OpenStack Commands Cheat Sheet

Published March 13, 2014 • Updated November 5, 2016

documentation **openstack** **cheat sheet**

I have found most of the OpenStack commands to be fairly intuitive. However, there are some commands that do not follow a standard syntax, and there are far too many commands and command line switches to remember. In addition, Google searching for specific commands can be time consuming.

This post will be an ever growing list of OpenStack commands I have used across the various OpenStack Projects.

Monitor OpenStack Service Logs

Here are some quick and dirty way to watch the necessary logs on the OpenStack Controller and Compute nodes.

Ubuntu

Controller logs:

```
tail -f
/var/log/{ceilometer,cinder,glance,keystone,mysql,neutron,nova,openvswitch,rabbitmq}/*.log
/var/log/syslog
```

Compute logs:

```
tail -f /var/log/{ceilometer,neutron,nova,openvswitch}/*.log /var/log/syslog
```

CentOS/RHEL

Controller logs:

```
tail -f
/var/log/{ceilometer,cinder,glance,keystone,mysql,neutron,nova,openvswitch,rabbitmq}/*.log
/var/log/messages
```

Compute logs:

```
tail -f /var/log/{ceilometer,neutron,nova,openvswitch}/*.log /var/log/messages
```

Keystone

See Status of Keystone Services

```
keystone service-list
```

List All Keystone Endpoints

```
keystone endpoint-list
```

Glance

List Current Glance Images

```
glance image-list
```

Upload Images to Glance

```
glance image-create --name <IMAGE-NAME> --is-public <true OR false> --container-format
<CONTAINER-FORMAT> --disk-format <DISK-FORMAT> --copy-from <URI>
```

Example 1: Upload the **cirros-0.3.2-x86_64** OpenStack cloud image:

```
glance image-create --name cirros-0.3.2-x86_64 --is-public true --container-format bare --
disk-format qcow2 --copy-from http://download.cirros-cloud.net/0.3.2/cirros-0.3.2-x86_64-
disk.img
```

Example 2: Upload the **ubuntu-server-12.04** OpenStack cloud image:

```
glance image-create --name ubuntu-server-12.04 --is-public true --container-format bare --
disk-format qcow2 --copy-from http://cloud-images.ubuntu.com/precise/current/precise-
server-cloudimg-amd64-disk1.img
```

Nova

See Status of Nova Services

```
nova service-list
```

List Current Nova Instances

```
nova list
```

Boot an Instance

Boot an instance assigned to a particular Neutron Network:

```
nova boot <INSTANCE-NAME> --image <GLANCE-IMAGE-ID> --flavor <FLAVOR-ID> --security-groups <SEC-GROUP-1,SEC-GROUP-2> --key-name <SSH-KEY-NAME> --nic net-id=<NET-ID> --availability-zone <AVAILABILITY-ZONE-NAME>
```

Boot an instance assigned to a particular Neutron Port:

```
nova boot <INSTANCE-NAME> --image <GLANCE-IMAGE-ID> --flavor <FLAVOR-ID> --security-groups <SEC-GROUP-1,SEC-GROUP-2> --key-name <SSH-KEY-NAME> --nic port-id=<PORT-ID> --availability-zone <AVAILABILITY-ZONE-NAME>
```

Create a Flavor

```
nova flavor-create <FLAVOR-NAME> <FLAVOR-ID> <RAM-IN-MB> <ROOT-DISK-IN-GB> <VCPU>
```

For example, create a new flavor called **m1.custom** with an ID of **6**, **512 MB** of RAM, **5 GB** of root disk space, and **1** vCPU:

```
nova flavor-create m1.custom 6 512 5 1
```

Create Nova Security Group

This command is only used if you are using **nova-network**.

```
nova secgroup-create <NAME> <DESCRIPTION>
```

Add Rules to Nova Security Group

These command is only used if you are using **nova-network**.

```
nova secgroup-add-rule <NAME> <PROTOCOL> <BEGINNING-PORT> <ENDING-PORT> <SOURCE-SUBNET>
```

Example 1: Add a rule to the **default** Nova Security Group to allow SSH access to instances:

```
nova secgroup-add-rule default tcp 22 22 0.0.0.0/0
```

Example 2: Add a rule to the **default** Nova Security Group Rule to allow ICMP communication to instances:

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

Apply Nova Security Group to Instance

This command is only used if you are using **nova-network**.

```
nova add-secgroup <NOVA-ID> <SECURITY-GROUP-ID>
```

Create Nova Floating IP Pool

These command is only used if you are using **nova-network**.

```
nova-manage floating create <SUBNET-NAME> <NAME-OF-POOL>
```

Create Nova Key SSH Pair

```
nova keypair-add --pub_key <SSH-PUBLIC-KEY-FILE-NAME> <NAME-OF-KEY>
```

Create Host Aggregate With Availability Zone

```
nova aggregate-create <HOST-AGG-NAME> <AVAIL-ZONE-NAME>
```

Add Compute Host to Host Aggregate

```
nova aggregate-add-host <HOST-AGG-ID> <COMPUTE-HOST-NAME>
```

Live Migrate an Instance

If your compute hosts use shared storage:

```
nova live-migration <INSTANCE-ID> <COMPUTE-HOST-ID>
```

If your compute hosts do *not* use shared storage:

```
nova live-migration --block-migrate <INSTANCE-ID> <COMPUTE-HOST-ID>
```

Attach Cinder Volume to Instance

Before running this command, you will need to have already created the particular Cinder Volume.

```
nova volume-attach <INSTANCE-ID> <CINDER-VOLUME-ID> <DEVICE (use auto)>
```

Create and Boot an Instance from a Cinder Volume

Before running this command, you will need to have already created the particular Cinder Volume from a Glance Image.

```
nova boot --flavor <FLAVOR-ID> --block_device_mapping vda=<CINDER-VOLUME-ID>:::0 <INSTANCE-NAME>
```

Create and Boot an Instance from a Cinder Volume Snapshot

Before running this command, you will have to have already created the particular Cinder Volume Snapshot:

```
nova boot --flavor <FLAVOR-ID> --block_device_mapping vda=<CINDER-SNAPSHOT-ID>:snap::0 <INSTANCE-NAME>
```

Reset the State of an Instance

If an instance gets stuck in a **delete** state, the instance state can be reset and then deleted:

```
nova reset-state <INSTANCE-ID>

nova delete <INSTANCE-ID>
```

You can also use the **active** command line switch to force an instance back into an active state:

```
nova reset-state --active <INSTANCE-ID>
```

Get Direct URL to Instance Console Using novnc

```
nova get-vnc-console <INSTANCE-ID> novnc
```

Get Direct URL to Instance Console Using xvpvnc

```
nova get-vnc-console <INSTANCE-ID> xvpvnc
```

Set OpenStack Project Nova Quota

The following command will set an unlimited quota for a particular OpenStack Project:

```
nova quota-update --instances -1 --cores -1 --ram -1 --floating-ips -1 --fixed-ips -1 --
metadata-items -1 --injected-files -1 --injected-file-content-bytes -1 --injected-file-
path-bytes -1 --key-pairs -1 --security-groups -1 --security-group-rules -1 --server-groups
-1 --server-group-members -1 <PROJECT ID>
```

Cinder

See Status of Cinder Services

```
cinder service-list
```

List Current Cinder Volumes

```
cinder list
```

Create Cinder Volume

```
cinder create --display-name <CINDER-IMAGE-DISPLAY-NAME> <SIZE-IN-GB>
```

Create Cinder Volume from Glance Image

```
cinder create --image-id <GLANCE-IMAGE-ID> --display-name <CINDER-IMAGE-DISPLAY-NAME>
<SIZE-IN-GB>
```

Create Snapshot of Cinder Volume

```
cinder snapshot-create --display-name <SNAPSHOT-DISPLAY-NAME> <CINDER-VOLUME-ID>
```

If the Cinder Volume is not available, i.e. it is currently attached to an instance, you must pass the force flag:

```
cinder snapshot-create --display-name <SNAPSHOT-DISPLAY-NAME> <CINDER-VOLUME-ID> --force
True
```

Neutron

See Status of Neutron Services

```
neutron agent-list
```

List Current Neutron Networks

```
neutron net-list
```

List Current Neutron Subnets

```
neutron subnet-list
```

Rename Neutron Network

```
neutron net-update <CURRENT-NET-NAME> --name <NEW-NET-NAME>
```

Rename Neutron Subnet

```
neutron subnet-update <CURRENT-SUBNET-NAME> --name <NEW-SUBNET-NAME>
```

Create Neutron Security Group

```
neutron security-group-create <SEC-GROUP-NAME>
```

Add Rules to Neutron Security Group

```
neutron security-group-rule-create --direction <ingress OR egress> --ethertype <IPv4 or
IPv6> --protocol <PROTOCOL> --port-range-min <PORT-NUMBER> --port-range-max <PORT-NUMBER>
<SEC-GROUP-NAME>
```

Example 1: Add a rule to the **default** Neutron Security Group to allow SSH access to instances:

```
neutron security-group-rule-create --direction ingress --ethertype IPv4 --protocol tcp --
port-range-min 22 --port-range-max 22 default
```

Example 2: Add a rule to the **default** Neutron Security Group to allow ICMP communication to instances:

```
neutron security-group-rule-create --direction ingress --ethertype IPv4 --protocol icmp
default
```

Create a Neutron Tenant Network

```
neutron net-create <NET-NAME>

neutron subnet-create --name <SUBNET-NAME> <NET-NAME> <SUBNET-CIDR>
```

Create a Neutron Provider Network

```
neutron net-create <NET-NAME> --provider:physical_network=<LABEL-PHYSICAL-INTERFACE> --
provider:network_type=<flat or vlan> --shared --router:external=True

neutron subnet-create --name <SUBNET-NAME> <NET-NAME> <SUBNET-CIDR>  --gateway <GATEWAY-IP>
--allocation-pool start=<STARTING-IP>,end=<ENDING-IP> --dns-nameservers list=true <DNS-1
DNS-2>
```

Create a Neutron Router

```
neutron router-create <ROUTER-NAME>
```

Set Default Gateway on a Neutron Router

```
neutron router-gateway-set <ROUTER-NAME> <NET-NAME>
```

Attach a Tenant Network to a Neutron Router

```
neutron router-interface-add <ROUTER-NAME> <SUBNET-NAME>
```

Create a Neutron Floating IP Pool

If you need N number of floating IP addresses, run this command N number of times:

```
neutron floatingip-create <NET-NAME>
```

Assign a Neutron Floating IP Address to an Instances

```
neutron floatingip-associate <FLOATING-IP-ID> <NEUTRON-PORT-ID>
```

Create a Neutron Port with a Fixed IP Address

```
neutron port-create <NET-NAME> --fixed-ip ip_address=<IP-ADDRESS>
```

Set OpenStack Project Neutron Quota

The following command will allow an unlimited number of Neutron Ports to be created within a particular OpenStack Project:

```
neutron quota-update --tenant-id=<PROJECT ID> --port -1
```