

# OpenStack Icehouse Multiple Virtual Machines Manual

Joseph Callen

August 14, 2014

# Chapter 1

## Packstack

### 1.1 Prerequisites

You can use any hypervisor: VMware Workstation, Fusion, Player, ESXi, KVM w/virt-manager (you don't really need virt-manager with KVM, just makes it easier) or VirtualBox. As of this writing download a network-based install CD of Fedora 20. This will allow you to create a minimal install that is all up to date. The OS disk needs minimally 8 GB and a secondary disk for cinder volumes which should be at least 20 GB. I recommend using the NAT network interface since it will provide a route and DNS. We will overlap IP addresses of the DHCP server but that shouldn't be an issue unless you have multiple virtual machines running.

After the OS has been installed lets create our cinder-volumes volume group. In this example the device is named vdb, yours could be different. To determine disk name use `dmesg` or `fdisk -l`.

---

Listing 1.1: Create Cinder Volume VG

---

```
1 vgcreate cinder-volumes /dev/vdb
```

---

---

Listing 1.2: Disable firewalld, enable iptables-services

---

```
1 systemctl enable network
2 systemctl disable firewalld
3 yum install iptables-services -y
4 systemctl enable iptables.service
```

---

The configurations files that Packstack creates will use your IP address not the hostname so we need to make sure that we have a static IP before generating the answer file. Lets first determine your current network information.

---

Listing 1.3: Current IP address

---

```
1 [root@virsatpaw001 ~]# ip a
2 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
```

```

3     link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
4     inet 127.0.0.1/8 scope host lo
5     inet6 ::1/128 scope host
6         valid_lft forever preferred_lft forever
7 2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
    state UP qlen 1000
8     link/ether 52:54:00:e7:12:47 brd ff:ff:ff:ff:ff:ff
9     inet 192.168.122.10/24 brd 192.168.122.255 scope global eth0
10    inet6 fe80::5054:ff:fee7:1247/64 scope link
11        valid_lft forever preferred_lft forever

```

---

Line 9 displays our current IP and subnet mask of 192.168.122.10 /24 (or 255.255.255.0).

#### Listing 1.4: Current Default Route

```

1 [root@virsatpaw001 ~]# ip route
2 192.168.122.0/24 dev eth0  proto kernel  scope link    src 192.168.122.10
3 169.254.0.0/16 dev eth0  scope link    metric 1002
4 default via 192.168.122.1 dev eth0

```

---

Line 4 displays our current default gateway of 192.168.122.1.

#### Listing 1.5: Current Resolver

```

1 [root@virsatpaw001 ~]# cat /etc/resolv.conf
2 nameserver 192.168.122.1

```

---

Based on your device name there will be a corresponding ifcfg file. DEVICE, IPADDR, NETMASK, GATEWAY and DNS1,2 will be based on your network. Modify the file removing options that don't exist in the example below, adding the options that do.

#### Listing 1.6: Modify Ethernet interface from DHCP to static

```

1 vi /etc/sysconfig/network-scripts/ifcfg-

```

---

#### Listing 1.7: Example Ethernet configuration

```

1 DEVICE=eth0
2 TYPE=Ethernet
3 ONBOOT=yes
4 NM_CONTROLLED=no
5 BOOTPROTO=static
6 NAME="eth0"
7 IPADDR=192.168.122.10
8 NETMASK=255.255.255.0
9 GATEWAY=192.168.122.1
10 DNS1=192.168.122.1

```

---

At this point let us reboot. When the virtual machine is available make sure that you can still reach the default gateway, then proceed to the next section.

## Chapter 2

# Physical Hardware

Listing 2.1: Open vSwitch

---

```
1 <network>
2   <name>ovs-network</name>
3   <uuid>2fde288e-242c-4b48-95f4-28f844c768f4</uuid>
4   <forward mode='bridge' />
5   <bridge name='ovsbr0' />
6   <virtualport type='openvswitch' />
7   <portgroup name='vlan-252'>
8     <vlan>
9       <tag id='252' />
10    </vlan>
11  </portgroup>
12  <portgroup name='vlan-253'>
13    <vlan>
14      <tag id='253' />
15    </vlan>
16  </portgroup>
17  <portgroup name='vlan-all'>
18    <vlan trunk='yes'>
19      <tag id='80' />
20      <tag id='81' />
21    </vlan>
22  </portgroup>
23 </network>
```

---

## 2.1 Clone and sysprep

Listing 2.2: Sysprep example

---

```
1 virt-sysprep -a /dev/virtualmachine/virtclpaw001 --hostname
   virtclpaw001.virtomation.com \
```

---

```
2 --firstboot-command "sed -i -r 's/IPADDR=(\b[0-9]{1,3}\.){3}[0-9]{1,3}\
    b'/IPADDR=10.53.252.61/ /etc/sysconfig/network-scripts/ifcfg-eth0" \
3 --firstboot-command 'systemctl restart network' \
4 --firstboot-command 'yum install -y http://rdo.fedorapeople.org/rdo-
    release.rpm' \
5 --firstboot-command 'yum install openstack-packstack -y'
```

---

# Chapter 3

## Prerequisites

Listing 3.1: Bash Aliases

---

```
1 alias yi="yum -y install"
2 alias start="systemctl start"
3 alias e="systemctl enable"
4 alias ocs="openstack-config --set"
```

---

Listing 3.2: Database Install

---

```
1 yi mariadb mariadb-server
2 e mariadb.service
3 start mariadb.service
4 netstat -tanp | grep 3306
5 mysql_secure_installation
```

---

Listing 3.3: RabbitMQ Install

---

```
1 yi rabbitmq-server
2 e rabbitmq-server
3 start rabbitmq-server.service
```

---

Listing 3.4: Create RabbitMQ User Accounts

---

```
1 for serv in "cinder" "nova" "neutron" "heat"; do passwd=`openssl rand -
  base64 8`; echo "$serv - $passwd"; rabbitmqctl add_user $serv
  $passwd; rabbitmqctl set_permissions -p / $serv ".*" ".*" ".*"; done
```

---

Listing 3.5: Result from user account creation

---

```
1 cinder - Q7gPp1FOK5g=
2 Creating user "cinder" ...
3 ...done.
4 nova - 2mM7OaVNFKM=
```

---

```
5 Creating user "nova" ...
6 ...done.
7 neutron - krPOwjPbKJs=
8 Creating user "neutron" ...
9 ...done.
10 heat - l2iDSln7nmw=
11 Creating user "heat" ...
12 ...done.
```

---

# Chapter 4

## Keystone

### 4.1 Installation

Listing 4.1: Install Keystone packages

---

```
1 yi openstack-keystone openstack-utils
2 mysql -u root -p
3 export SERVICE_TOKEN=$(openssl rand -hex 10)
4 echo $SERVICE_TOKEN > ~/ks_admin_token
```

---

### 4.2 Database

Listing 4.2: Keystone configuration

---

```
1 chown -R keystone:keystone /var/log/keystone /etc/keystone/ssl/
2 chmod -R o-rwx /etc/keystone/ssl
3 ocs /etc/keystone/keystone.conf DEFAULT admin_token $SERVICE_TOKEN
4 ocs /etc/keystone/keystone.conf sql connection mysql://keystone:
  trustn0l@10.53.252.61/keystone
5 keystone-manage pki_setup --keystone-user keystone --keystone-group
  keystone
6 su -s /bin/sh -c "keystone-manage db_sync" keystone
```

---

Listing 4.3: Start and enable Keystone

---

```
1 start openstack-keystone
2 e openstack-keystone
```

---

### 4.3 Admin User and Tenant



Listing 4.4: Keystone ???

---

```
1 export OS_SERVICE_TOKEN=$SERVICE_TOKEN
2 export OS_SERVICE_ENDPOINT=http://10.53.252.61:35357/v2.0
3 source /etc/bash_completion.d/keystone.bash_completion
4
5 keystone user-create --name=admin --pass=trustn01
6 keystone role-create --name=admin
7 keystone tenant-create --name=admin --description="Admin Tenant"
8 keystone user-role-add --user=admin --tenant=admin --role=admin
9 keystone user-role-add --user=admin --role=_member_ --tenant=admin
10 keystone tenant-create --name=service --description="Service Tenant"
11 keystone service-create --name=keystone --type=identity --description="
    OpenStack Identity" keystone endpoint-create --service=keystone --
    publicurl=http://10.53.252.61:5000/v2.0 --internalurl=http
    ://10.53.252.61:5000/v2.0 --adminurl=http://10.53.252.61:35357/v2.0
```

---

Listing 4.5: Unset Environment variables

---

```
1 unset OS_SERVICE_ENDPOINT
2 unset OS_ENDPOINT
3 unset OS_SERVICE_TOKEN
4 unset SERVICE_TOKEN
```

---



Make sure that you unset environmental variables or you will receive keystone errors.

Listing 4.6: Keystone Error Message


---

```
1 [root@virctlpaw001 ~]# keystone catalog
2 'NoneType' object has no attribute 'has_service_catalog'
```

---

# Chapter 5

## Swift

 This chapter is a mess, ignore

Listing 5.1: Install Swift

---

```
1 yi glance...
2 yum install -y openstack-swift-proxy \
3 openstack-swift-object \
4 openstack-swift-container \
5 openstack-swift-account \
6 openstack-utils \
7 memcached
```

---

Listing 5.2: Install Swift

---

```
1 fdisk /dev/vdb
2 mkfs.ext4 /dev/vdb1
3 [root@virctlpaw001 ~(keystone_admin)]# blkid /dev/vdb1
4 /dev/vdb1: UUID="7cefc9b8-3313-40cb-941b-78b35c029bac" TYPE="ext4"
   PARTUUID="9faed234-01"
5 vi /etc/fstab
6 mkdir -p /srv/node/d1
7 mount -a
```

---

Listing 5.3: Swift account, container, object

---

```
1 ocs /etc/swift/swift.conf swift-hash swift_hash_path_prefix $(openssl
   rand -hex 10)
2 ocs /etc/swift/swift.conf swift-hash swift_hash_path_suffix $(openssl
   rand -hex 10)
3 ocs /etc/swift/object-server.conf DEFAULT bind_ip 10.53.252.61
4 ocs /etc/swift/account-server.conf DEFAULT bind_ip 10.53.252.61
5 ocs /etc/swift/container-server.conf DEFAULT bind_ip 10.53.252.61
6 for ops_service in "openstack-swift-account" "openstack-swift-container"
   " "openstack-swift-object"; do systemctl enable $ops_service;
   systemctl start $ops_service; done
```

---

---

**Listing 5.4: Swift Proxy**

---

```
1 ocs /etc/swift/proxy-server.conf filter:authtoken auth_host
   10.53.252.61
2 ocs /etc/swift/proxy-server.conf filter:authtoken auth_tenant_name
   service
3 ocs /etc/swift/proxy-server.conf filter:authtoken admin_user swift
4 ocs /etc/swift/proxy-server.conf filter:authtoken admin_password
   trustn01
5 for ops_service in "memcached" "openstack-swift-proxy" ; do systemctl
   enable $ops_service; systemctl start $ops_service; done
```

---

## Chapter 6

# Glance

Listing 6.1: Glance Keystone create

---

```
1 keystone user-create --name glance --pass trustn01
2 keystone user-role-add --user glance --role admin --tenant service
3 keystone service-create --name glance --type image --description "
  Glance Image Service"
4 keystone endpoint-create --service glance --publicurl "http
  ://10.53.252.61:9292" --adminurl "http://10.53.252.61:9292" --
  internalurl "http://10.53.252.61:9292"
```

---

### 6.1 Configuration

Listing 6.2: Glance API

---

```
1 openstack-config --set /etc/glance/glance-api.conf DEFAULT
  sql_connection mysql://glance:trustn01@10.53.252.61/glance
2 ocs /etc/glance/glance-api.conf paste_deploy flavor keystone
3 ocs /etc/glance/glance-api.conf keystone_auth token auth_host
  10.53.252.61
4 ocs /etc/glance/glance-api.conf keystone_auth token auth_port 35357
5 ocs /etc/glance/glance-api.conf keystone_auth token auth_protocol http
6 ocs /etc/glance/glance-api.conf keystone_auth token admin_tenant_name
  service
7 ocs /etc/glance/glance-api.conf keystone_auth token admin_user glance
8 ocs /etc/glance/glance-api.conf keystone_auth token admin_password
  trustn01
```

---

Listing 6.3: Glance Registry

---

```
1 ocs /etc/glance/glance-registry.conf DEFAULT sql_connection mysql://
  glance:trustn01@10.53.252.61/glance
```

---

```

2 ocs /etc/glance/glance-registry.conf paste_deploy flavor keystone
3 ocs /etc/glance/glance-registry.conf keystone_authtoken auth_host
  10.53.252.61
4 ocs /etc/glance/glance-registry.conf keystone_authtoken auth_port 35357
5 ocs /etc/glance/glance-registry.conf keystone_authtoken auth_protocol
  http
6 ocs /etc/glance/glance-registry.conf keystone_authtoken
  admin_tenant_name service
7 ocs /etc/glance/glance-registry.conf keystone_authtoken admin_user
  glance
8 ocs /etc/glance/glance-registry.conf keystone_authtoken admin_password
  trustn01

```

---

#### Listing 6.4: Bugzilla 1090648 - glance-manage db\_sync silently fails to prepare the database

---

```

2014-08-12 15:06:55.083 1694 CRITICAL glance [-] ValueError: Tables "
  migrate_version" have non utf8 collation, please make sure all
  tables are CHARSET=utf8
2014-08-12 15:06:55.083 1694 TRACE glance Traceback (most recent call
  last):
2014-08-12 15:06:55.083 1694 TRACE glance File "/bin/glance-manage",
  line 10, in <module>
2014-08-12 15:06:55.083 1694 TRACE glance sys.exit(main())
2014-08-12 15:06:55.083 1694 TRACE glance File "/usr/lib/python2.7/
  site-packages/glance/cmd/manage.py", line 259, in main
2014-08-12 15:06:55.083 1694 TRACE glance return CONF.command.
  action_fn()
2014-08-12 15:06:55.083 1694 TRACE glance File "/usr/lib/python2.7/
  site-packages/glance/cmd/manage.py", line 160, in sync
2014-08-12 15:06:55.083 1694 TRACE glance CONF.command.
  current_version)
2014-08-12 15:06:55.083 1694 TRACE glance File "/usr/lib/python2.7/
  site-packages/glance/cmd/manage.py", line 137, in sync
2014-08-12 15:06:55.083 1694 TRACE glance sanity_check=self.
  _need_sanity_check()
2014-08-12 15:06:55.083 1694 TRACE glance File "/usr/lib/python2.7/
  site-packages/glance/openstack/common/db/sqlalchemy/migration.py",
  line 195, in db_sync
2014-08-12 15:06:55.083 1694 TRACE glance _db_schema_sanity_check(
  engine)
2014-08-12 15:06:55.083 1694 TRACE glance File "/usr/lib/python2.7/
  site-packages/glance/openstack/common/db/sqlalchemy/migration.py",
  line 221, in _db_schema_sanity_check
2014-08-12 15:06:55.083 1694 TRACE glance ) % ','.join(table_names)
  )
2014-08-12 15:06:55.083 1694 TRACE glance ValueError: Tables "
  migrate_version" have non utf8 collation, please make sure all
  tables are CHARSET=utf8
2014-08-12 15:06:55.083 1694 TRACE glance

```

---

---

**Listing 6.5: Workaround - db\_enforce\_mysql\_charset=False**

---

```
1 vi /etc/glance/glance-api.conf
2 su -s /bin/sh -c "glance-manage db_sync" glance
3 mysql -u glance -p -e "show tables" glance
```

---

---

**Listing 6.6: Start and enable Glance Services**

---

```
1 for ops_service in "openstack-glance-registry" "openstack-glance-api" ;
  do systemctl enable $ops_service; systemctl start $ops_service;
done
```

---

---

**Listing 6.7: Add Cirros Image**

---

```
1 wget http://download.cirros-cloud.net/0.3.2/cirros-0.3.2-x86_64-disk.
  img
2
3 glance image-create --name "cirros" --is-public true --disk-format
  qcow2 --container-format bare --file cirros-0.3.2-x86_64-disk.img
```

---

## Chapter 7

# Cinder

---

```
1 keystone user-create --name cinder --pass trustn01
2
3 keystone user-role-add --user cinder --role admin --tenant service
4 keystone service-create --name cinder --type volume --description "
  Cinder Volume Service"
5
6 keystone endpoint-create --service cinder --publicurl "http
  ://10.53.252.62:8776/v1/\$(tenant_id)s" --adminurl "http
  ://10.53.252.62:8776/v1/\$(tenant_id)s" --internalurl "http
  ://10.53.252.62:8776/v1/\$(tenant_id)s"
7
8 mysql -u cinder -p -e "show tables" cinder
```

---

```
1 yi openstack-cinder openstack-utils scsi-target-utils
2
3 ocs /etc/cinder/cinder.conf DEFAULT auth_strategy keystone
4 ocs /etc/cinder/cinder.conf keystone_auth_token auth_host 10.53.252.61
5 ocs /etc/cinder/cinder.conf keystone_auth_token admin_tenant_name
  service
6 ocs /etc/cinder/cinder.conf keystone_auth_token admin_user cinder
7 ocs /etc/cinder/cinder.conf keystone_auth_token admin_password trustn01
8
9 vgcreate cinder-volumes /dev/vdb
10
11
12 ocs /etc/cinder/cinder.conf DEFAULT rpc_backend cinder.openstack.common
  .rpc.impl_kombu
13 ocs /etc/cinder/cinder.conf DEFAULT rabbit_host 10.53.252.61
14 ocs /etc/cinder/cinder.conf DEFAULT rabbit_port 5672
15 ocs /etc/cinder/cinder.conf DEFAULT rabbit_userid cinder
16 ocs /etc/cinder/cinder.conf DEFAULT rabbit_password Q7gPp1FOK5g=
17
18 ocs /etc/cinder/cinder.conf DEFAULT sql_connection mysql://cinder:
```

```
trustn01@10.53.252.61/cinder
19
20 su -s /bin/sh -c "cinder-manage db sync" cinder
```

---

---

Listing 7.1: add \* to include /etc/cinder/volumes/

---

```
1 add * to include /etc/cinder/volumes/
2 vi /etc/tgt/conf.d/cinder.conf
```

---



### Confirm RabbitMQ Status

---

```
1 rabbitmqctl status
```

---