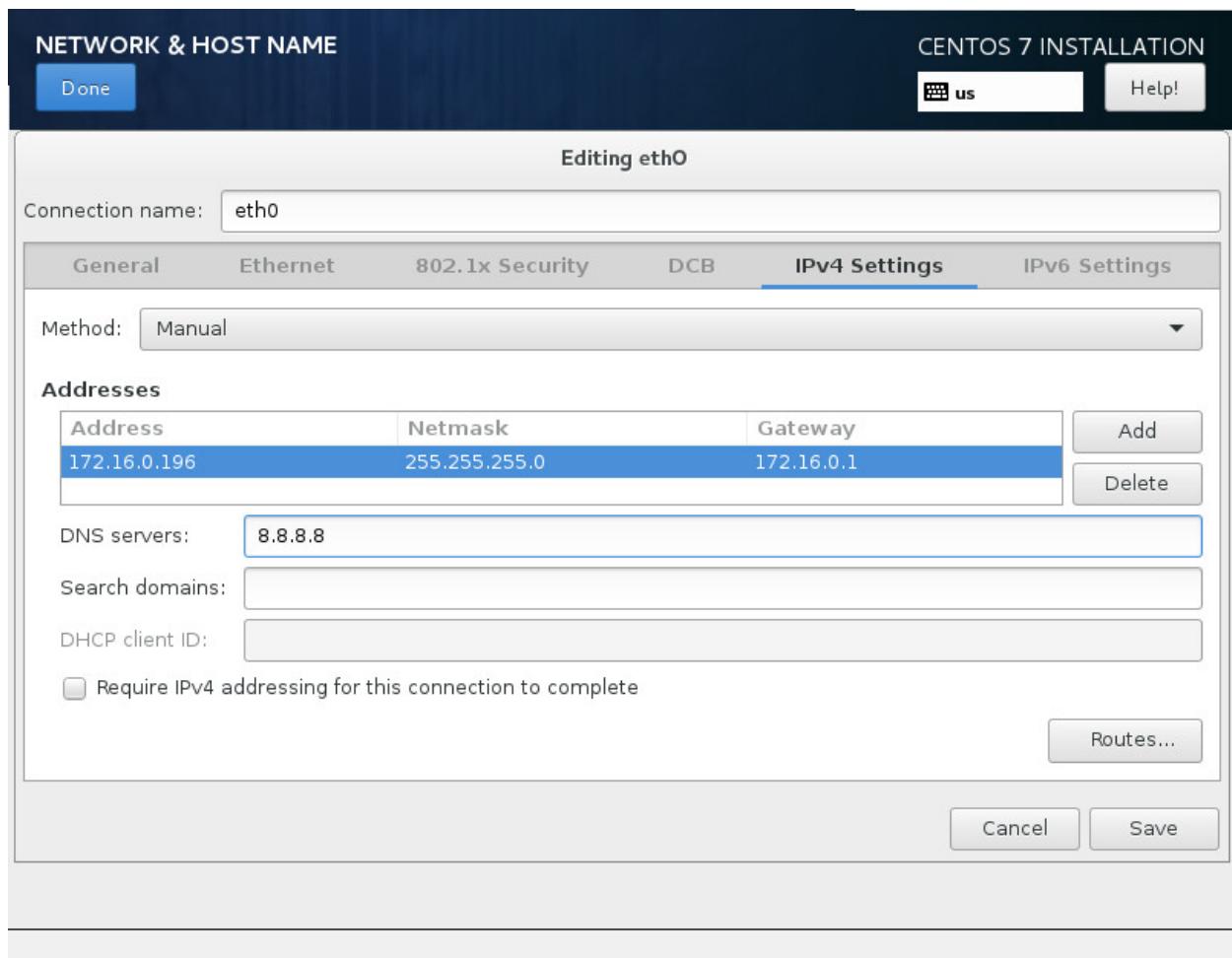
This page is based on the following OpenStack Installation Guide pages:

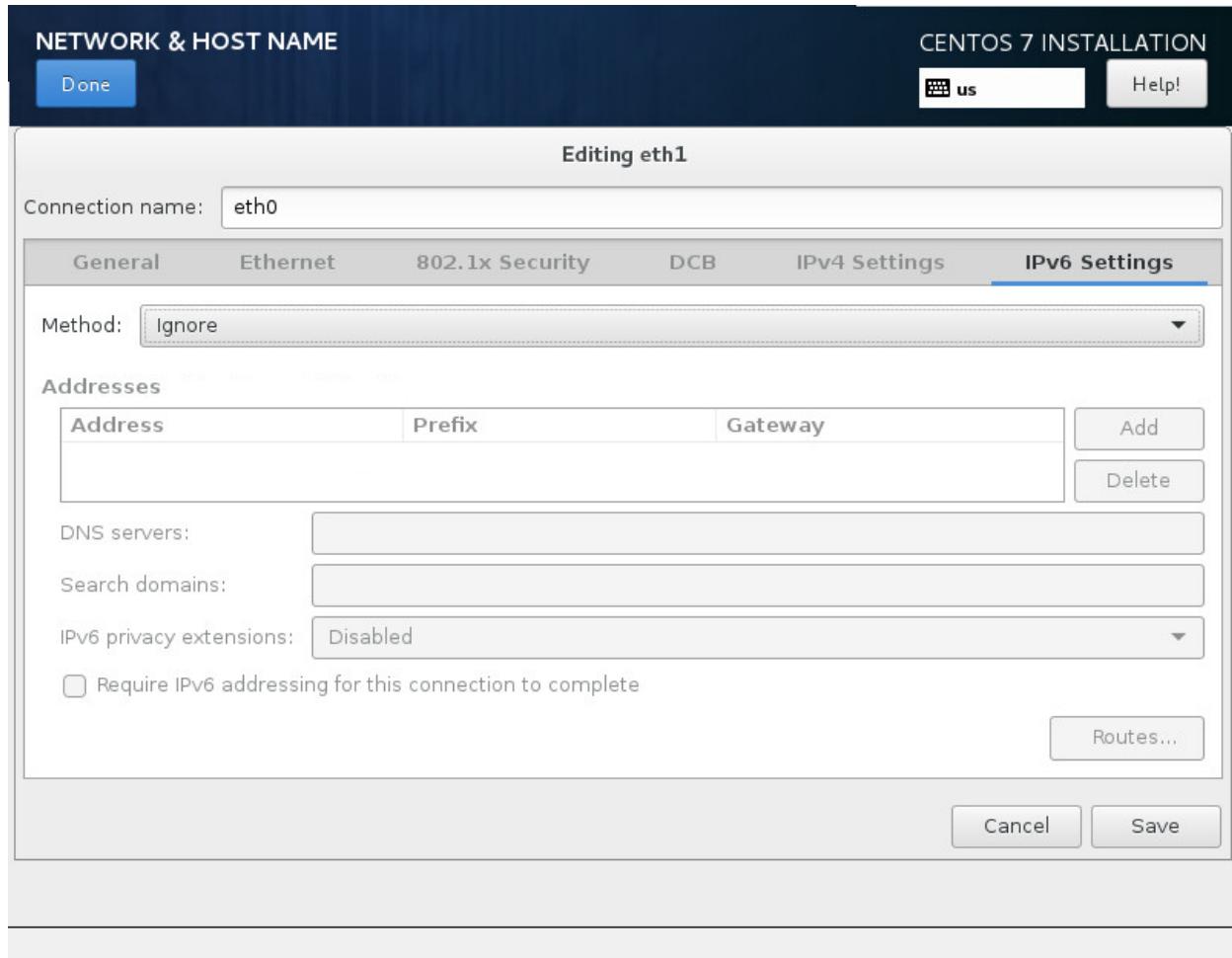
<http://docs.openstack.org/liberty/install-guide-rdo/environment-networking-storage-cinder.html>

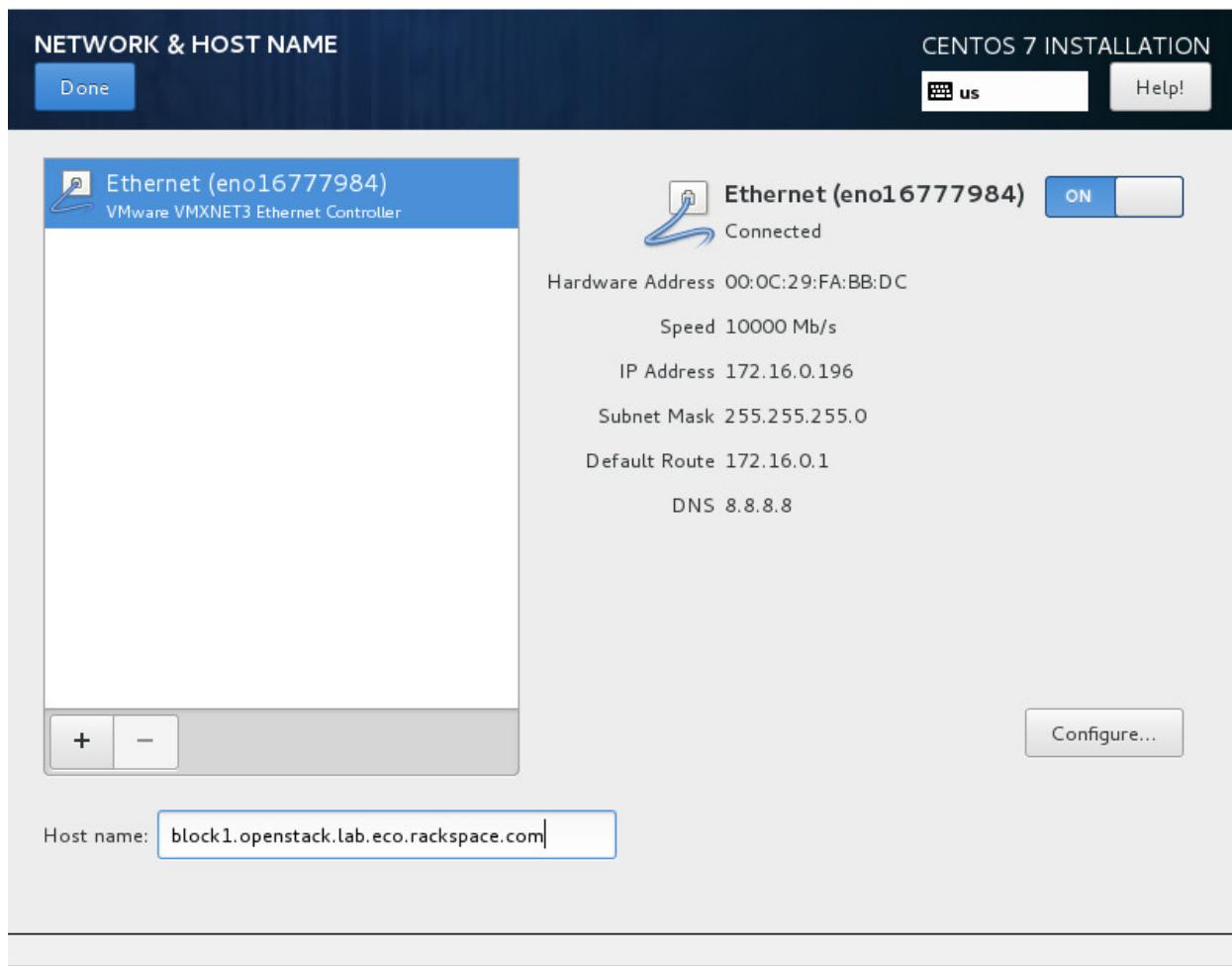
<http://docs.openstack.org/liberty/install-guide-rdo/environment-ntp-other.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-packages.html>

1. The block1 node will need to have a large second disk on which to store the cinder volumes. You may also wish to give it a large amount of storage at /var/lib/cinder/conversion (or /) if you will be writing large images to cinder volumes. It will only need a connection to the Management Network.
2. Boot the control node with the CentOS 7.2.1511 DVD.
3. Set your time zone and language.
4. For “Software Selection”, set this to “Infrastructure Server”.
5. Keep automatic partitioning. Allow to install only on first disk.
6. Set the controller’s IPv4 address and hostname. Disable IPv6. Give the connection the name “eth0”.







7. Click on “Begin Installation”.
8. Set a good root password.
9. Once installation is complete, reboot the server, and remove the DVD/ISO from the server.
10. SSH in to server as root.
11. Stop and disable the firewalld service:

```
# systemctl disable firewalld.service  
# systemctl stop firewalld.service
```

12. Disable SELINUX:

```
# setenforce 0  
# vim /etc/sysconfig/selinux  
  
SELINUX=permissive
```

13. Update all packages on the server:

```
# yum update
```

14. If running the control node on VMWare, install the VM tools:

```
# yum install open-vm-tools
```

15. We need persistent network interface names, so we'll configure udev to give us these. Replace `00:00:00:00:00:00` with the MAC address of your block1 node:

```
# vim /etc/udev/rules.d/90-persistent-net.rules

SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="eno*", NAME="eth0"
```

- Note: This file is case-sensitive, and the MAC addresses should be lower-case.
16. Rename the network interface configuration file to `eth0`. Replace `eno00000001` with the name of your control node's interfaces:

```
# cd /etc/sysconfig/network-scripts
# mv ifcfg-eno00000001 ifcfg-eth0
```

17. Modify the interface configuration files, replacing any instances of `eno00000001` (or whatever your interface name is) with `eth0`:

```
# vim ifcfg-eth0

NAME=eth0
DEVICE=eth0
```

18. Reboot the control node:

```
# systemctl reboot
```

19. SSH back in as root after the reboot.

20. Check that ifconfig now shows `eth0`:

```
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 172.16.0.196 netmask 255.255.255.0 broadcast 172.16.0.255
      inet6 fe80::20c:29ff:fe:bbdc prefixlen 64 scopeid 0x20<link>
        ether 00:0c:29:fa:bb:dc txqueuelen 1000 (Ethernet)
          RX packets 322224 bytes 137862468 (131.4 MiB)
          RX errors 0 dropped 35 overruns 0 frame 0
          TX packets 408936 bytes 108141349 (103.1 MiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 0 (Local Loopback)
          RX packets 6 bytes 564 (564.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 6 bytes 564 (564.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

21. Update the system hosts file with entries for all nodes:

```
# vim /etc/hosts

172.16.0.192 controller controller.openstack.lab.eco.rackspace.com
```

```
172.16.0.203 compute1 compute1.openstack.lab.eco.rackspace.com
172.16.0.204 compute1-vm compute1-vm.openstack.lab.eco.rackspace.com
172.16.0.195 compute2 compute2.openstack.lab.eco.rackspace.com
172.16.0.196 block1 block1.openstack.lab.eco.rackspace.com
172.16.0.197 object1 object1.openstack.lab.eco.rackspace.com
172.16.0.198 object2 object2.openstack.lab.eco.rackspace.com
```

22. Update the chrony configuration to use the controller as a time source:

```
# vim /etc/chrony.conf

server controller iburst
```

- Remove any other servers listed, leaving only “controller”.

23. Restart the chrony service, and confirm that “controller” is listed as a source:

```
# systemctl restart chronyd.service
# chronyc sources
 210 Number of sources = 1
   MS Name/IP address          Stratum Poll Reach LastRx Last sample
   ========
   ^* controller                  3      6     17      6 -3374ns [+2000ns] +/- 6895us
```

24. Enable the OpenStack-Liberty yum repository:

```
# yum install centos-release-openstack-liberty
```

25. Install the OpenStack client and SELINUX support:

```
# yum install python-openstackclient openstack-selinux
```

CHAPTER 14

14. Install Block Storage (cinder) on controller

This page is based on the following OpenStack Installation Guide page:

<http://docs.openstack.org/liberty/install-guide-rdo/cinder-controller-install.html>

1. Open the MySQL client and create the “cinder” database. Replace `*CINDER_DBPASS*` with your own:

```
# mysql
> create database cinder;
> grant all privileges on cinder.* to 'cinder'@'localhost' identified by
-> '*CINDER_DBPASS*';
> grant all privileges on cinder.* to 'cinder'@'%' identified by '*CINDER_
->DBPASS*';
> quit
```

2. Create the “cinder” user, role, services and endpoints. Provide `*CINDER_PASS*` when prompted:

```
# source admin-openrc.sh
# openstack user create --domain default --password-prompt cinder
# openstack role add --project service --user cinder admin
# openstack service create --name cinder --description "OpenStack Block Storage"
-> volume
# openstack service create --name cinderv2 --description "OpenStack Block Storage
-> " volumev2
# openstack endpoint create --region RegionOne volume public http://
-> controller:8776/v1/%\{(tenant_id\)\}s
# openstack endpoint create --region RegionOne volume internal http://
-> controller:8776/v1/%\{(tenant_id\)\}s
# openstack endpoint create --region RegionOne volume admin http://
-> controller:8776/v1/%\{(tenant_id\)\}s
# openstack endpoint create --region RegionOne volumev2 public http://
-> controller:8776/v2/%\{(tenant_id\)\}s
# openstack endpoint create --region RegionOne volumev2 internal http://
-> controller:8776/v2/%\{(tenant_id\)\}s
# openstack endpoint create --region RegionOne volumev2 admin http://
-> controller:8776/v2/%\{(tenant_id\)\}s
```

3. Install the cinder packages:

```
# yum install openstack-cinder python-cinderclient
```

4. Configure cinder. Replace *SERVER_IP*, *CINDER_DBPASS*, *CINDER_PASS* and *RABBIT_PASS* with your own:

```
# vim /etc/cinder/cinder.conf

[database]
connection = mysql://cinder:*CINDER_DBPASS*@controller/cinder

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
my_ip = *SERVER_IP*
nova_catalog_info = compute:nova:publicURL
nova_catalog_admin_info = compute:nova:adminURL

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = cinder
password = *CINDER_PASS*

[oslo_concurrency]
lock_path = /var/lib/cinder/tmp
```

5. Populate the cinder database:

```
# su -s /bin/sh -c "cinder-manage db sync" cinder
```

6. Reconfigure nova for cinder:

```
# vim /etc/nova/nova.conf

[cinder]
os_region_name = RegionOne
```

7. Restart the nova service:

```
# systemctl restart openstack-nova-api.service
```

8. Enable and start the cinder services:

```
# systemctl enable openstack-cinder-api.service openstack-cinder-scheduler.service
# systemctl start openstack-cinder-api.service openstack-cinder-scheduler.service
```

CHAPTER 15

15. Install Block Storage (cinder) on storage node

This page is based on the following OpenStack Installation Guide page:

<http://docs.openstack.org/liberty/install-guide-rdo/cinder-storage-install.html>

Steps 3, 4, 5, 6, 8, 9 and 10 have specific changes for the use of XenServer.

1. Create the LVM volume group on the second disk:

```
# pvcreate /dev/sdb
# vgcreate cinder-volumes /dev/sdb
```

2. Update the LVM configuration to prevent scanning of cinder volumes' contents:

```
# vim /etc/lvm/lvm.conf

devices {
    ...
    filter = [ "a/sda/", "a/sdb/", "r/.*/" ]
```

- Note: Do not replace the entire “devices” section, only the “filter” line.

3. **Enable the centos-virt-xen and epel-release repositories:**

```
# yum install centos-release-xen epel-release
```

4. **Disable kernel updates from the centos-virt-xen repository:**

```
# vim /etc/yum.repos.d/CentOS-Xen.repo

[centos-virt-xen]
exclude=kernel*
```

5. **Install special packages needed from outside of the openstack-liberty repositories:**

```
# yum install scsi-target-utils xen-runtime
```

6. Remove the epel-release repository again:

```
# yum remove epel-release
```

7. Install the cinder packages:

```
# yum install openstack-cinder python-oslo-policy
```

8. Configure cinder. Replace *CINDER_DBPASS*, *SERVER_IP*, *RABBIT_PASS* and *CINDER_PASS* with your own:

```
# vim /etc/cinder/cinder.conf

[database]
connection = mysql://cinder:*CINDER_DBPASS*@controller/cinder

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
my_ip = *SERVER_IP*
enabled_backends = lvm
glance_host = controller

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_authtoken]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = cinder
password = *CINDER_PASS*

[lvm]
volume_driver = cinder.volume.drivers.lvm.LVMVolumeDriver
volume_group = cinder-volumes
iscsi_protocol = iscsi
iscsi_helper = tgtadm

[oslo_concurrency]
lock_path = /var/lib/cinder/tmp
```

9. Update the tgtd.conf configuration. There are other lines in this file. Don't change those, just add this one:

```
# vim /etc/tgt/tgtd.conf

include /var/lib/cinder/volumes/*
```

10. Enable and start the tgtd and cinder services:

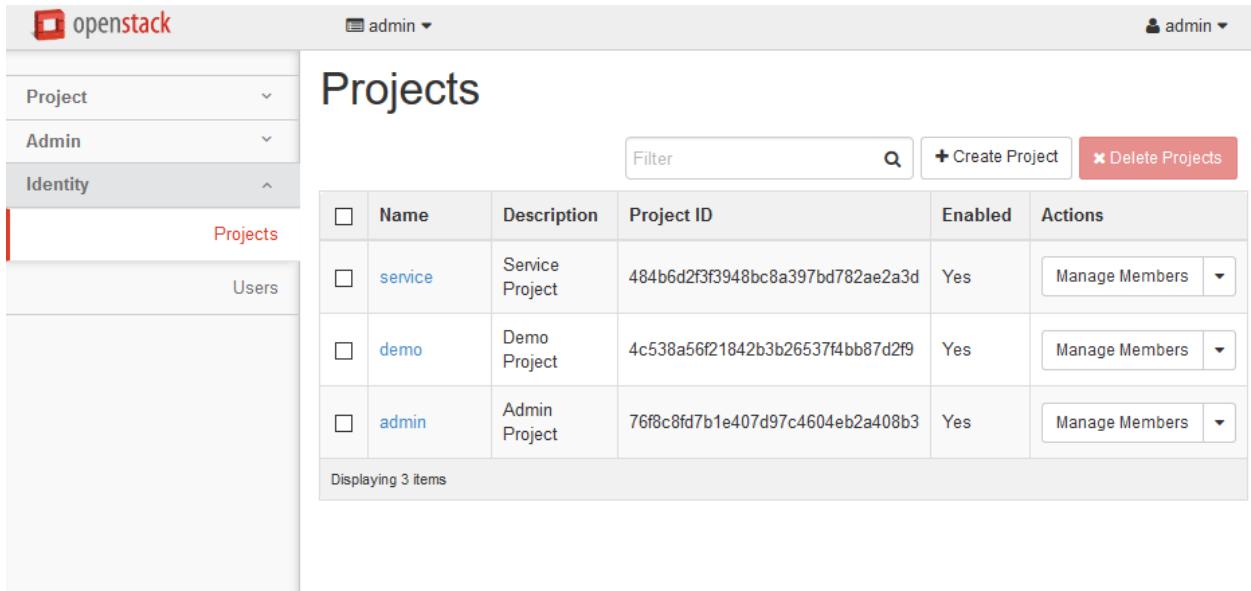
```
# systemctl enable tgtd.service openstack-cinder-volume.service
# systemctl start tgtd.service openstack-cinder-volume.service
```

CHAPTER 16

16. Fix cinder quotas for the demo project

This page is not based on the OpenStack Installation Guide. I found that a bug causes nova to believe that the demo project has a 0 quota for cinder volumes, even though neutron states that the quota is 10. Re-saving the value populates the value properly in nova.

1. From a web browser, access `http://*CONTROLLER_ADDRESS*/dashboard`
2. Log in using the admin credentials.
3. In the left-hand menu, under “Identity”, click on “Projects”:



The screenshot shows the OpenStack dashboard with the "Projects" page selected. The left sidebar has "Project" selected. The main content area displays a table of projects with the following data:

<input type="checkbox"/>	Name	Description	Project ID	Enabled	Actions
<input type="checkbox"/>	service	Service Project	484b6d2f3f3948bc8a397bd782ae2a3d	Yes	<button>Manage Members</button>
<input type="checkbox"/>	demo	Demo Project	4c538a56f21842b3b26537f4bb87d2f9	Yes	<button>Manage Members</button>
<input type="checkbox"/>	admin	Admin Project	76f8c8fd7b1e407d97c4604eb2a408b3	Yes	<button>Manage Members</button>

At the bottom of the table, it says "Displaying 3 items".

4. In the “Actions” drop-down for the “demo” project, select modify quotas:

Edit Project

Project Information * Project Members Quota *

Metadata Items *	128
VCPUs *	20
Instances *	10
Injected Files *	5
Injected File Content (Bytes) *	10240
Volumes *	10
Volume Snapshots *	10
Total Size of Volumes and Snapshots (GB) *	1000
RAM (MB) *	51200

5. Don't make any changes. Just click "Save".

CHAPTER 17

17. Launch a test Boot-From-Volume instance from Horizon

This page is not based on the OpenStack Installation Guide.

1. From a web browser, access http://*CONTROLLER_ADDRESS*/dashboard.
2. Log in using the demo credentials.
3. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”. Click on Launch instance:

The screenshot shows the OpenStack Horizon dashboard. The top navigation bar has 'demo' selected for both project and user. The left sidebar is under the 'Compute' section, with 'Instances' selected. The main content area is titled 'Instances' and displays a table with the following columns: Instance Name, Image Name, IP Address, Size, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. A message at the bottom of the table says 'No items to display.' and 'Displaying 0 items'. There is a 'Launch Instance' button at the top right of the table header.

4. Give the instance the name “test bfv”, and select “Boot from image (creates a new volume)” and the “cirros-xen” image. Launch the instance:

Launch Instance

[Details *](#) [Access & Security](#) [Networking *](#) [Post-Creation](#) [Advanced Options](#)

Availability Zone

Instance Name *

Flavor * ?

Instance Count * ?

Instance Boot Source * ?

Image Name *

Device size (GB) * ?

Delete on Terminate ?

Specify the details for launching an instance.
The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	m1.tiny
VCPUs	1
Root Disk	1 GB
Ephemeral Disk	0 GB
Total Disk	1 GB
RAM	512 MB

Project Limits

Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

[Cancel](#) [Launch](#)

- Once the instance enters “Active” status, click on its name:

The screenshot shows the 'Instance Details' page for an instance named 'test bfv'. The top navigation bar includes 'demo' dropdowns for project, user, and a 'Create Snapshot' button. The left sidebar has sections for Project, Compute (selected), Overview, Instances, Volumes, Images, Access & Security, Network (selected), and Identity. The main content area displays the instance overview with the following details:

Name	test bfv
ID	d37916d2-50dd-4ef2-a547-08027b9e0622
Status	Active
Availability Zone	nova
Created	Jan. 29, 2016, 2:08 p.m.
Time Since Created	1 minute
Specs	
Flavor	m1.tiny
Flavor ID	1
RAM	512MB
VCPUs	1 VCPU
Disk	1GB
IP Addresses	
Public	192.168.100.67
Security Groups	
default	ALLOW IPv4 from default ALLOW IPv6 to ::/0 ALLOW IPv4 to 0.0.0.0/0 ALLOW IPv6 from default
Metadata	
Key Name	None
Image Name	None

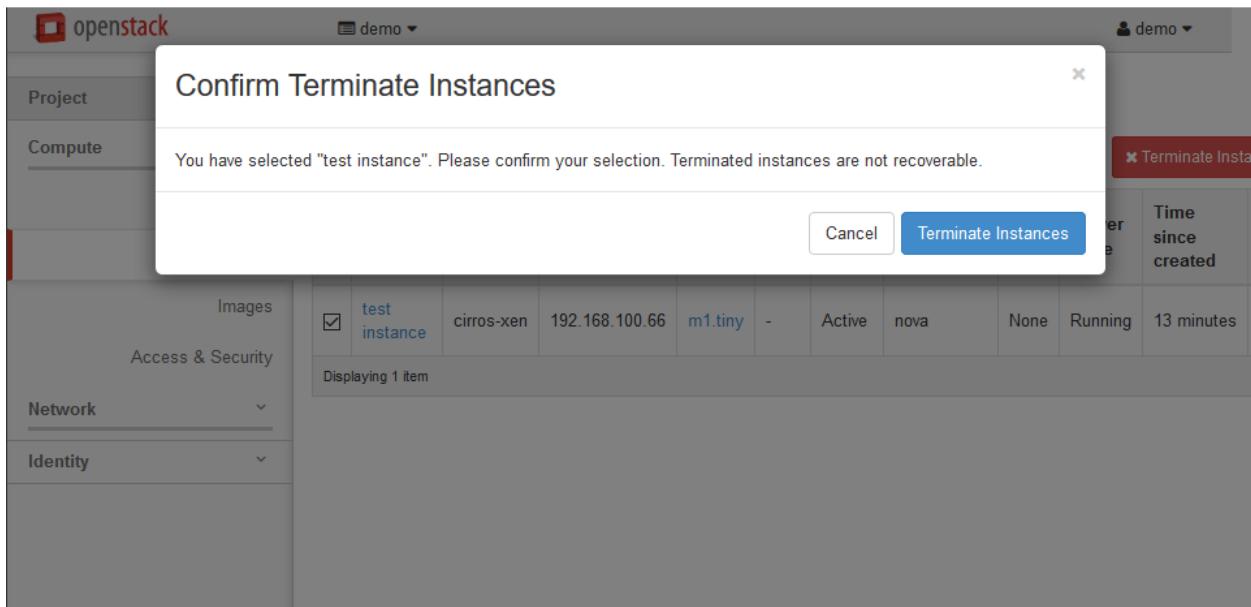
- Click on the “Console” tab, and you should see the instance booting. Wait for the login prompt:

```

Connected (unencrypted) to: XenServer Virtual Terminal
Send CtrlAltDel
[ 1.180441] mousedev: PS/2 mouse device common for all mice
[ 1.220417] rtc_cmos rtc_cmos: rtc core: registered rtc_cmos as rtc0
[ 1.220564] rtc_cmos: probe of rtc_cmos failed with error -38
[ 1.220772] device-mapper: uevent: version 1.0.3
[ 1.220932] device-mapper: ioctl: 4.22.0-ioctl (2011-10-19) initialised: dm-d
evel@redhat.com
[ 1.221052] EFI Variables Facility v0.08 2004-May-17
[ 1.221449] TCP cubic registered
[ 1.221636] NET: Registered protocol family 10
[ 1.222659] NET: Registered protocol family 17
[ 1.222729] Registering the dns_resolver key type
[ 1.222971] registered taskstats version 1
[ 1.233800] blkfront: xuda: flush diskcache: enabled
[ 1.235693] xuda: xuda1
[ 1.328095] Magic number: 1:252:3141
[ 1.328236] /build/buildd/linux-3.2.0/drivers/rtc/hctosys.c: unable to open r
tc device (rtc0)
[ 1.328351] BIOS EDD facility v0.16 2004-Jun-25, 0 devices found
[ 1.328428] EDD information not available.
[ 1.329113] Freeing unused kernel memory: 928k freed
[ 1.329642] Write protecting the kernel read-only data: 12288k
[ 1.336735] Freeing unused kernel memory: 1596k freed
[ 1.338111] Freeing unused kernel memory: 1184k freed

```

7. Once the login prompt has appeared, check that you can ping and SSH to the instance. The credentials are:
 - Username: `cirros`
 - Password: `cubswin:)`
8. In the left-hand menu, click on “Instances” again, select the “test instance” in the list and click on “Terminate Instances”:



CHAPTER 18

18. Build KVM Host

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/environment-networking-compute.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-ntp-other.html>

<http://docs.openstack.org/liberty/install-guide-rdo/environment-packages.html>

1. In this guide I am using a server with a small RAID-1 for the OS, and a large RAID-10 for the VMs. There are four network interfaces, although only the first two are in use.
2. Boot the KVM host with the CentOS 7.2.1511 DVD.
3. Set your time zone and language.
4. For “Software Selection”, set this to “Infrastructure Server”.
5. Keep automatic partitioning. Allow to install only on first disk.
6. Set the node’s IPv4 address on the management network interface and disable IPv6. Give the connection the name “eth1”. Set the node’s hostname:

Editing eth1

Connection name: **eth1**

General Ethernet 802.1x Security DCB **IPv4 Settings** IPv6 Settings

Method: **Manual**

Addresses

Address	Netmask	Gateway	
172.16.0.195	255.255.255.0	172.16.0.1	Add
			Delete

DNS servers: **8.8.8.8**

Search domains:

DHCP client ID:

Require IPv4 addressing for this connection to complete

Routes...

Cancel **Save**

The screenshot shows a network configuration interface for a connection named 'eth1'. The 'IPv4 Settings' tab is active. The connection method is set to 'Manual'. An address entry is present with the details: Address 172.16.0.195, Netmask 255.255.255.0, and Gateway 172.16.0.1. There are also fields for DNS servers (set to 8.8.8.8), search domains, and DHCP client ID. A checkbox for requiring IPv4 addressing is checked. At the bottom, there are buttons for routes, cancel, and save.

NETWORK & HOST NAME

CENTOS 7 INSTALLATION

Done Help!

Editing eth1

Connection name: eth1

General Ethernet 802.1x Security DCB IPv4 Settings **IPv6 Settings**

Method: Ignore

Addresses

Address	Prefix	Gateway	Add	Delete

DNS servers:

Search domains:

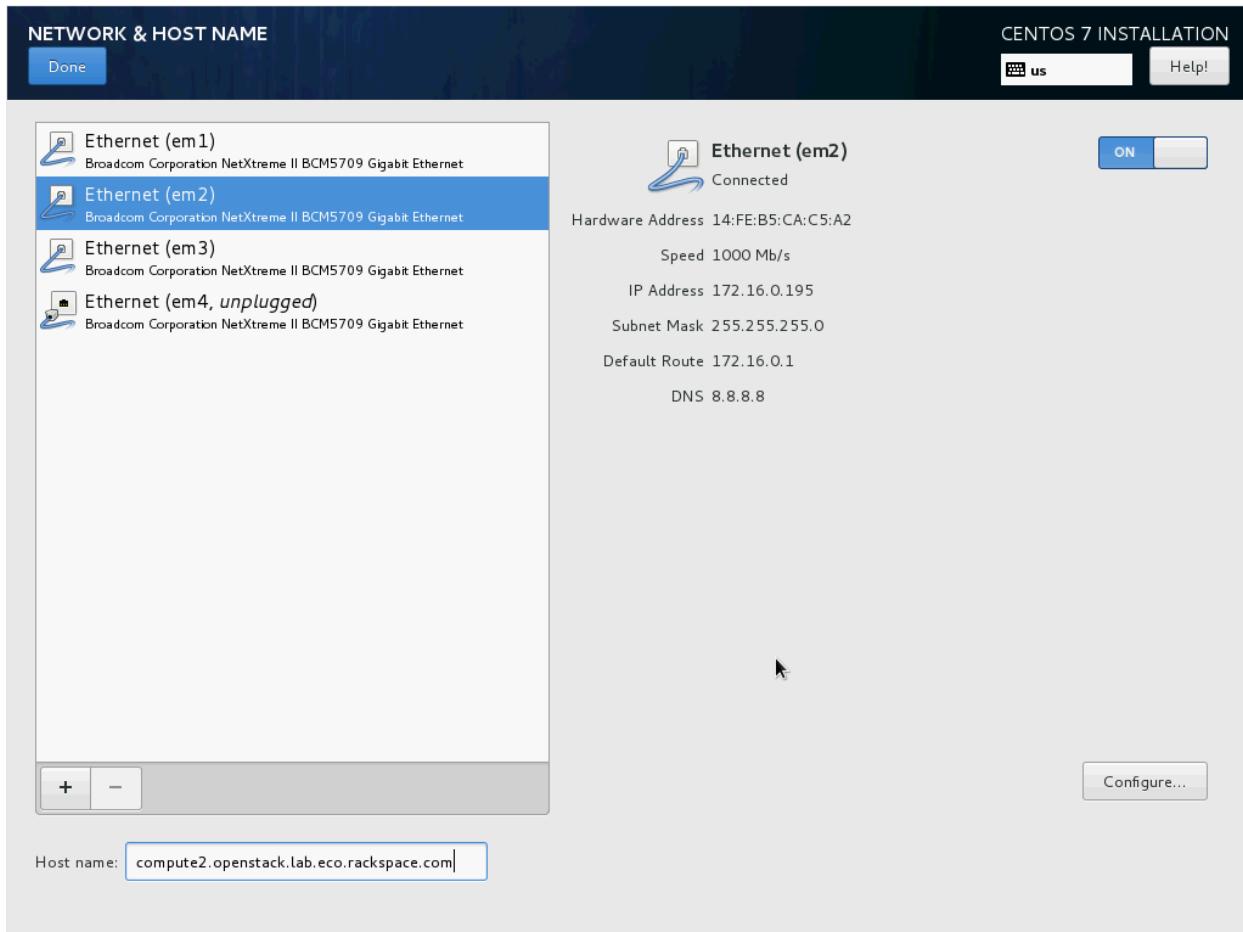
IPv6 privacy extensions: Disabled

Require IPv6 addressing for this connection to complete

Routes...

Cancel Save

This screenshot shows the 'Editing eth1' configuration screen for a network connection. The 'IPv6 Settings' tab is active. The 'Method' is set to 'Ignore'. There are no entries in the 'Addresses' table. The 'DNS servers' and 'Search domains' fields are empty. The 'IPv6 privacy extensions' dropdown is set to 'Disabled'. The 'Require IPv6 addressing for this connection to complete' checkbox is not checked. At the bottom right, there are 'Routes...', 'Cancel', and 'Save' buttons.



7. Click on “Begin Installation”.
8. Set a good root password.
9. Once installation is complete, reboot the server, and remove the DVD/ISO from the server.
10. SSH in to server as root.
11. Stop and disable the firewalld service:

```
# systemctl disable firewalld.service
# systemctl stop firewalld.service
```

12. Disable SELINUX:

```
# setenforce 0
# vim /etc/sysconfig/selinux

SELINUX=permissive
```

13. Update all packages on the server:

```
# yum update
```

14. We need persistent network interface names, so we'll configure udev to give us these. Replace 00:00:00:00:00:00 with the MAC addresses of your KVM node:

```
# vim /etc/udev/rules.d/90-persistent-net.rules

SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="em*", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="em*", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="em*", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:00:00:00:00:00"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x0", ATTR{type}=="1", KERNEL=="em*", NAME="eth3"
```

- Note: This file is case-sensitive, and the MAC addresses should be lower-case.
15. Rename the network interface configuration files to eth0 and eth1. Replace em1 , em2 , em3 and em4 with the names of your KVM node's interfaces:

```
# cd /etc/sysconfig/network-scripts
# mv ifcfg-em1 ifcfg-eth0
# mv ifcfg-em2 ifcfg-eth1
# mv ifcfg-em3 ifcfg-eth2
# mv ifcfg-em4 ifcfg-eth3
```

16. Modify the interface configuration files, replacing any instances of em1 , em2 , em3 , em4 (or whatever your interface names are) with eth0 , eth1 , eth2 and eth3 respectively:

```
# vim ifcfg-eth0

NAME=eth0
DEVICE=eth0

# vim ifcfg-eth1

NAME=eth1
DEVICE=eth1

# vim ifcfg-eth2

NAME=eth2
DEVICE=eth2

# vim ifcfg-eth3

NAME=eth3
DEVICE=eth3
```

17. Reboot the KVM node:

```
# systemctl reboot
```

18. SSH back in as root after the reboot.

19. Check that ifconfig now shows eth0 , eth1 , eth2 and eth3:

```
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 14:fe:b5:ca:c5:a0 txqueuelen 1000  (Ethernet)
      RX packets 1195904 bytes 1012346616 (965.4 MiB)
      RX errors 0 dropped 0 overruns 0 frame 0
```

```
TX packets 366843 bytes 28571196 (27.2 MiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 172.16.0.195 netmask 255.255.255.0 broadcast 172.16.0.255
      inet6 fe80::16fe:b5ff:fea:c5a2 prefixlen 64 scopeid 0x20<link>
        ether 14:fe:b5:ca:c5:a2 txqueuelen 1000 (Ethernet)
          RX packets 12004890 bytes 15236092868 (14.1 GiB)
          RX errors 0 dropped 156 overruns 0 frame 0
          TX packets 12647929 bytes 15934829339 (14.8 GiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 14:fe:b5:ca:c5:a4 txqueuelen 1000 (Ethernet)
      RX packets 1985034 bytes 180158767 (171.8 MiB)
      RX errors 0 dropped 252 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 14:fe:b5:ca:c5:a6 txqueuelen 1000 (Ethernet)
      RX packets 0 bytes 0 (0.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 0 (Local Loopback)
        RX packets 9855259 bytes 517557258 (493.5 MiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 9855259 bytes 517557258 (493.5 MiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

20. Update the system hosts file with entries for all nodes:

```
# vim /etc/hosts

172.16.0.192 controller controller.openstack.lab.eco.rackspace.com
172.16.0.203 compute1 compute1.openstack.lab.eco.rackspace.com
172.16.0.204 compute1-vm compute1-vm.openstack.lab.eco.rackspace.com
172.16.0.195 compute2 compute2.openstack.lab.eco.rackspace.com
172.16.0.196 block1 block1.openstack.lab.eco.rackspace.com
172.16.0.197 object1 object1.openstack.lab.eco.rackspace.com
172.16.0.198 object2 object2.openstack.lab.eco.rackspace.com
```

21. Update the chrony configuration to use the controller as a time source:

```
# vim /etc/chrony.conf

server controller iburst
```

- Remove any other servers listed, leaving only “controller”.

22. Restart the chrony service, and confirm that “controller” is listed as a source:

```
# systemctl restart chronyd.service
# chronyc sources
 210 Number of sources = 1
 MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
 ^* controller                3      6     17      6 -3374ns [+2000ns] +/- 6895us
```

23. Enable the OpenStack-Liberty yum repository:

```
# yum install centos-release-openstack-liberty
```

24. Install the OpenStack client and SELINUX support:

```
# yum install python-openstackclient openstack-selinux
```


CHAPTER 19

19. Install Compute (nova) on KVM Host

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/nova-compute-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/cinder-storage-install.html>

<http://docs.openstack.org/liberty/install-guide-rdo/nova-verify.html>

1. Install nova packages:

```
# yum install openstack-nova-compute sysfsutils
```

2. Format and mount the second array for instance storage:

```
# parted -s -- /dev/sdb mklabel gpt
# parted -s -a optimal -- /dev/sdb mkpart primary 2048s -1
# parted -s -- /dev/sdb align-check optimal 1
# parted /dev/sdb set 1 lvm on
# parted /dev/sdb unit s print
# mkfs.xfs /dev/sdb1
# mount /dev/sdb1 /var/lib/nova/instances
# tail -1 /etc/mtab >> /etc/fstab
# chown nova:nova /var/lib/nova/instances
```

3. Update the LVM configuration to prevent scanning of instances' contents:

```
# vim /etc/lvm/lvm.conf

devices {
  ...
  filter = [ "a/sda/", "a/sdb/", "r/.*/"]
```

- Note: Do not replace the entire “devices” section, only the “filter” line.

4. Configure nova. Replace *SERVER_IP*, *RABBIT_PASS*, *NOVA_PASS* and *CONTROLLER_ADDRESS* with your own:

```
# vim /etc/nova/nova.conf

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone
my_ip = *SERVER_IP*
network_api_class = nova.network.neutronv2.api.API
security_group_api = neutron
linuxnet_interface_driver = nova.network.linux_net.
↳NeutronLinuxBridgeInterfaceDriver
firewall_driver = nova.virt.firewall.NoopFirewallDriver

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_authtoken]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = nova
password = *NOVA_PASS*

[vnc]
enabled = True
vncserver_listen = 0.0.0.0
vncserver_proxyclient_address = $my_ip
novncproxy_base_url = http://*CONTROLLER_ADDRESS*:6080/vnc_auto.html

[glance]
host = controller

[oslo_concurrency]
lock_path = /var/lib/nova/tmp

[libvirt]
virt_type = kvm
```

5. Enable and start the nova and libvirt services:

```
# systemctl enable libvirtd.service openstack-nova-compute.service
# systemctl start libvirtd.service openstack-nova-compute.service
```

6. Log on to the control node as root.

7. Load the “admin” credential file:

```
# source admin-openrc.sh
```

8. Check the nova service list:

```
# nova service-list
+----+-----+-----+-----+
↳+-----+-----+-----+-----+
```

Id Binary	Host		Zone
↳ Status	State	Updated_at	Disabled Reason
1 nova-consoleauth	controller.openstack.lab.eco.rackspace.com	-	↳
↳internal enabled	up	2016-02-09T17:19:38.000000	-
2 nova-scheduler	controller.openstack.lab.eco.rackspace.com	-	↳
↳internal enabled	up	2016-02-09T17:19:41.000000	-
3 nova-conductor	controller.openstack.lab.eco.rackspace.com	-	↳
↳internal enabled	up	2016-02-09T17:19:41.000000	-
4 nova-cert	controller.openstack.lab.eco.rackspace.com	-	↳
↳internal enabled	up	2016-02-09T17:19:38.000000	-
5 nova-compute	compute1-vm.openstack.lab.eco.rackspace.com	nova	↳
↳ enabled	up	2016-02-09T17:19:39.000000	-
6 nova-compute	compute2.openstack.lab.eco.rackspace.com	nova	↳
↳ enabled	up	2016-02-09T17:19:36.000000	-

- The list should include `compute1-vm` and `compute2` running `nova-compute`.

CHAPTER 20

20. Install Networking (neutron) on KVM Host

This page is based on the following OpenStack Installation Guide pages:

<http://docs.openstack.org/liberty/install-guide-rdo/neutron-compute-install.html>

All steps except 2 have modifications for XenServer.

1. **Install the neutron and ovs packages:**

```
# yum install openstack-neutron openstack-neutron-openvswitch ebtables ipset
˓→openvswitch
```

2. Configure neutron. Replace *RABBIT_PASS* and *NEUTRON_PASS* with your own:

```
# vim /etc/neutron/neutron.conf

[DEFAULT]
rpc_backend = rabbit
auth_strategy = keystone

[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = *RABBIT_PASS*

[keystone_auth_token]
auth_uri = http://controller:5000
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
project_name = service
username = neutron
password = *NEUTRON_PASS*

[oslo_concurrency]
lock_path = /var/lib/neutron/tmp
```

- Make sure that any connection options under [database] are deleted or commented-out.
- Delete or comment-out any pre-existing lines in the [keystone_auth_token] section.

3. Configure the neutron ovs agent. Replace *XAPI_BRIDGE* with your own:

```
# vim /etc/neutron/plugins/ml2/openvswitch_agent.ini

[ovs]
integration_bridge = *XAPI_BRIDGE*
bridge_mappings = public:br-eth0

[securitygroup]
firewall_driver = neutron.agent.firewall.NoopFirewallDriver
```

4. Reconfigure nova to use neutron. Replace *NEUTRON_PASS* and *XAPI_BRIDGE* with your own:

```
# vim /etc/nova/nova.conf

[neutron]
url = http://controller:9696
auth_url = http://controller:35357
auth_plugin = password
project_domain_id = default
user_domain_id = default
region_name = RegionOne
project_name = service
username = neutron
password = *NEUTRON_PASS*
ovs_bridge = *XAPI_BRIDGE*

[DEFAULT]
linuxnet_ovs_integration_bridge = *XAPI_BRIDGE*
```

5. Enable and start the ovs service:

```
# systemctl enable openvswitch.service
# systemctl start openvswitch.service
```

6. Set up the ovs bridge to the public network:

```
# ovs-vsctl add-br br-eth0
# ovs-vsctl add-port br-eth0 eth0
```

7. Enable and start the neutron service:

```
# systemctl enable neutron-openvswitch-agent.service
# systemctl start neutron-openvswitch-agent.service
```

CHAPTER 21

21. Update images for dual-hypervisor environment

This page is not based on the OpenStack Installation Guide.

1. Log on to the controller node as root.
2. Download the cirros image for KVM hypervisors:

```
# wget http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86_64-disk.img
```

3. Upload the image to glance:

```
# source admin-openrc.sh
# glance image-create --name "cirros-kvm" --file cirros-0.3.4-x86_64-disk.img --
#   disk-format qcow2 --container-format bare --visibility public --progress
```

4. From a web browser, access http://*CONTROLLER_ADDRESS*/dashboard
5. Log in using the admin credentials.
6. In the left-hand menu, under “Admin”, and then “System”, click on “Images”. Click on the “cirros-kvm” image:

The screenshot shows the OpenStack Xenserver interface. The top navigation bar has 'admin' in the top-left and top-right corners. The left sidebar menu is collapsed under 'Project'. The main content area displays 'Image Details: cirros-kvm'. A red box highlights the 'Edit Image' button in the top right. Below it, the 'Image Overview' section shows the image name 'cirros-kvm' and its ID 'fdf88ac1-0bb8-40bf-a45f-707c201fa8a5'. The 'Information' section lists various metadata fields:

Name	cirros-kvm
ID	fdf88ac1-0bb8-40bf-a45f-707c201fa8a5
Owner	76f8c8fd7b1e407d97c4604eb2a408b3
Status	Active
Public	Yes
Protected	No
Checksum	ee1eca47dc88f4879d8a229cc70a07c6
Created	Feb. 1, 2016, 2:19 p.m.
Updated	Feb. 1, 2016, 2:19 p.m.

The 'Specs' section shows the image's size as 12.7 MB and its container format as BARE, with a disk format of QCOW2. The 'Custom Properties' section is currently empty.

7. In the top-right drop-down, click on “Update Metadata”:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

The screenshot shows a user interface for managing image metadata. It consists of two main sections: 'Available Metadata' on the left and 'Existing Metadata' on the right. The 'Available Metadata' section includes a search bar with a magnifying glass icon and a 'Filter' button. Below this is a 'Custom' input field and a blue '+' button. A message 'No available metadata' is displayed. The 'Existing Metadata' section also has a search bar with a magnifying glass icon and a 'Filter' button. It displays the message 'No existing metadata'.

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

8. On the left-hand side, in the “custom” box, enter “hypervisor_type”, and then click on the + button:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

The screenshot shows a user interface for managing image metadata. On the left, under 'Available Metadata', there is a 'Custom' section containing the key 'hypervisor_type' and a blue '+' button. A search bar with a magnifying glass icon is positioned above this section. Below it, a message says 'No available metadata'. On the right, under 'Existing Metadata', there is a search bar and a message saying 'No existing metadata'.

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

10. Now, on the right-hand side, in the “hypervisor_type” box, enter “kvm” and click “Save”:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

The screenshot shows a user interface for managing image metadata. On the left, under 'Available Metadata', there is a 'Custom' section with a '+' button. Below it, a message says 'No available metadata'. On the right, under 'Existing Metadata', there is a row for 'hypervisor_type' set to 'kvm', with a '-' button next to it. At the bottom, a modal window displays the selected 'hypervisor_type' value. At the very bottom, there are 'Cancel' and 'Save' buttons.

Available Metadata

Existing Metadata

No available metadata

hypervisor_type (hypervisor_type)

Cancel Save

11. In the left-hand menu, under “Admin”, and then “System”, again click on “Images”. This time click on the “cirros-xen” image.
12. Again click on “Update Metadata” in the drop-down. Follow the same steps, but set “hypervisor_type” to “xen”.

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

Available Metadata		Filter	Search
Custom	+/-		
No available metadata			

Existing Metadata		Filter	Search
hypervisor_type	hypervisor_type	xen	-
vm_mode	vm_mode	xen	-

hypervisor_type (*hypervisor_type*)

CHAPTER 22

22. Create Xen CentOS 7 Image

This page is not based on the OpenStack Installation Guide.

1. Log on to the control node as root.
2. Download the CentOS 7 ISO, and upload it to glance:

```
# wget http://mirror.rackspace.com/CentOS/7.2.1511/isos/x86_64/CentOS-7-x86_64-
˓NetInstall-1511.iso
# source admin-openrc.sh
# glance image-create --name "CentOS 7 ISO" --file CentOS-7-x86_64-NetInstall-
˓1511.iso --disk-format iso --container-format bare --visibility public --
˓progress
```

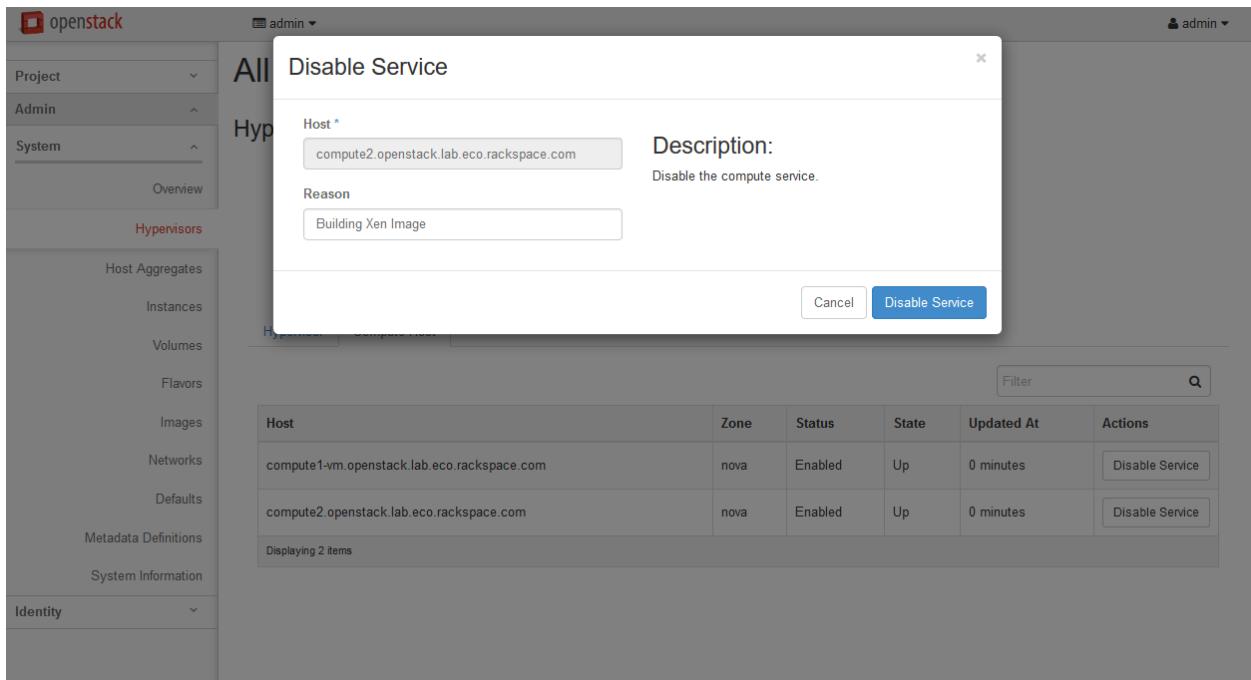
3. From a web browser, access http://*CONTROLLER_ADDRESS*/dashboard
4. Log in using the admin credentials.
5. In the left-hand menu, under “Admin”, and then “System”, click on “Hypervisors”:

Hostname	Type	VCPUs (used)	VCPUs (total)	RAM (used)	RAM (total)	Local Storage (used)	Local Storage (total)	Instances
compute1.openstack.lab.eco.rackspace.com	xen	0	48	512MB	128GB	0Bytes	1.8TB	0
compute2.openstack.lab.eco.rackspace.com	QEMU	0	48	512MB	125.7GB	0Bytes	1.8TB	0

6. Click on the “Compute Host” tab:

Host	Zone	Status	State	Updated At	Actions
compute1-vm.openstack.lab.eco.rackspace.com	nova	Enabled	Up	0 minutes	<button>Disable Service</button>
compute2.openstack.lab.eco.rackspace.com	nova	Enabled	Up	0 minutes	<button>Disable Service</button>

7. Next to “compute2”, click on “Disable Service”.
8. Enter a reason of “Building Xen image”, and click “Disable Service”:



9. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”. Click on “Launch Instance”.
10. Give the instance the name “centos7–xen-build”, use the flavor m1.small (for a 20GB disk), and select “Boot from image” and the “CentOS 7 ISO” image. Launch the instance:

Launch Instance

Details * Access & Security Networking * Post-Creation Advanced Options

Availability Zone

nova

Instance Name *

centos7-xen-build

Flavor * ?

m1.small

Instance Count * ?

1

Instance Boot Source * ?

Boot from image

Image Name *

CentOS 7 ISO (376.0 MB)

Flavor Details

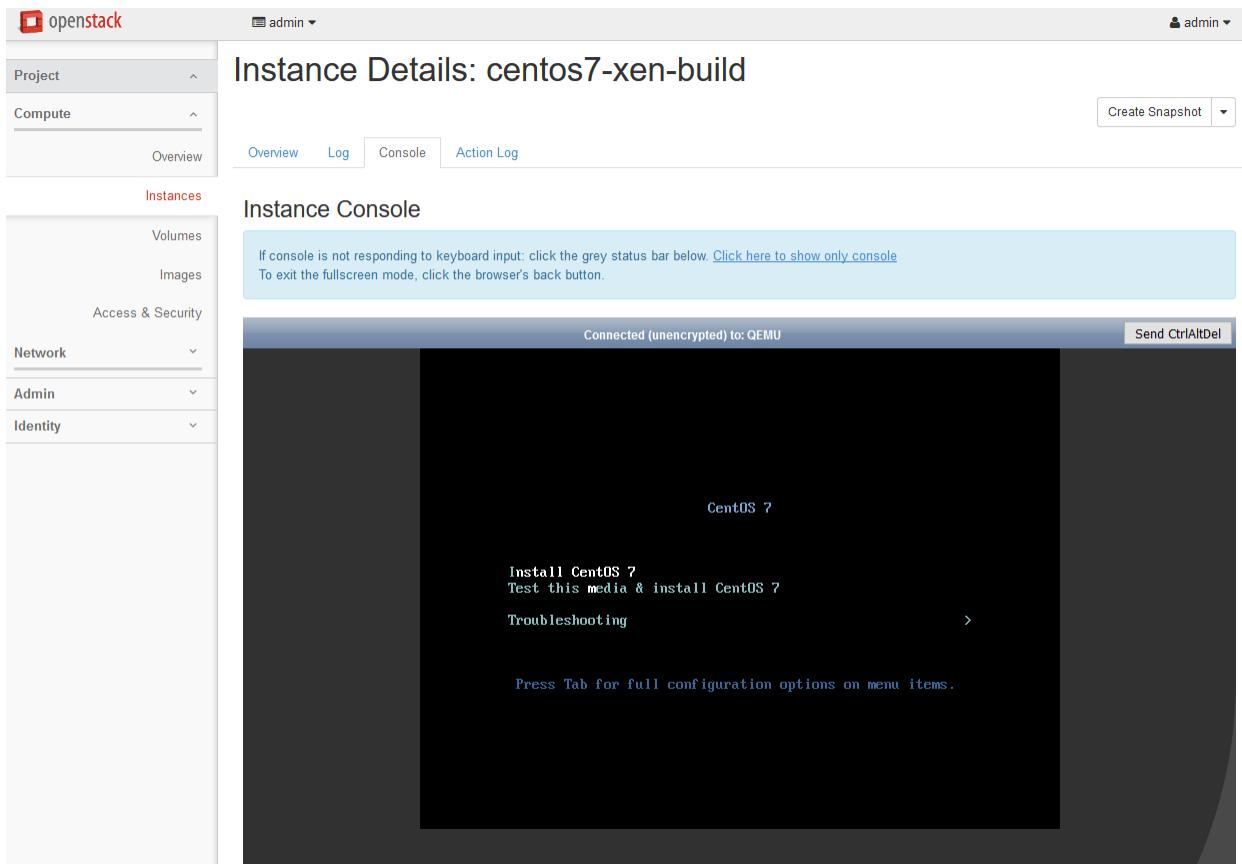
Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

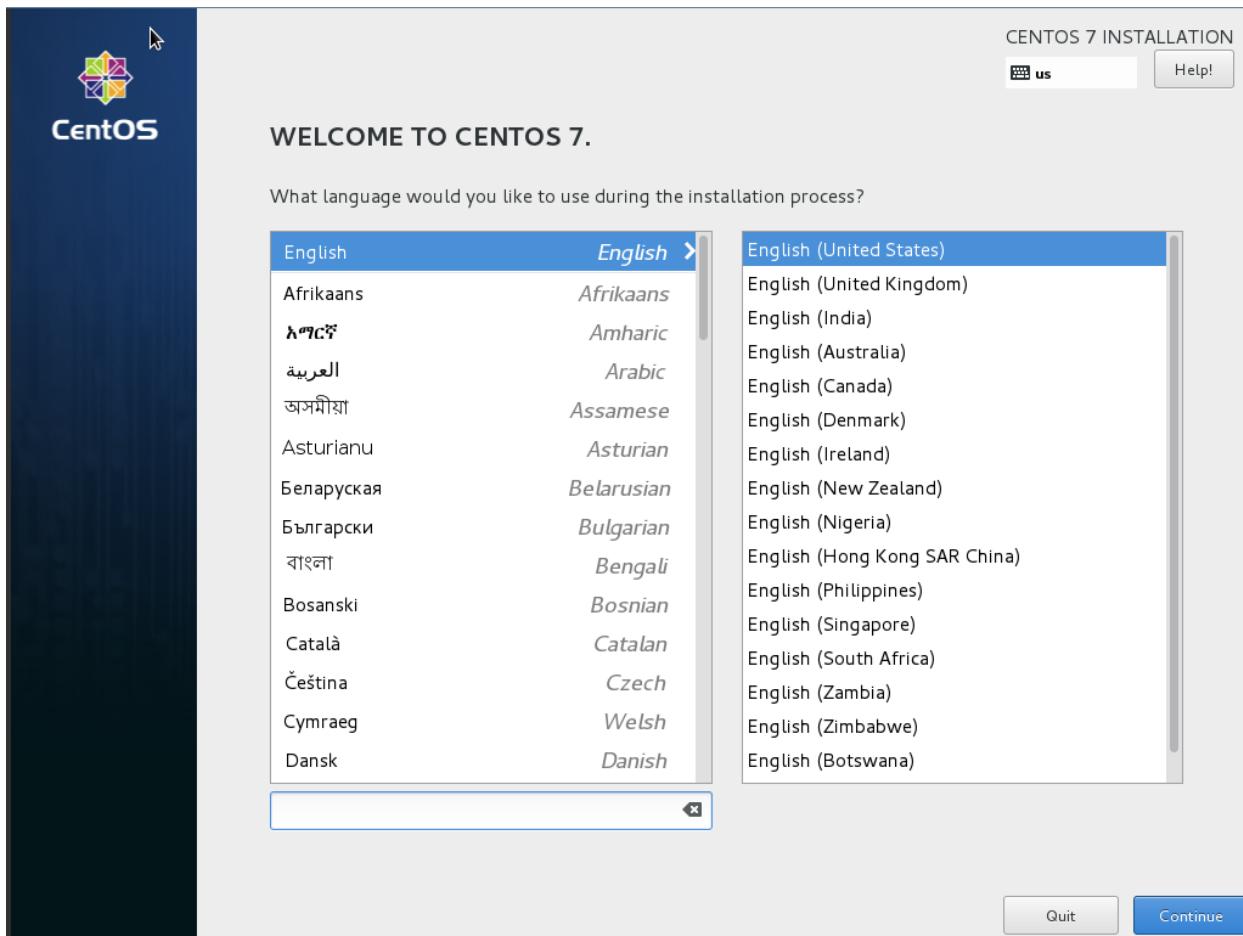
Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

Cancel **Launch**

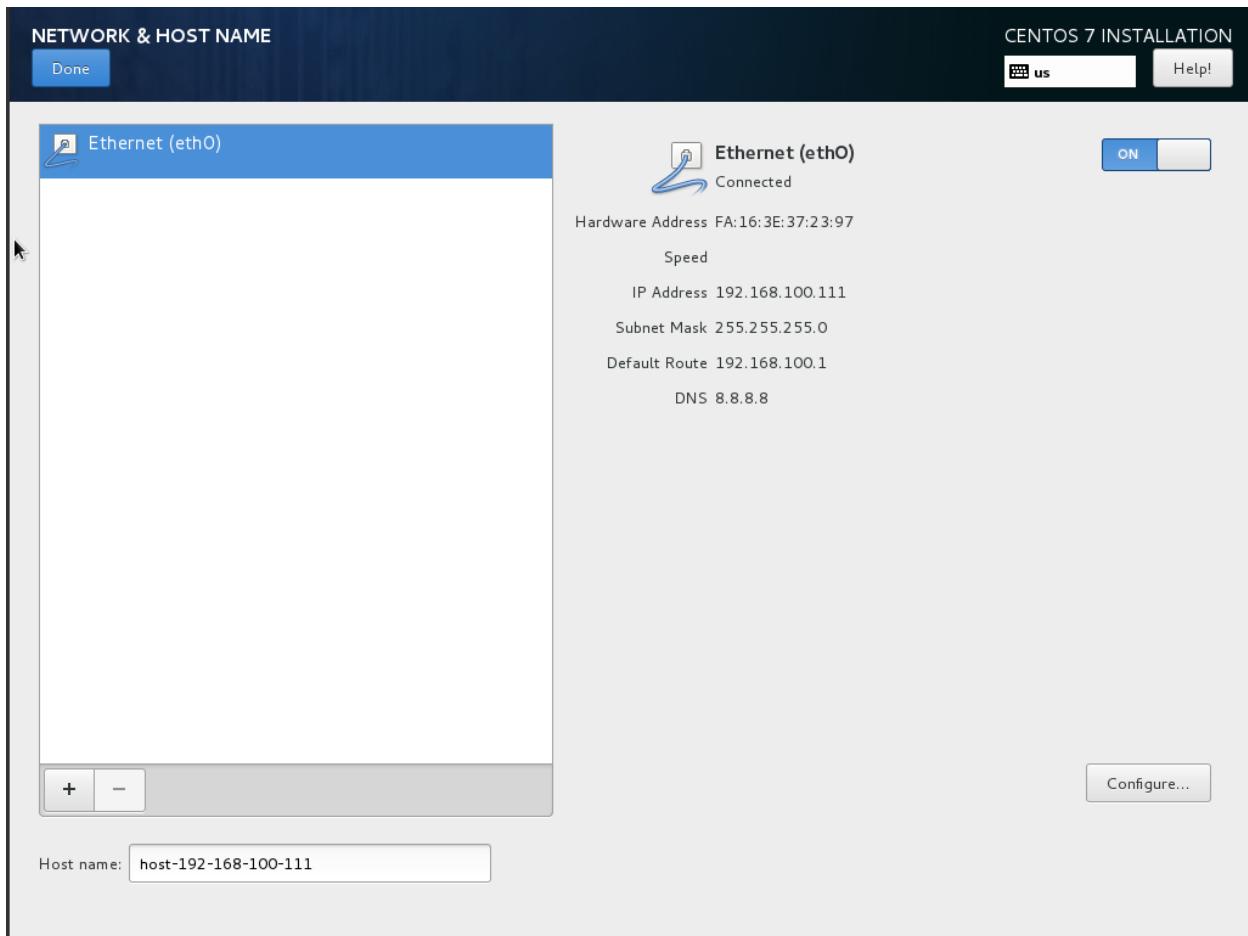
11. Wait for the instance to enter “Active” state. Then click on the instance. Click on the “Console” tab, and then click on the grey “Connected (unencrypted) to: QEMU” bar so that keyboard input will be directed to the console:



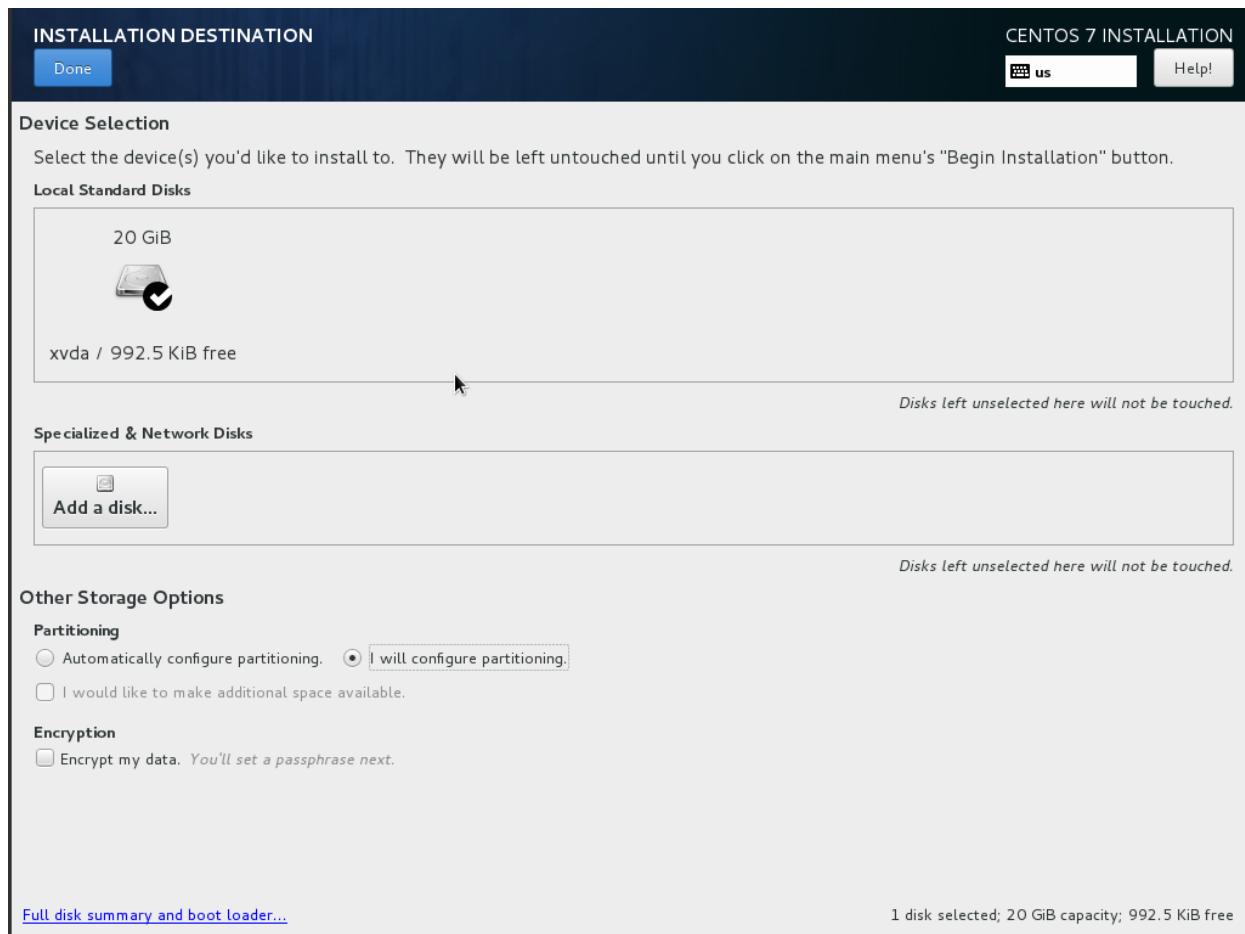
12. Highlight “Install CentOS 7”, and press Enter. Wait for the installer to start:



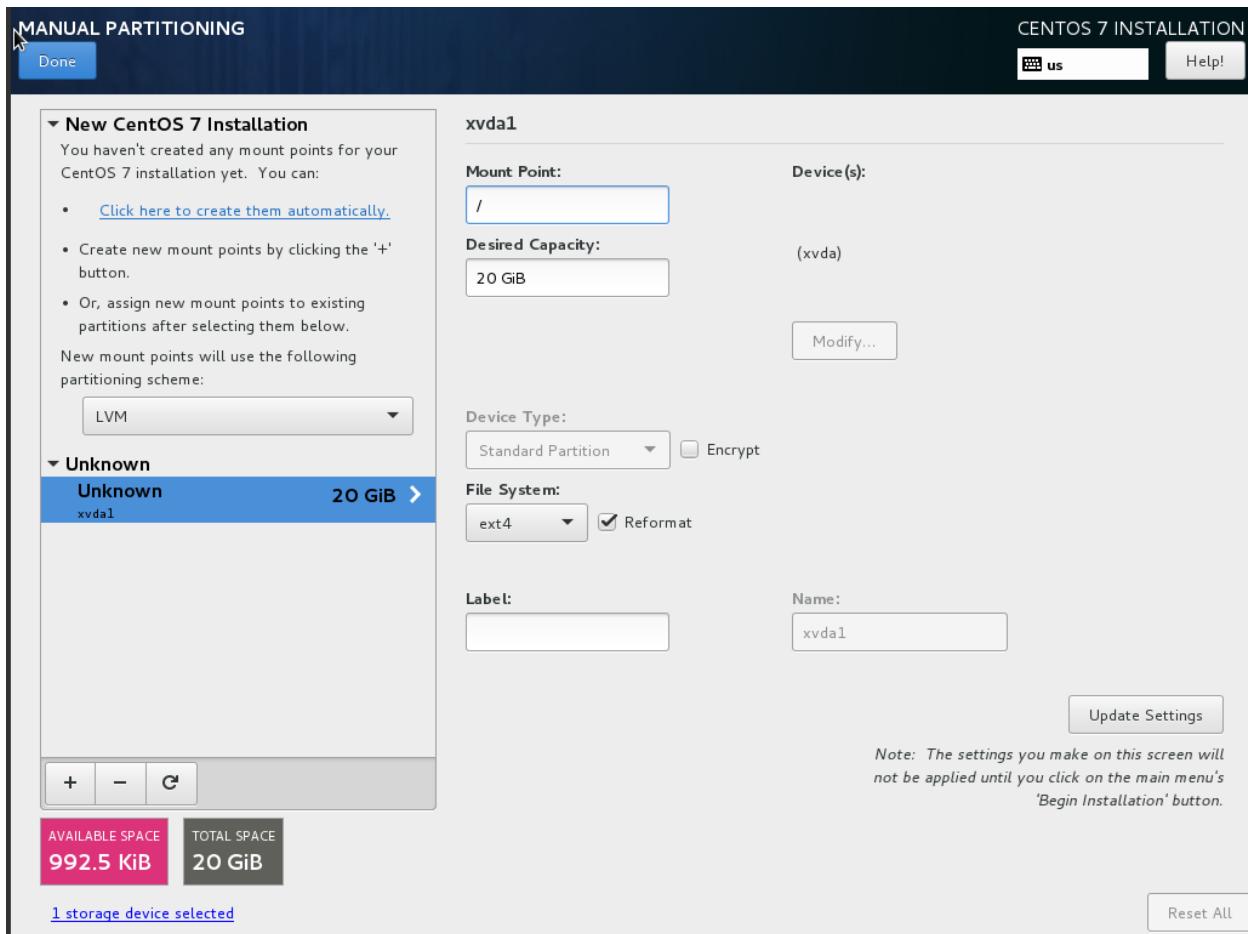
13. Set language and timezone.
14. Click on “Network & Hostname”. Enable the network interface by setting the switch to “On”:



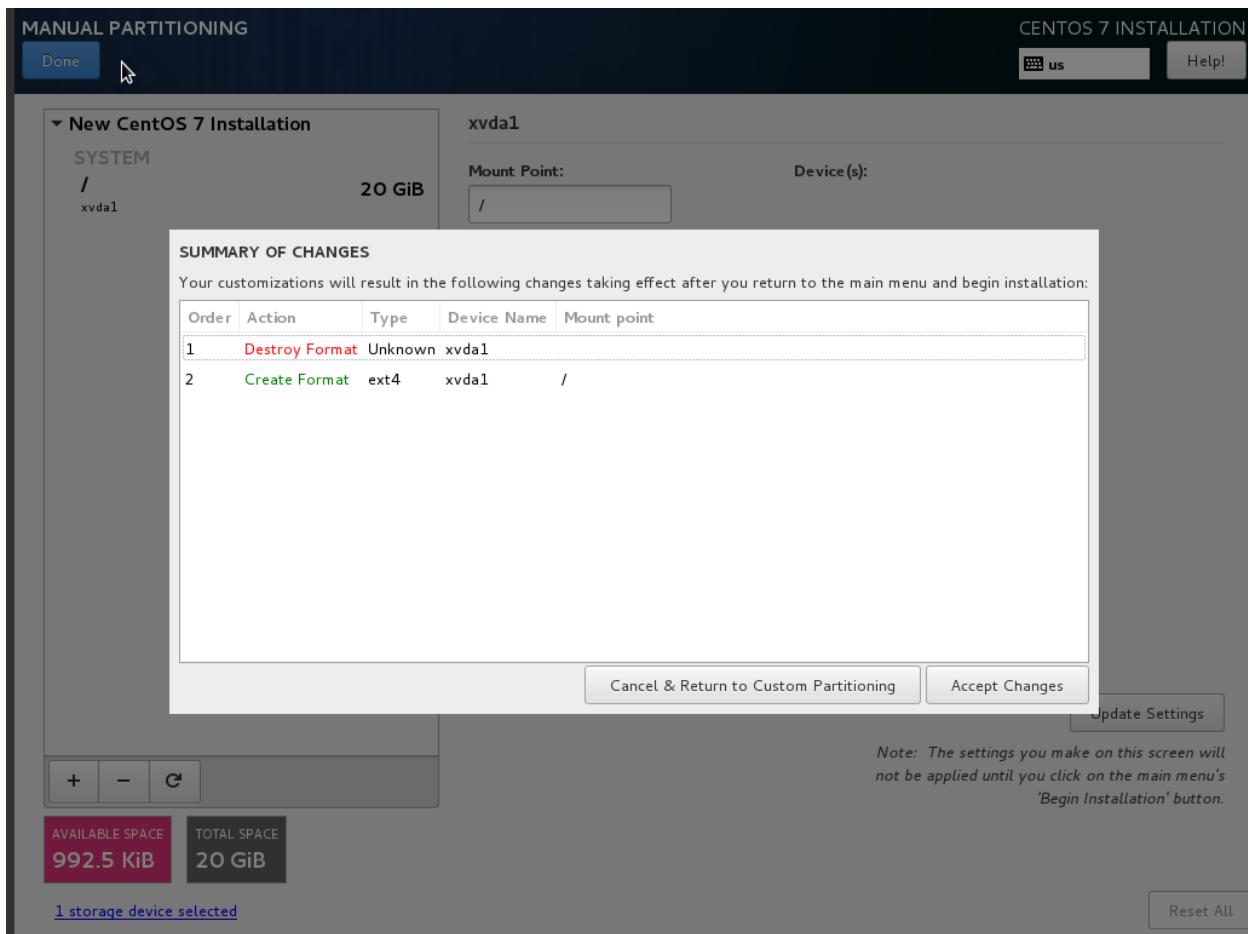
15. Click on “Installation Source”. Set the source to network, and then define a known-good mirror. You can use http://mirror.rackspace.com/CentOS/7.2.1511/os/x86_64/.
16. Click on “Installation Destination”. Select “I will configure partitioning” and click on “Done”:



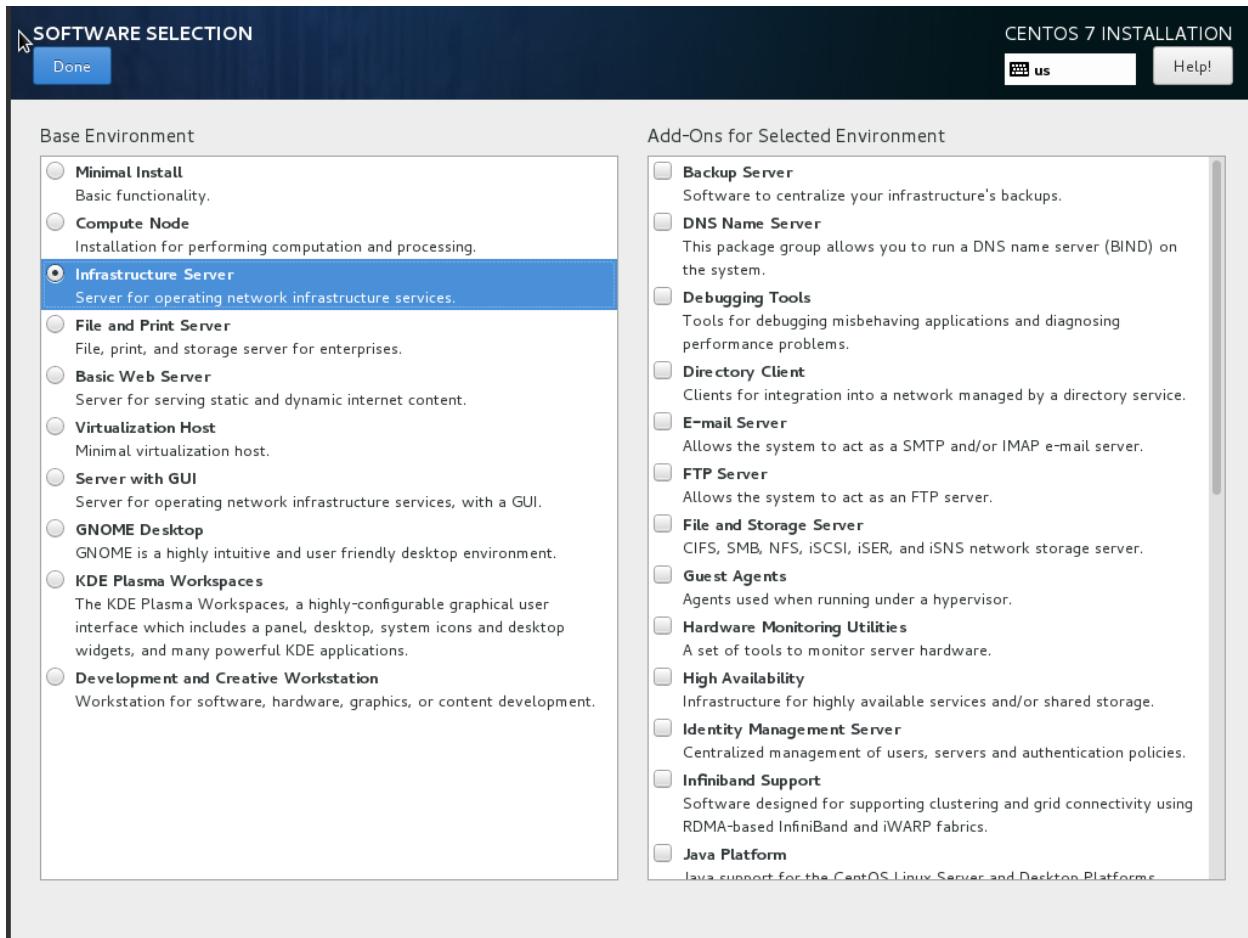
17. Click the arrow next to the word “Unknown” to expand that section and display the partition. Select “Reformat”, set the file system to “ext4”, and set the mount point to “/”. Click Done:



18. A yellow warning bar will appear. Click “Done” again, and then click on “Accept Changes”.



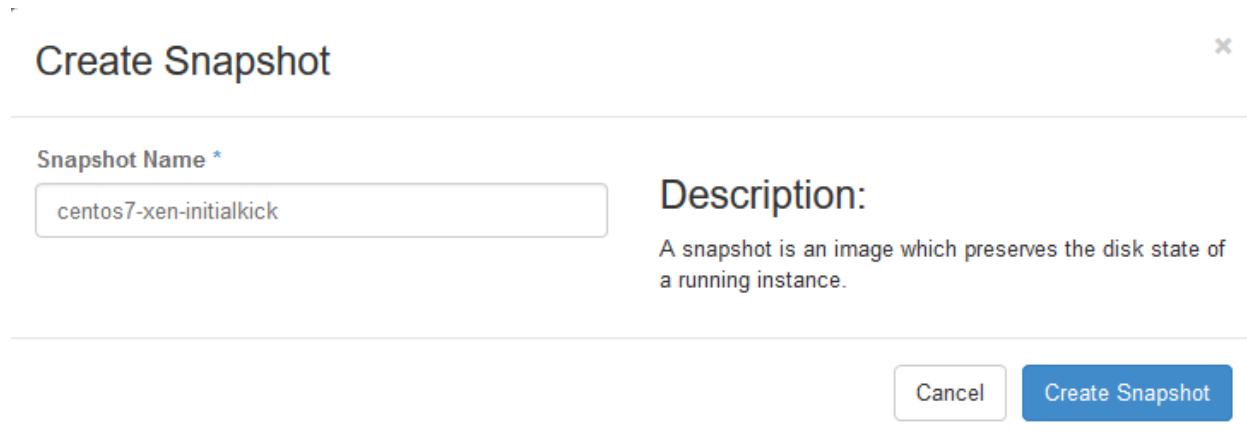
19. Click on “Software Selection”. Select “Infrastructure Server”, and click “Done”.



20. Click “Begin Installation”. Click on “Root Password” and set a good password.
21. Once installation is complete, click “Reboot”.
22. When reboot completes, your connection to the console will likely die. Refresh the page, click on the “Console” tab again, and then click on the grey banner again.
23. The server will be attempting to boot from the ISO once more. Press any key to stop the countdown.
24. In the top-right of the page, click the “Create Snapshot” button:

The screenshot shows the 'Instance Details' page for an instance named 'centos7-xen-build'. The 'Console' tab is selected. At the top right, there is a 'Create Snapshot' button. The main area displays the 'Instance Console' with instructions for keyboard input and exiting fullscreen mode. Below this is a large terminal window showing a black screen, indicating it is connected to QEMU.

25. Call the image “centos7-xen-initialkick” and click on “Create Snapshot”:



26. Horizon will show the “Images” page. Wait until “centos7-xen-initialkick” reaches “Active” status, and then click on the image.
27. In the top-right drop-down, click on “Update Metadata”.
28. On the left-hand side, in the “custom” box, enter “vm_mode” and click on the + button.
29. On the right-hand side, in the “vm_mode” box, enter “hvm”.
30. On the left-hand side, in the “custom” box, enter “hypervisor_type” and click on the + button.
31. On the right-hand side, in the “hypervisor_type” box, enter “xen”, and click on the “Save” button:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

The screenshot shows a user interface for managing image metadata. On the left, under 'Available Metadata', there is a 'Custom' section with a '+' button and a message 'No available metadata'. On the right, under 'Existing Metadata', several metadata items are listed: 'auto_disk_config' (True), 'base_image_ref' (d3cd60e2-753d-), 'hypervisor_type' (xen) which is highlighted with a blue background, 'image_type' (snapshot), 'instance_uuid' (25b6d5dc-da58-), 'os_type' (linux), 'user_id' (31766cbe74d54), and 'vm_mode' (hvm). At the bottom, a tooltip for 'hypervisor_type' is displayed, and at the very bottom are 'Cancel' and 'Save' buttons.

Available Metadata	Existing Metadata
Custom	auto_disk_config True
No available metadata	base_image_ref d3cd60e2-753d-
	hypervisor_type xen
	image_type snapshot
	instance_uuid 25b6d5dc-da58-
	os_type linux
	user_id 31766cbe74d54
	vm_mode hvm

32. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”.
33. Highlight the “centos7-xen-build” instance, and click on “Terminate Instances”.

The screenshot shows the 'Instances' page of the openstack-xenserver interface. The left sidebar has 'Compute' selected under 'Project'. The main area shows a table with one row for 'centos7-xen-build'. The columns include Instance Name, Image Name, IP Address, Size, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. The 'Actions' column for this instance contains a 'Terminate Instances' button.

	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
<input checked="" type="checkbox"/>	centos7-xen-build	CentOS 7 ISO	192.168.100.72	m1.small	-	Active	nova	None	Running	21 minutes	<button>Create Snapshot</button>

34. Click “Terminate Instance” again to confirm:

Confirm Terminate Instances

You have selected "centos7-xen-build". Please confirm your selection. Terminated instances are not recoverable.

Cancel Terminate Instances

35. Click on “Launch Instance”. Give the instance the name “centos7-xen-build”, use the flavor m1.small (for a 20GB disk), and select “Boot from image” and the “centos7-xen-initialkick” image. Launch the instance:

Launch Instance

Details * Access & Security Networking * Post-Creation Advanced Options

Availability Zone
nova

Instance Name *
centos7-xen-build

Flavor * ?
m1.small
Some flavors not meeting minimum image requirements have been disabled.

Instance Count * ?
1

Instance Boot Source * ?
Boot from image

Image Name *
centos7-xen-initialkick (476.7 MB)

Flavor Details

Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

Cancel **Launch**

36. Wait for the instance to enter “Active” state. SSH to the new instance as “root”, using the root password used during setup.
37. Delete the static hostname file:

```
# rm /etc/hostname
```

38. Stop and disable the firewalld service:

```
# systemctl disable firewalld.service
# systemctl stop firewalld.service
```

39. Disable SELINUX:

```
# setenforce 0
# vim /etc/sysconfig/selinux
```

```
SELINUX=permissive
```

40. Update all packages on the server:

```
# yum update
```

41. Download and install the XenServer tools:

```
# wget http://boot.rackspace.com/files/xentools/xs-tools-6.5.0-20200.iso
# mkdir /mnt/cdrom
# mount -o loop xs-tools-6.5.0-20200.iso /mnt/cdrom
# cd /mnt/cdrom/Linux
# rpm -Uvh xe-guest-utilities-xenstore-6.5.0-1427.x86_64.rpm xe-guest-utilities-6.
# cd ~
# umount /mnt/cdrom
# rm xs-tools-6.5.0-20200.iso
```

42. Reboot the instance:

```
# systemctl reboot
```

43. Wait for the server to reboot, and then log back in as root.

44. Install the nova-agent:

```
# rpm -Uvh https://github.com/rackerlabs/openstack-guest-agents-unix/releases/
# download/1.39.1/nova-agent-1.39-1.x86_64.rpm
```

45. Create a CentOS 7.2-compatible systemd unit file for the nova-agent service:

```
# vim /usr/lib/systemd/system/nova-agent.service

[Unit]
Description=nova-agent service
After=xe-linux-distribution.service

[Service]
EnvironmentFile=/etc/nova-agent.env
ExecStart=/usr/sbin/nova-agent -n -l info /usr/share/nova-agent/nova-agent.py

[Install]
WantedBy=multi-user.target
```

46. Create a python environment file for the nova-agent service:

```
# vim /etc/nova-agent.env

LD_LIBRARY_PATH="${LD_LIBRARY_PATH}:/usr/share/nova-agent/1.39.1/lib"
PYTHONPATH="${PYTHONPATH}:/usr/share/nova-agent/1.39.1/lib/python2.6/site-
# packages:/usr/share/nova-agent/1.39.1/lib/python2.6/"
```

47. Reload systemd to import the new unit file:

```
# systemctl daemon-reload
```

48. Enable and start the nova-agent service:

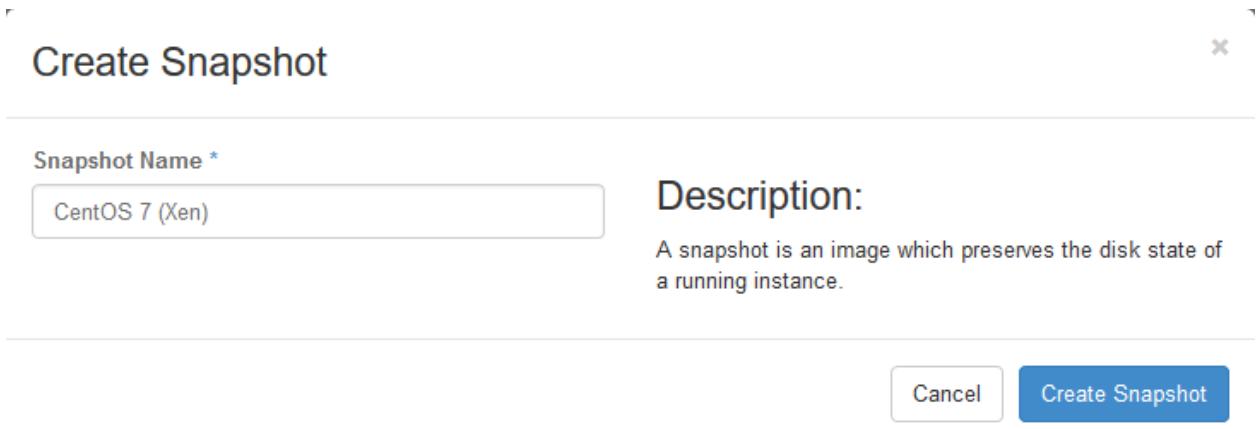
```
# systemctl enable nova-agent.service  
# systemctl start nova-agent.service
```

49. Remove the static network configuration file:

```
# rm /etc/sysconfig/network-scripts/ifcfg-eth0
```

50. Clear the root bash history:

```
# rm /root/.bash_history; history -c
```

51. In horizon, click the “Create Snapshot” button next to the Instance. Name the image “CentOS 7 (Xen)”: 

52. Wait for the image to go to “Active” state and then, from the drop-down box next to the image, click on “Update Metadata”.
53. On the left-hand side, in the “Custom” box, enter “xenapi_use_agent”, and then click the + button.
54. On the right-hand side, in the “xenapi_use_agent”, enter “true” and then click the Save button:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

Available Metadata

Filter

Custom	
--------	--

No available metadata

Existing Metadata

Filter

base_image_ref	d3cd60e2-753d-	
hypervisor_type	xen	
image_type	snapshot	
instance_uuid	87f0c0e7-799c-4	
os_type	linux	
user_id	31766cbe74d54	
vm_mode	hvm	
xenapi_use_agent	true	

xenapi_use_agent (xenapi_use_agent)

Cancel
 Save

55. In the drop-down box next to the image, click on “Edit Image”.
56. Check the “public” and “protected” boxes, and click on “Update Image”:

138

Chapter 22. 22. Create Xen CentOS 7 Image

Update Image

Name *
CentOS 7 (Xen)

Description:
Edit the image details.

Description

Kernel ID

Ramdisk ID

Architecture

Format *
VHD - Virtual Hard Disk

Minimum Disk (GB) ?
20

Minimum RAM (MB) ?
0

Public

Protected

57. Select the “centos7-xen-initialkick” image, and click on “Delete Images”. Click “Delete Images” to confirm:

The screenshot shows the 'Images' section of the openstack-xenserver interface. The left sidebar has 'Project' selected under 'Compute'. The main area shows a table of images. The second row, 'CentOS 7 (Xen)', is unselected. The third row, 'centos7-xen-initialkick', has a checked checkbox in the first column and is selected for deletion. The top navigation bar shows 'admin' and the top right has a 'Delete Images' button. The table columns are: Image Name, Type, Status, Public, Protected, Format, Size, and Actions.

	Image Name	Type	Status	Public	Protected	Format	Size	Actions
<input type="checkbox"/>	CentOS 7 (Xen)	Snapshot	Active	No	No	VHD	843.8 MB	<input type="button" value="Launch Instance"/>
<input checked="" type="checkbox"/>	centos7-xen-initialkick	Snapshot	Active	No	No	VHD	476.7 MB	<input type="button" value="Launch Instance"/>

58. In the left-hand menu, under “Project” and then “Compute”, click on “Instances”.
59. Highlight the “centos7-xen-build” instance, and click on “Terminate Instances”. Click “Terminate Instances” to confirm:

Confirm Terminate Instances

You have selected "centos7-xen-build". Please confirm your selection. Terminated instances are not recoverable.

[Cancel](#) [Terminate Instances](#)

60. In the left-hand menu, under “Admin” and then “System” click on “Hypervisors”. Next to “compute2”, click on “Enable Service”.

CHAPTER 23

23. Launch test Xen CentOS 7 Instance

This page is not based on the OpenStack Installation Guide.

1. From a web browser, access http://*CONTROLLER ADDRESS*/dashboard.
2. Log in using the demo credentials.
3. In the left-hand menu, under “Project”, and then “Compute”, click on “Access & Security”. Click on the “Key Pairs” tab:

The screenshot shows the OpenStack dashboard interface. The top navigation bar includes the OpenStack logo, user info (demo), and a search bar. The left sidebar has dropdown menus for Project (selected), Compute (selected), Network, and Identity. Under Compute, it shows Overview, Instances, Volumes, and Images. Below the sidebar, a red link labeled "Access & Security" is visible. The main content area is titled "Access & Security" and contains tabs for "Security Groups", "Key Pairs" (which is the active tab, indicated by a blue background), and "API Access". A sub-header "Key Pair Name" is followed by a table with columns "Key Pair Name", "Fingerprint", and "Actions". The table body displays the message "No items to display." and "Displaying 0 items". At the bottom right of the table are buttons for "Create Key Pair" and "Import Key Pair".

4. If you have an SSH keypair already available which you would like to use, click on “Import Key Pair”. Give the key a name and then paste in your public key:

Import Key Pair

Key Pair Name *

Public Key *

Description:

Key Pairs are how you login to your instance after it is launched.

Choose a key pair name you will recognise and paste your SSH public key into the space provided.

SSH key pairs can be generated with the ssh-keygen command:

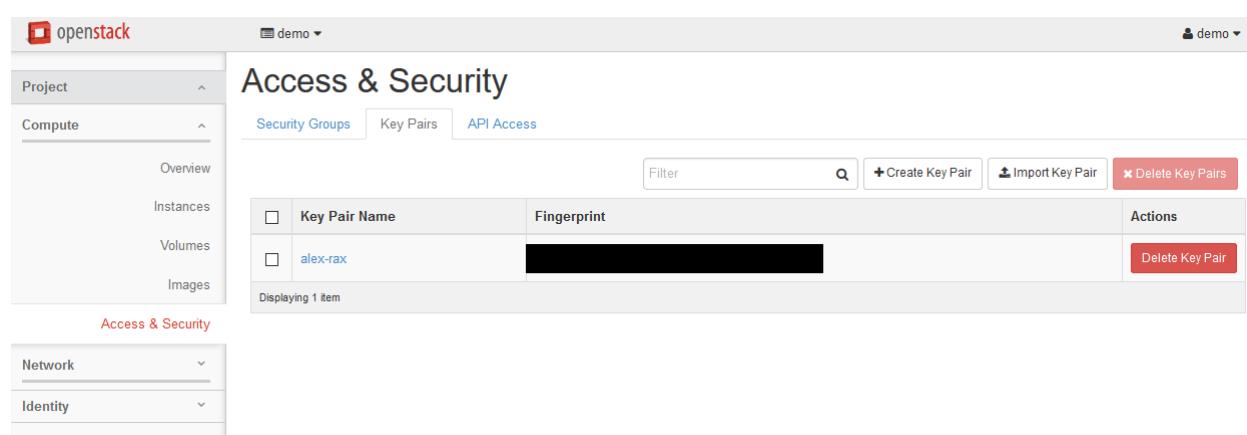
```
ssh-keygen -t rsa -f cloud.key
```

This generates a pair of keys: a key you keep private (cloud.key) and a public key (cloud.key.pub). Paste the contents of the public key file here.

After launching an instance, you login using the private key (the username might be different depending on the image you launched):

```
ssh -i cloud.key <username>@<instance_ip>
```

Cancel
Import Key Pair



	Key Pair Name	Fingerprint	Actions
<input type="checkbox"/>	alex-rax		Delete Key Pair

5. Alternatively, if you would like to create a new pair, click on “Create Key Pair. Give the key a name and click on “Create Key Pair. Download the key for use in your SSH client:

Create Key Pair

Key Pair Name *

Description:

Key pairs are ssh credentials which are injected into images when they are launched. Creating a new key pair registers the public key and downloads the private key (a .pem file).

Protect and use the key as you would any normal ssh private key.

Cancel Create Key Pair

openstack demo demo

Project Compute Overview Instances Volumes Images

Access & Security Network Identity

Download Key Pair

The key pair "created" should download automatically. If not use the link below.

[Download key pair "created"](#)

6. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”.
7. Click on “Launch Instance”. Name the instance “centos7-test”, select the “m1.small” flavor, and “boot from image”. Choose the “CentOS 7 (Xen)” image. Before clicking on “Launch”, click on the “Access & Security” tab:

Launch Instance

Details * Access & Security Networking * Post-Creation Advanced Options

Availability Zone

nova

Instance Name *

centos7-test

Flavor * ?

m1.small

Some flavors not meeting minimum image requirements have been disabled.

Instance Count * ?

1

Instance Boot Source * ?

Boot from image

Image Name *

CentOS 7 (Xen) (843.8 MB)

Specify the details for launching an instance.
The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

Cancel **Launch**

8. Ensure that the key pair you just created or imported is selected, and then click on Launch:

Launch Instance

Details * Access & Security **Networking *** Post-Creation Advanced Options

Key Pair ?

alex-rax ▼ +

Control access to your instance via key pairs, security groups, and other mechanisms.

Security Groups ?

default

Cancel Launch

9. Wait for the instance to go to “Active” state, and then SSH to the server as “root”, using the key pair you just created or imported.
10. When you are satisfied that the test instance is working, select it and then click on “Terminate Instances”. Click on “Terminate Instances” to confirm.

openstack demo demo ▾

Project ▾ Compute ▾ Overview Instances Volumes Images Access & Security Network ▾ Identity ▾

Instances

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
<input checked="" type="checkbox"/> centos7-test	CentOS 7 (Xen)	192.168.100.77	m1.small	alex-rax	Active	nova	None	Running	4 minutes	Create Snapshot ▾

Displaying 1 item

CHAPTER 24

24. Create KVM CentOS 7 Image

This page is not based on the OpenStack Installation Guide.

1. From a web browser, access `http://*CONTROLLER_ADDRESS*/dashboard`.
2. Log in using the admin credentials.
3. In the left-hand menu, under “Admin”, and then “System”, click on “Hypervisors”:

The screenshot shows the OpenStack dashboard with the "Hypervisor" tab selected. The main area displays "Hypervisor Summary" with three circular icons representing VCPU Usage (Used 0 of 96), Memory Usage (Used 1GB of 253.7GB), and Local Disk Usage (Used 0Bytes of 3.6TB). Below this, a table lists two compute hosts:

Hostname	Type	VCPUs (used)	VCPUs (total)	RAM (used)	RAM (total)	Local Storage (used)	Local Storage (total)	Instances
compute1.openstack.lab.eco.rackspace.com	xen	0	48	512MB	128GB	0Bytes	1.8TB	0
compute2.openstack.lab.eco.rackspace.com	QEMU	0	48	512MB	125.7GB	0Bytes	1.8TB	0

A message at the bottom of the table says "Displaying 2 items".

4. Click on the “Compute Host” tab:

The screenshot shows the OpenStack Xenserver dashboard under the 'Hypervisors' tab. On the left, a sidebar lists various categories: Project, Admin, System, Overview, Hypervisors (selected), Host Aggregates, Instances, Volumes, Flavors, Images, Networks, Defaults, Metadata Definitions, System Information, and Identity. The main content area is titled 'All Hypervisors' and 'Hypervisor Summary'. It displays three large circular icons representing VCPU Usage (Used 0 of 96), Memory Usage (Used 1GB of 253.7GB), and Local Disk Usage (Used 0Bytes of 3.6TB). Below this is a table listing two hosts:

Host	Zone	Status	State	Updated At	Actions
compute1-vm.openstack.lab.eco.rackspace.com	nova	Enabled	Up	0 minutes	<button>Disable Service</button>
compute2.openstack.lab.eco.rackspace.com	nova	Enabled	Up	0 minutes	<button>Disable Service</button>

At the bottom of the table, it says 'Displaying 2 items'.

5. Next to “compute1-vm”, click on “Disable Service”.
6. Enter a reason of “Building KVM image”, and click “Disable Service”:

Disable Service

Host *

Description:

Disable the compute service.

Reason

Cancel Disable Service

7. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”. Click on “Launch Instance”.
8. Give the instance the name “centos7-kvm-build”, use the flavor m1.small (for a 20GB disk), and select “Boot from image” and the “CentOS 7 ISO” image. Launch the instance:

Launch Instance

Details * Access & Security Networking * Post-Creation Advanced Options

Availability Zone

nova

Instance Name *

centos7-kvm-build

Flavor * ?

m1.small

Instance Count * ?

1

Instance Boot Source * ?

Boot from image

Image Name *

CentOS 7 ISO (376.0 MB)

Specify the details for launching an instance.

The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

Number of Instances	1 of 10 Used
Number of VCPUs	1 of 20 Used
Total RAM	2,048 of 51,200 MB Used

Cancel **Launch**

9. Wait for the instance to enter “Active” state. Then, in the left-hand menu, under “Project”, and then “Compute”, click on “Volumes”. Click on “Create Volume”.
10. Name the image “centos7-kvm-build”, and set the size to 20 GB. Click “Create Volume”:

Create Volume

Volume Name
centos7-kvm-build

Description

Volume Source
No source, empty volume

Type
No volume type

Size (GB) *
20

Availability Zone
nova

Description:
Volumes are block devices that can be attached to instances.

Volume Type Description:
If "No volume type" is selected, the volume will be created without a volume type.

Volume Limits

Total Gigabytes (0 GB) 1,000 GB Available

Number of Volumes (0) 10 Available

[Cancel](#) [Create Volume](#)

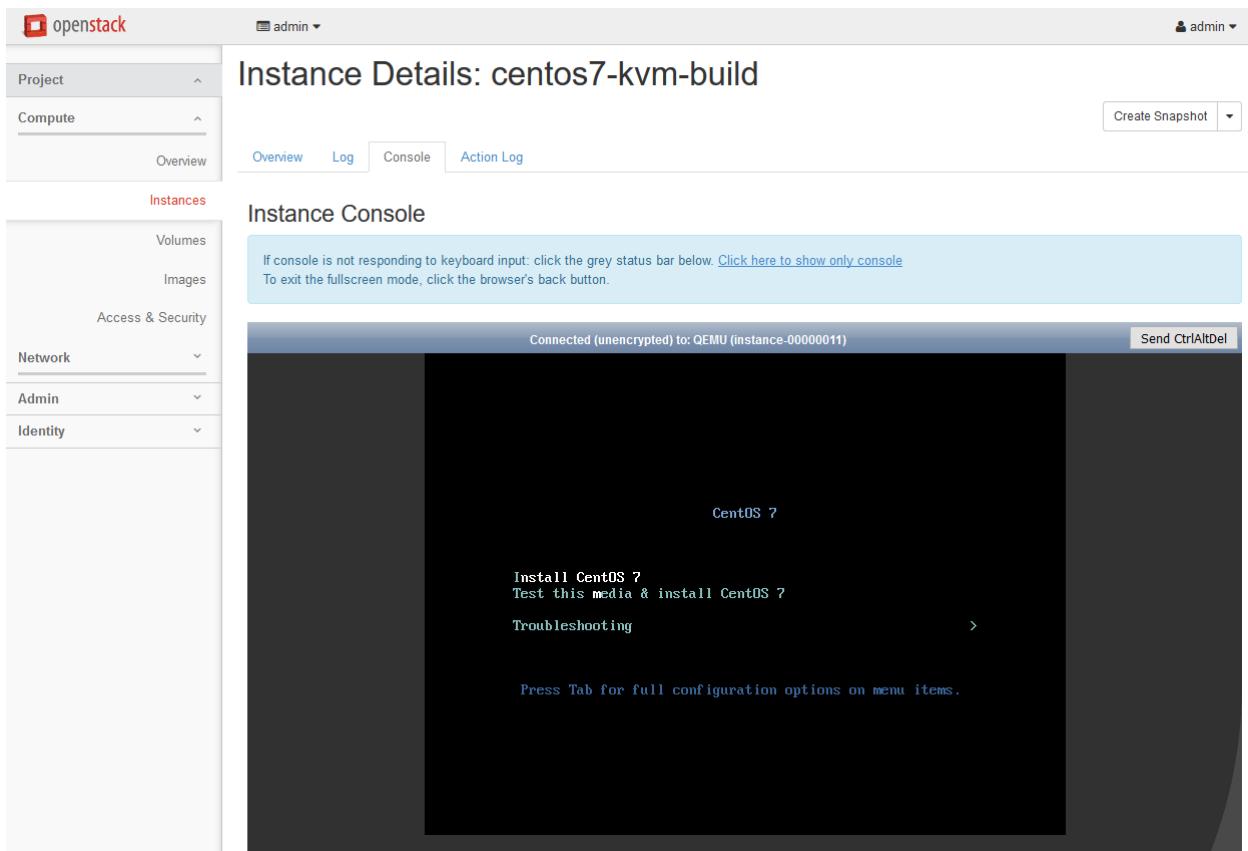
11. Once the volume enters “Available” status, click the “Actions” drop-down next to the volume, and select “Manage Attachments”.
12. Under “Attach to instance”, select “centos7-kvm-build”, and click “Attach Volume”:

The screenshot shows two interface components. At the top is a table titled 'Manage Volume Attachments' with columns for 'Instance', 'Device', and 'Actions'. A message 'No items to display.' is centered in the table. Below this is a section titled 'Attach To Instance' containing a dropdown menu with the option 'centos7-kvm-build (d5144aed-0052-4a05-b5da-fe5f939f9753)'. At the bottom right of this section are 'Cancel' and 'Attach Volume' buttons.

13. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”. Under the “Actions” drop-down for the “centos7-kvm-build” instance, click on “Hard Reboot Instance”. Click on “Hard Reboot Instance” to confirm:

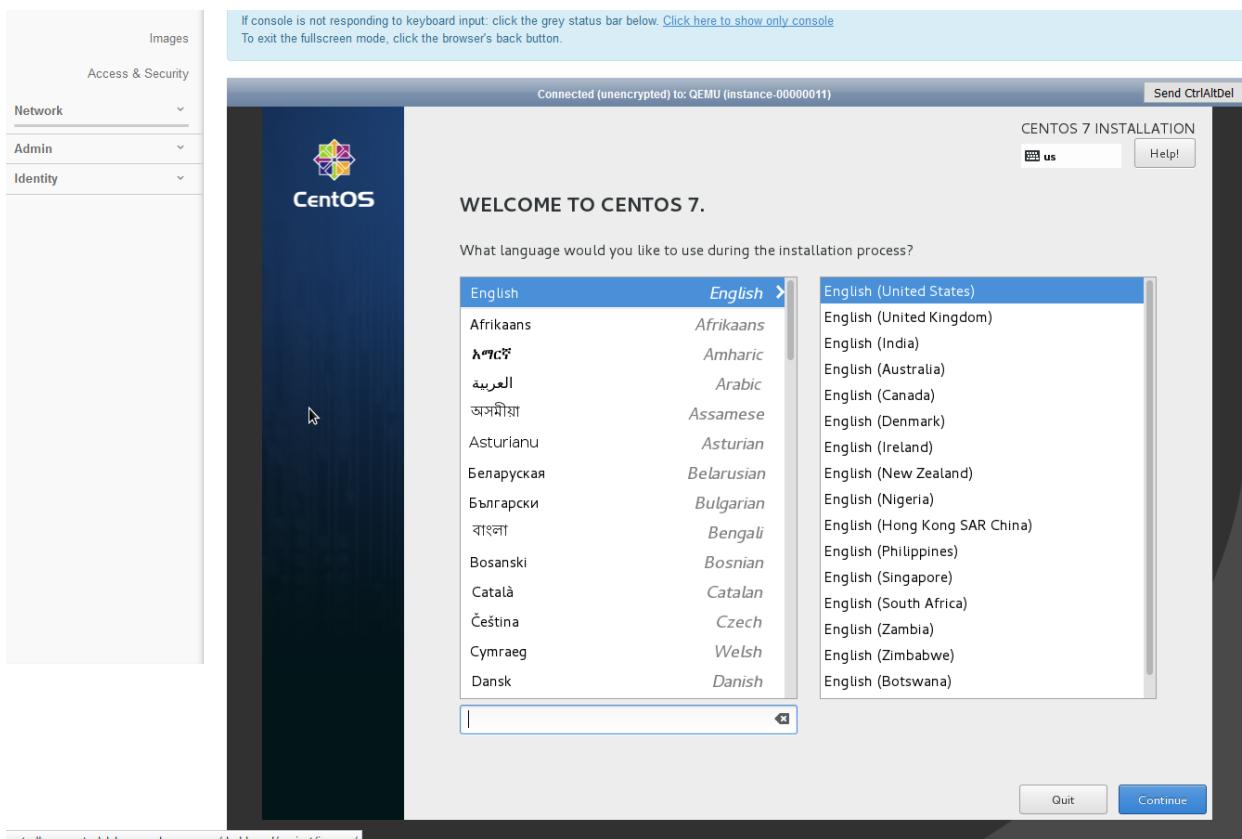
The screenshot shows a confirmation dialog titled 'Confirm Hard Reboot Instance'. It contains the text: "You have selected \"centos7-kvm-build\". Please confirm your selection. Restarted instances will lose any data not saved in persistent storage." At the bottom right are 'Cancel' and 'Hard Reboot Instance' buttons.

14. Wait for the instance to go back to “Active” state, and then click on the instance. Click on the “Console” tab, and then click on the grey “Connected (unencrypted) to: QEMU” bar so that keyboard input will be directed to the console:



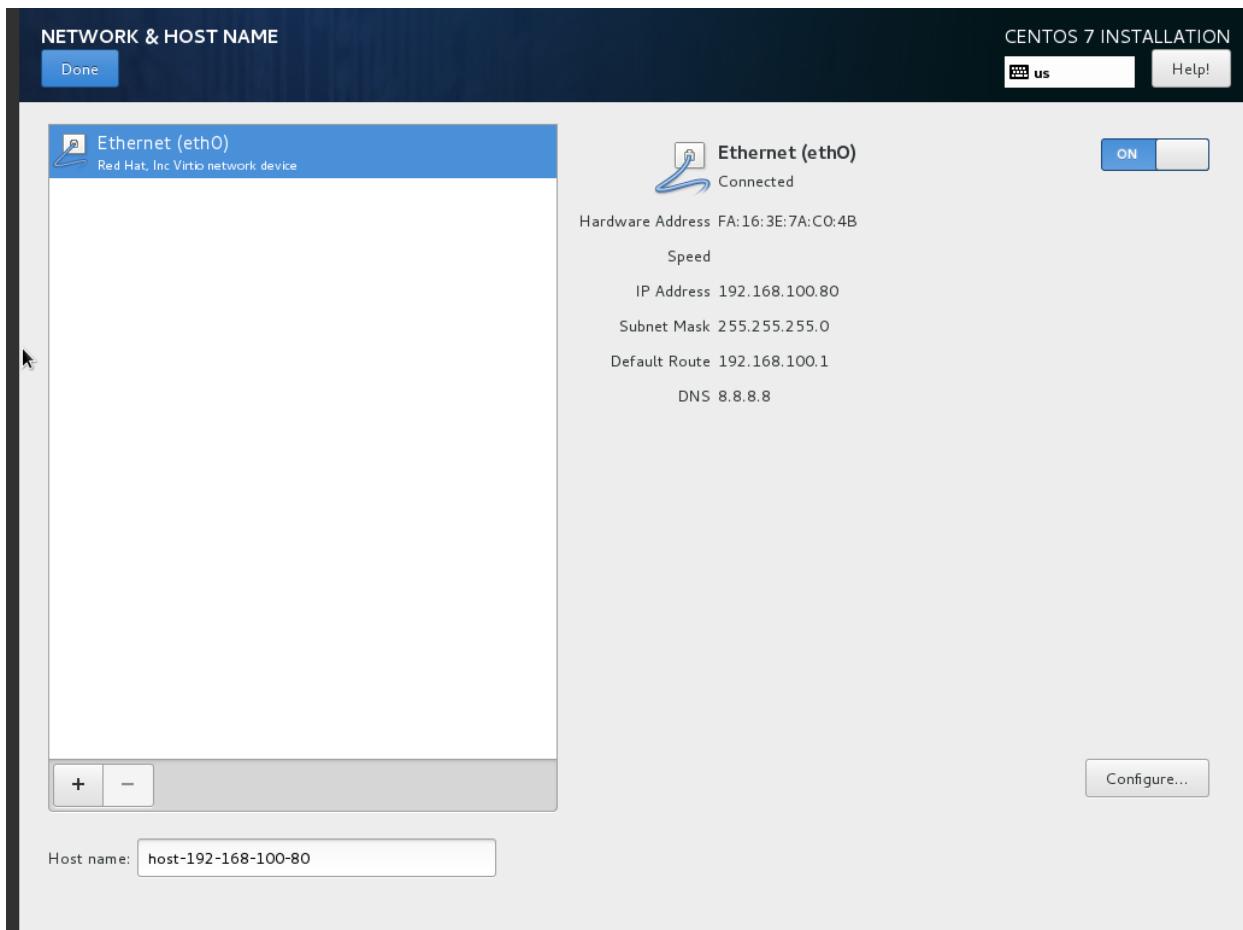
15. Highlight “Install CentOS 7”, and Enter.

16. Wait for the installer to boot:

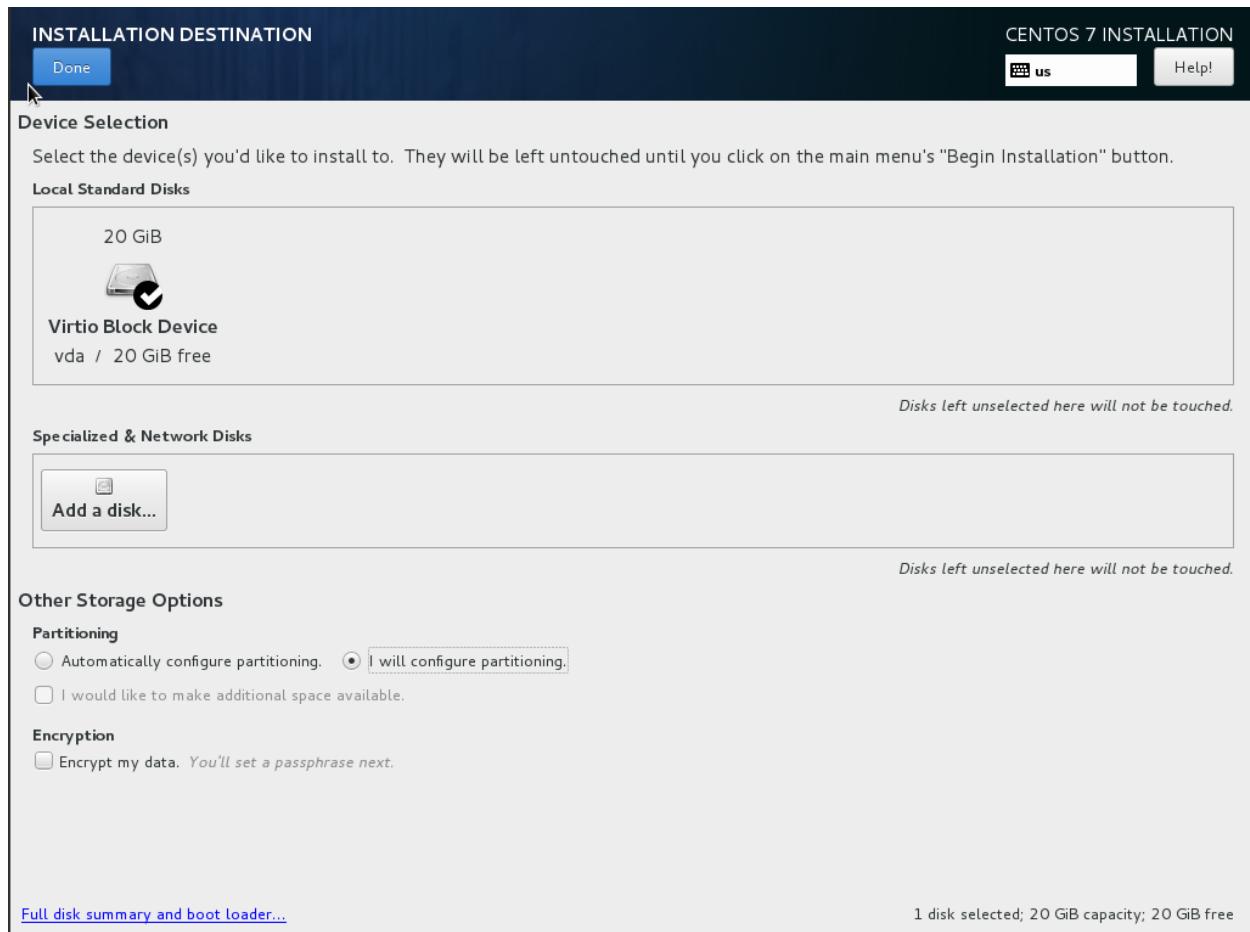


17. Select language and set the timezone.

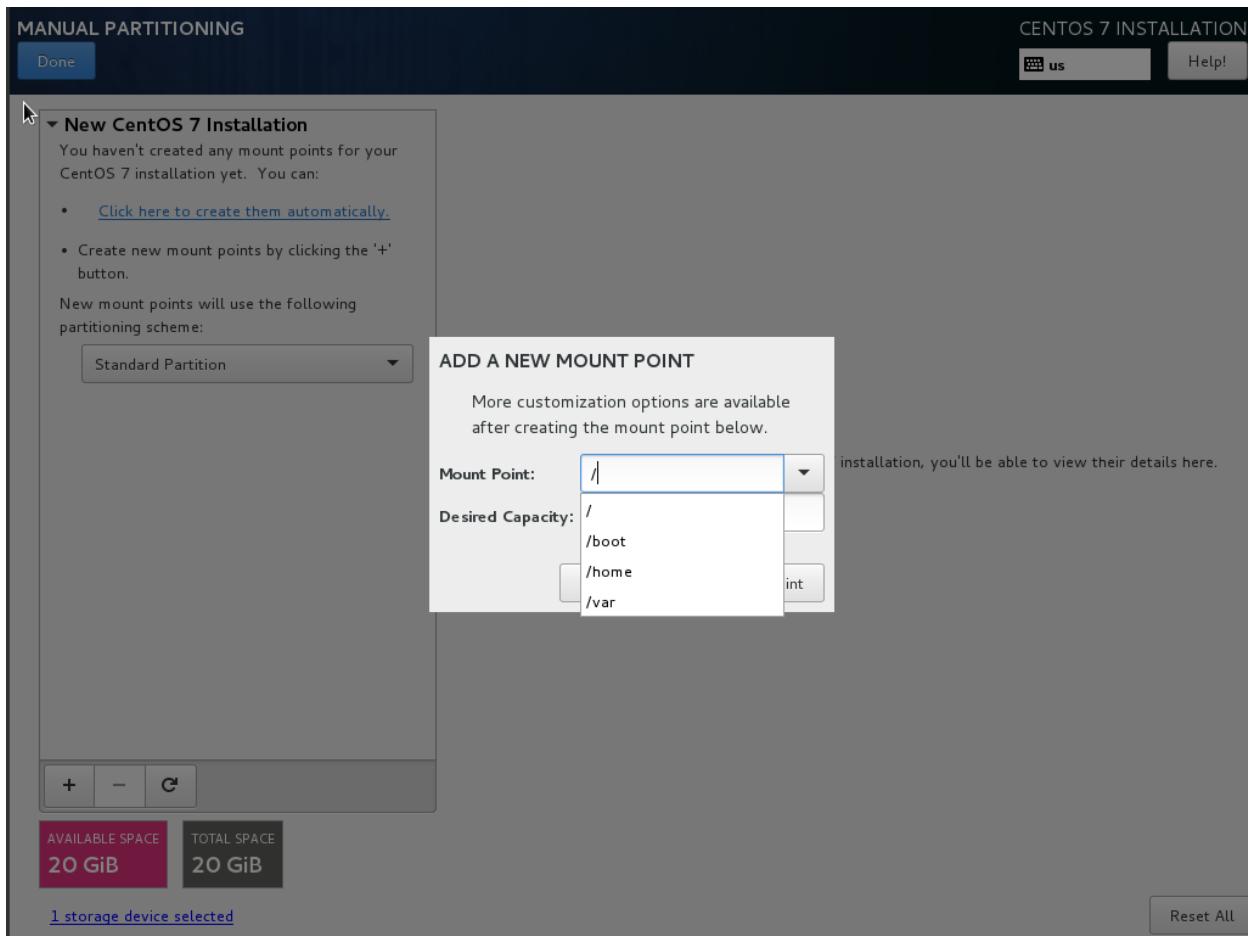
18. Click on “network & hostname” and activate the network interface by setting the switch to “On”:



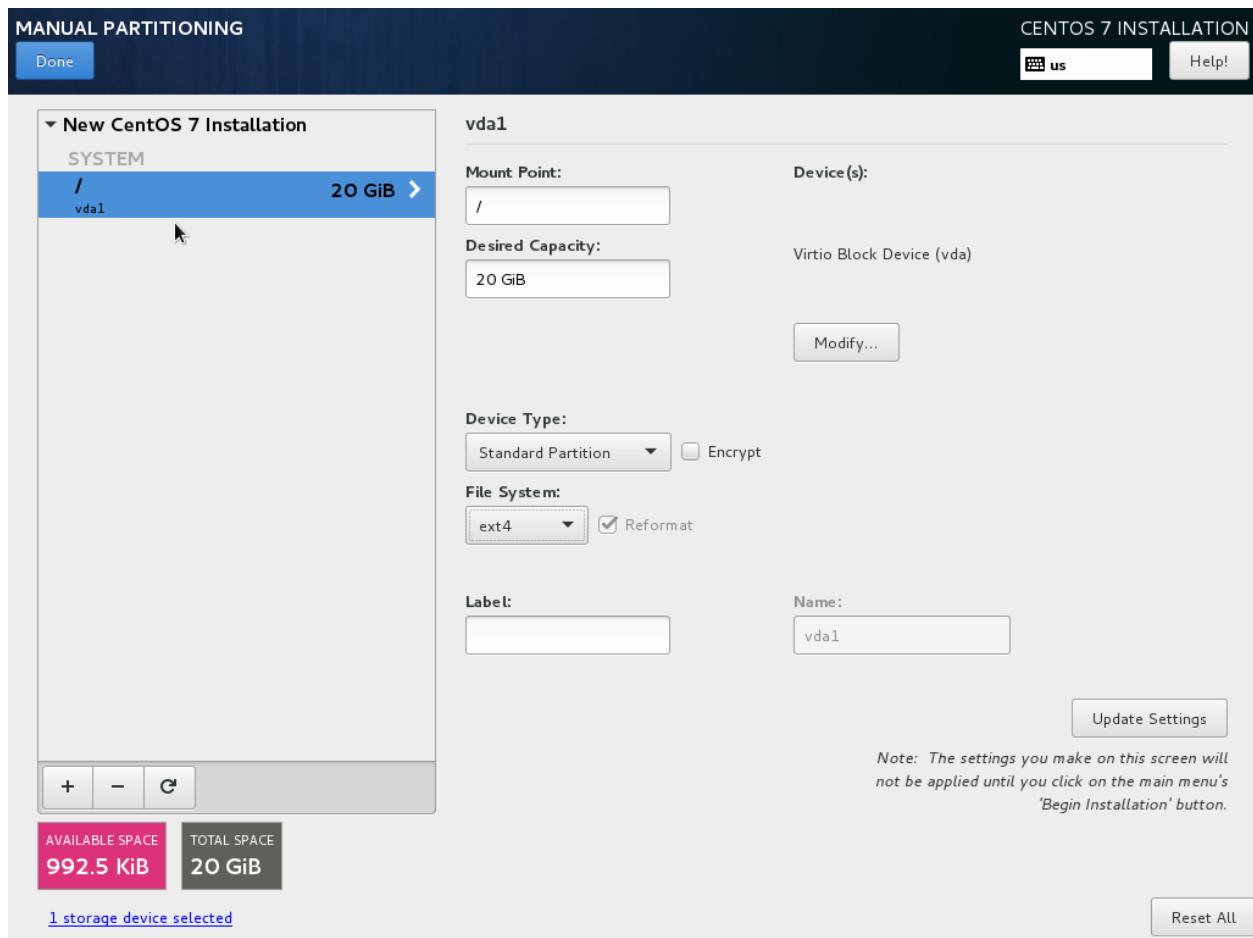
19. Click on “Installation Source”. Set the source to network, and then define a known-good mirror. You can use http://mirror.rackspace.com/CentOS/7.2.1511/os/x86_64/.
20. Click on “Installation Destination”. Select “I will configure partitioning” and click on “Done”:



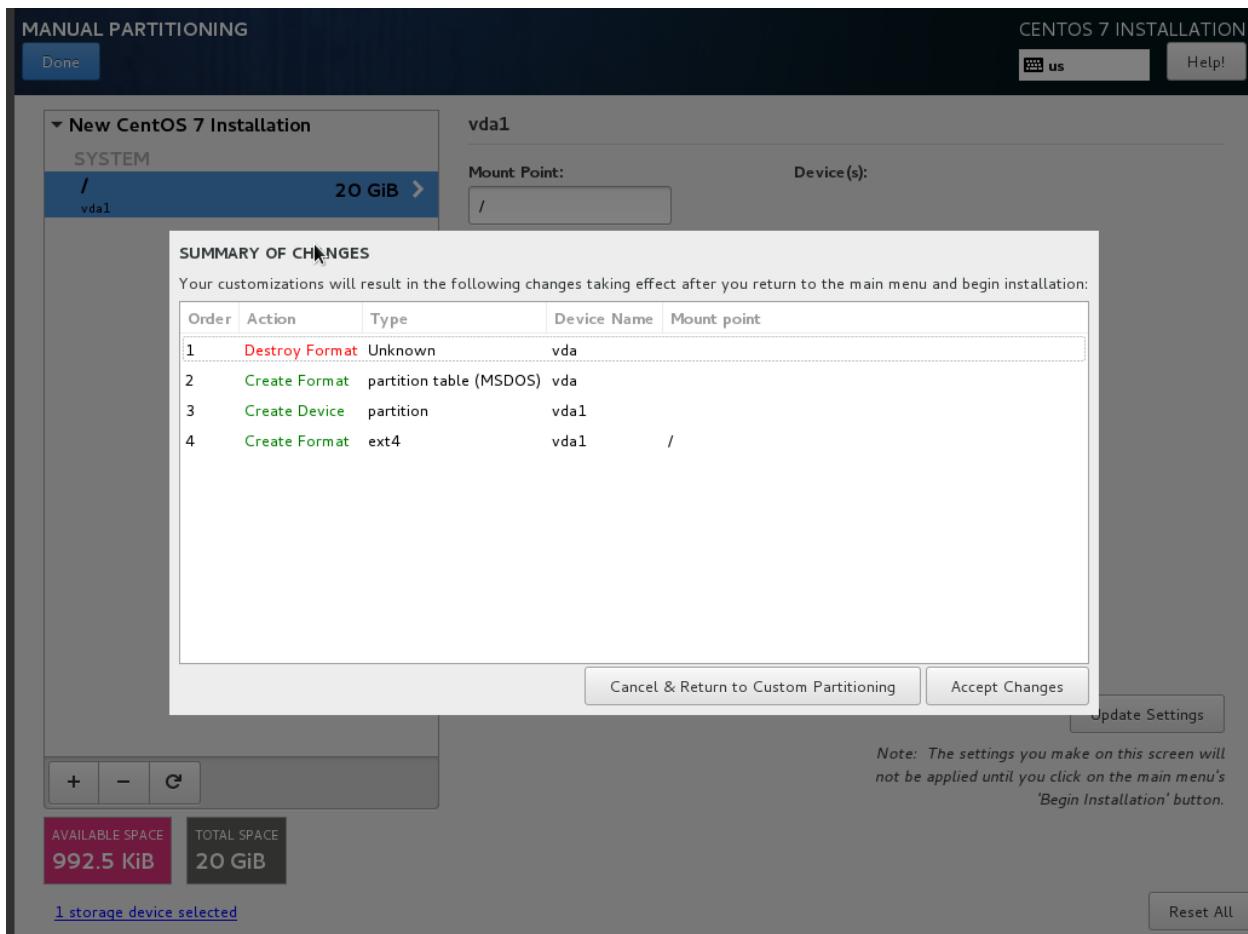
21. Under “New mount points will use the following partition scheme”, select “Standard Partition”.
22. Click on the + button. Set the mount point to / and click “Add mount point”:



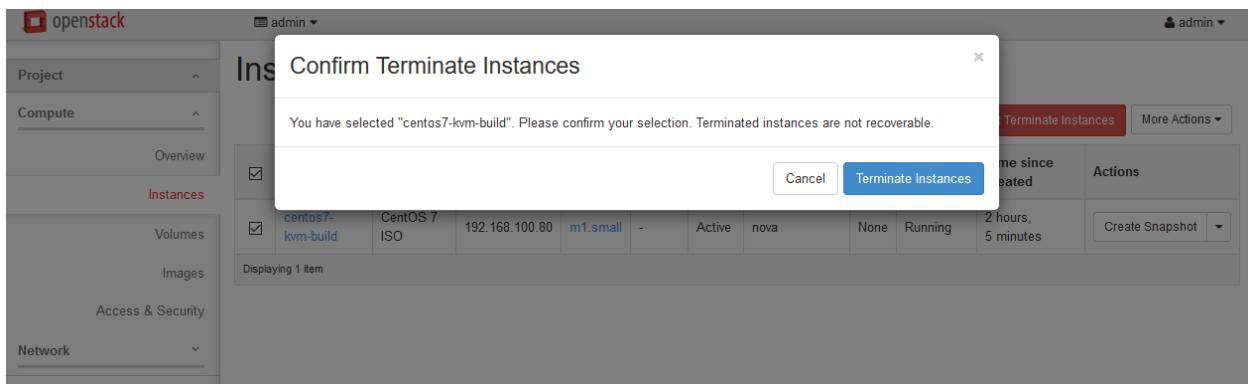
23. Set “File System” to “ext4”, and then click “Done”:



24. A yellow warning bar will appear. Click “Done” again, and then click on “Accept Changes”:



25. Click “Begin installation”. Click on “Root Password” and set a good password.
26. Once installation is complete, click “Reboot”.
27. The server will be attempting to boot from the ISO once more. Press any key to stop the countdown.
28. In the left-hand menu, under “Project” and then “Compute”, click on “Instances”. Select the “centos7-kvm-build” instance, and then click on “Terminate Instances”. Click “Terminate Instances” to confirm:



29. In the left-hand menu, under “Project” and then “Compute”, click on Volumes.
30. Click on the “Actions” drop-down next to “centos7-kvm-build”, and click on “Upload to Image”. Name the image “centos7-kvm-initialkick”, and set the “Disk Format” to “QCOW2”. Upload the image:

Upload Volume to Image

Volume Name *

Image Name *

Disk Format

 ▼

Description:

Upload the volume to the Image Service as an image. This is equivalent to the `cinder upload-to-image` command.

Choose "Disk Format" for the image. The volume images are created with the QEMU disk image utility.

Cancel Upload

31. The volume will go to “Uploading” state. Wait for this to return to “Available” state.
32. In the left-hand menu, under “Project” and then “Compute”, click on “Images”. Click on the “centos7-kvm-initialkick” image, which should be in “Active” state.
33. In the top-right drop-down, click on “Update Metadata”.
34. On the left-hand side, in the “custom” box, enter “hypervisor_type” and click on the + button.
35. On the right-hand side, in the “hypervisor_type” box, enter “kvm”.
36. On the left-hand side, in the “custom” box, enter “auto_disk_config”, and click on the + button.
37. On the right-hand side, in the “auto_disk_config” box, enter “true”.
38. On the left-hand side, in the “custom” box, enter “hw_qemu_guest_agent” and click on the + button.
39. On the right-hand side, in the “hw_qemu_guest_agent” box, enter “true”, and click on the “Save” button:

Update Image Metadata

You can specify resource metadata by moving items from the left column to the right column. In the left columns there are metadata definitions from the Glance Metadata Catalog. Use the "Other" option to add metadata with the key of your choice.

Available Metadata

Custom

+

No available metadata

Existing Metadata

auto_disk_config true

-

hw_qemu_guest_ag... true

-

hypervisor_type kvm

-

auto_disk_config (auto_disk_config)

Cancel

Save

40. In the left-hand menu, under “Project”, and then “Compute”, click on “Volumes”. Highlight the “centos7-kvm-build” volume, and click on “Delete Volumes”. Click “Delete Volumes” to confirm:

<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
<input checked="" type="checkbox"/>	centos7-kvm-build	-	20GB	Available	-		nova	No	No	<button>Edit Volume</button>

Displaying 1 item

41. In the left-hand menu, under “Project” and then “Compute”, click on “Instances”.
42. Click on “Launch Instance”. Give the instance the name “centos7-kvm-build”, use the flavor m1.small (for a 20GB disk), and select “Boot from image” and the “centos7-kvm-initialkick” image. Launch the instance:

Launch Instance

Details * Access & Security Networking * Post-Creation Advanced Options

Availability Zone
nova

Instance Name *
centos7-kvm-build

Flavor * ?
m1.small

Instance Count * ?
1

Instance Boot Source * ?
Boot from image

Image Name *
centos7-kvm-initialkick (1.2 GB)

Flavor Details

Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

Buttons: Cancel Launch

43. Wait for the instance to enter “Active” state. SSH to the new instance as “root”, using the root password used during setup.

44. Delete the static hostname file:

```
# rm /etc/hostname
```

45. Stop and disable the firewalld:

```
# systemctl disable firewalld.service
# systemctl stop firewalld.service
```

46. Disable SELINUX:

```
# setenforce 0
# vim /etc/sysconfig/selinux
```

```
SELINUX=permissive
```

47. Update all packages on the instance:

```
# yum update
```

48. Install the qemu guest agent, cloud-init and cloud-utils:

```
# yum install qemu-guest-agent cloud-init cloud-utils
```

49. Enable and start the qemu-guest-agent service:

```
# systemctl enable qemu-guest-agent.service  
# systemctl start qemu-guest-agent.service
```

50. Enable kernel console logging:

```
# vim /etc/sysconfig/grub
```

- Append “console=ttyS0 console=tty0” to the end of the GRUB_CMDLINE_LINUX setting. For example:

```
GRUB_CMDLINE_LINUX="crashkernel=auto rhgb quiet console=ttyS0 console=tty0"
```

51. Rebuild the grub config file:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

52. Disable user creation at instance creation time:

```
# vim /etc/cloud/cloud.cfg  
  
disable_root: 0
```

- Also delete the “default_user:” section under “system_info”.

53. Delete the static network configuration file:

```
# rm /etc/sysconfig/network-scripts/ifcfg-eth0
```

54. Clear the root bash history:

```
# rm /root/.bash_history; history -c
```

55. In horizon, click the “Create Snapshot” button next to the Instance. Name the image “CentOS 7 (KVM)”:

Create Snapshot

Snapshot Name *

Description:

A snapshot is an image which preserves the disk state of a running instance.

Cancel **Create Snapshot**

56. Wait for the image to go to “Active” state and then, in the drop-down box next to the image, click on “Edit Image”.
57. Check the “public” and “protected” boxes, and click on “Update Image”:

Update Image

Name *
CentOS 7 (KVM)

Description:
Edit the image details.

Description

Kernel ID

Ramdisk ID

Architecture

Format *
QCOW2 - QEMU Emulator

Minimum Disk (GB) ?
20

Minimum RAM (MB) ?
0

Public

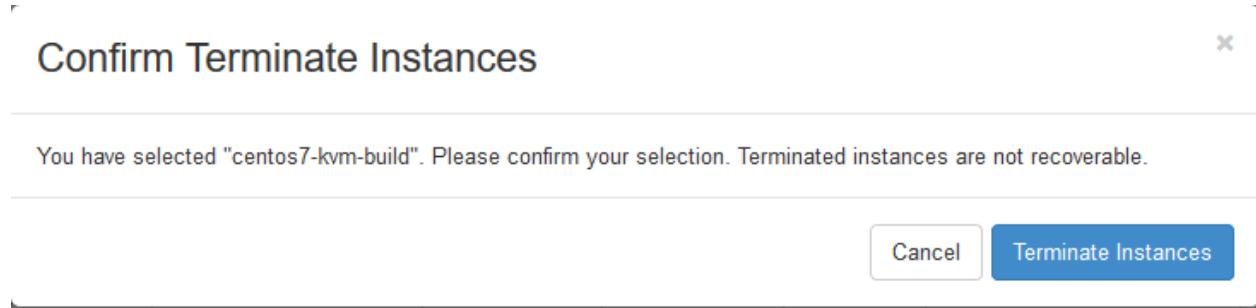
Protected

58. Select the “centos7-kvm-initialkick” image, and click on “Delete Images”. Click “Delete Images” to confirm:

The screenshot shows the OpenStack dashboard with the 'Images' tab selected. The left sidebar shows 'Project' and 'Compute' sections with 'Overview', 'Instances', 'Volumes', and 'Images' options. The main area displays a table of images. The 'centos7-kvm-initialkick' image is selected (indicated by a checked checkbox in the first column). The table columns are: Image Name, Type, Status, Public, Protected, Format, Size, and Actions. The 'Actions' column contains 'Launch Instance' buttons for both rows. At the top of the table, there are filters for 'Project (6)', 'Shared with Me (0)', 'Public (5)', a '+ Create Image' button, and a 'Delete Images' button.

	Image Name	Type	Status	Public	Protected	Format	Size	Actions
<input type="checkbox"/>	CentOS 7 (KVM)	Snapshot	Active	Yes	Yes	QCOW2	1.5 GB	<input type="button" value="Launch Instance"/>
<input checked="" type="checkbox"/>	centos7-kvm-initialkick	Image	Active	No	No	QCOW2	1.2 GB	<input type="button" value="Launch Instance"/>

59. In the left-hand menu, under “Project” and then “Compute”, click on “Instances”.
60. Highlight the “centos7-kvm-build” instance, and click on “Terminate Instances”. Click “Terminate Instances” to confirm:



61. In the left-hand menu, under “Admin” and then “System” click on “Hypervisors”. Next to “compute1-vm”, click on “Enable Service”.

CHAPTER 25

25. Create test KVM CentOS 7 Instance

This page is not based on the OpenStack Installation Guide.

1. From a web browser, access http://*CONTROLLER_ADDRESS*/dashboard.
2. Log in using the demo credentials.
3. In the left-hand menu, under “Project”, and then “Compute”, click on “Instances”.
4. Click on “Launch Instance”. Name the instance “centos7-test”, select the “m1.small” flavor, and “boot from image”. Choose the “CentOS 7 (Xen)” image. Before clicking on “Launch”, click on the “Access & Security” tab:

Launch Instance

[Details *](#) [Access & Security](#) [Networking *](#) [Post-Creation](#) [Advanced Options](#)

Availability Zone

nova

Instance Name *

centos7-test

Flavor * [?](#)

m1.small

Some flavors not meeting minimum image requirements have been disabled.

Instance Count * [?](#)

1

Instance Boot Source * [?](#)

Boot from image

Image Name *

CentOS 7 (KVM) (1.5 GB)

Flavor Details

Name	m1.small
VCPUs	1
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	2,048 MB

Project Limits

Number of Instances	0 of 10 Used
Number of VCPUs	0 of 20 Used
Total RAM	0 of 51,200 MB Used

[Cancel](#) [Launch](#)

5. Ensure that the key pair you just created or imported on page 23 is selected, and then click on Launch:

Launch Instance

Details * Access & Security **Networking *** Post-Creation Advanced Options

Key Pair ?

alex-rax ▼ +

Control access to your instance via key pairs, security groups, and other mechanisms.

Security Groups ?

default

Cancel Launch

6. Wait for the instance to go to “Active” state, and then SSH to the server as “root”, using the key pair you previously created or imported.
7. When you are satisfied that the test instance is working, select it and then click on “Terminate Instances”. Click on “Terminate Instances” to confirm:

The screenshot shows the OpenStack Compute (nova) dashboard. On the left, there's a sidebar with 'Project' dropdown, 'Compute' tab selected (with 'Overview' and 'Instances' sub-options), 'Volumes', and 'Images'. The main area is titled 'Instances' with a sub-header 'Displaying 1 item'. A table lists one instance: 'centos7-test' (Image Name: CentOS 7 (KVM), IP Address: 192.168.100.93, Size: m1.small, Key Pair: alex-rax, Status: Active, Availability Zone: nova, Task: None, Power State: Running, Time since created: 0 minutes). To the right of the table is a 'Actions' column containing a red button labeled 'Terminate Instances'.