



**KUMARAGURU**  
college of technology  
character is life ,

**Department of Information Technology**

**U18ITE0221 CLOUD INFRASTRUCTURE AND  
ARCHITECTURE**

**Lab Manual**

Prepared by  
Mr. C.Jeganathan

## CONTENTS

Ex no.	Title of the Exercise	Page no.
1.	Configure NOVA compute Node	
2.	Configure Swift object storage	
3.	Construct a cinder block node	
4.	Build a horizon node – Monitor node	
5.	Launching an instance- Register an account at openstack, Create SSH Key, validate network.	
6.	Sharing project environment among multiple users.	

## Ex.No: 01

Configure NOVA compute Node

### Aim:

To Configure NOVA compute Node using Openstack

### Procedure:

#### Minimum Requirements

Before we begin, ensure you have the following minimum prerequisites

1. A fresh Ubuntu 22.04 installation
2. User with sudo privileges
3. 4 GB RAM
4. 2 vCPUs
5. Hard disk capacity of 10 GB
6. Internet connection

With the minimum requirements satisfied, we can now proceed.

#### Step 1: Update and Upgrade the System

To start off, log into your Ubuntu 22.04 system using SSH protocol and update & upgrade system repositories using the following command.

```
apt update -y && apt upgrade -y
```

Next reboot the system using the command.

```
sudo reboot
```

OR

```
init 6
```

#### Step 2: Create Stack user and assign sudo privilege

Best practice demands that devstack should be run as a regular user with [sudo](#) privileges. With that in mind, we are going to add a new user called “stack” and assign sudo privileges. To create stack user execute

```
sudo adduser -s /bin/bash -d /opt/stack -m stack
```

```
sudo chmod +x /opt/stack
```

Next, run the command below to assign sudo privileges to the user

```
echo "stack ALL=(ALL) NOPASSWD: ALL" | sudo tee /etc/sudoers.d/stack
```

#### Step 3: Install git and download DevStack

Once you have successfully created the user ‘stack’ and assigned sudo privileges, switch to the user using the command.

```
su - stack
```

In most Ubuntu 22.04 systems, git comes already installed. If by any chance git is missing, install it by running the following command.

```
sudo apt install git -y
```

Sample output

```
root@ubuntu:~# su - stack
stack@ubuntu:~$
stack@ubuntu:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.17.1-1ubuntu0.4).
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Using git, clone devstack’s git repository as shown.

```
git clone https://git.openstack.org/openstack-dev/devstack
```

Sample output

```

stack@ubuntu:~$ git clone https://git.openstack.org/openstack-dev/devstack
Cloning into 'devstack'...
warning: redirecting to https://opendev.org/openstack/devstack/
remote: Enumerating objects: 43615, done.
remote: Counting objects: 100% (43615/43615), done.
remote: Compressing objects: 100% (12575/12575), done.
remote: Total 43615 (delta 31152), reused 42370 (delta 30360)
Receiving objects: 100% (43615/43615), 8.27 MiB | 24.61 MiB/s, done.
Resolving deltas: 100% (31152/31152), done.
stack@ubuntu:~$
stack@ubuntu:~$ ls
devstack
stack@ubuntu:~$

```

Step 4: Create devstack configuration file

In this step, navigate to the devstack directory.

```
cd devstack
```

Then create a local.conf configuration file.

```
vim local.conf
```

Paste the following content

```
[[local|localrc]]# Password for KeyStone, Database, RabbitMQ and Service
```

```
ADMIN_PASSWORD=StrongAdminSecret
```

```
DATABASE_PASSWORD=$ADMIN_PASSWORD
```

```
RABBIT_PASSWORD=$ADMIN_PASSWORD
```

```
SERVICE_PASSWORD=$ADMIN_PASSWORD# Host IP - get your Server/VM IP address
from ip addr command
```

```
HOST_IP=10.208.0.10
```

Save and exit the text editor. NOTE:

1. The ADMIN\_PASSWORD is the password that you will use to log in to the OpenStack login page. The default username is admin.
2. The HOST\_IP is your system's IP address that is obtained by running ifconfig or ip addr commands.

Step 5: Install OpenStack with Devstack

To commence the installation of OpenStack on Ubuntu 22.04, run the script below contained in devstack directory.

```
./stack.sh
```

The following features will be installed:

- Horizon — OpenStack Dashboard
- Nova — Compute Service
- Glance — Image Service
- Neutron — Network Service
- Keystone — Identity Service
- Cinder — Block Storage Service
- Placement — Placement API

The deployment takes about 10 to 15 minutes depending on the speed of your system and internet connection. In our case, it took roughly 12 minutes. At the very end, you should see output similar to what we have below.

```

        print a[2]
    }
    ' /opt/stack/devstack/local.conf
+./stack.sh:main:1489                               set to xtrace

=====
DevStack Component Timing
(times are in seconds)
=====
run_process           53
test_with_retry       2
apt-get-update        1
osc                   177
wait_for_service      21
dbsync                56
pip_install           149
apt-get               7
-----
Unaccounted time      418
-----
Total runtime         884

This is your host IP address: 10.128.0.8
This is your host IPv6 address: ::1
Horizon is now available at http://10.128.0.8/dashboard
Keystone is serving at http://10.128.0.8/identity/
The default users are: admin and demo
The password: StrongAdminSecret

WARNING:
Using lib/neutron-legacy is deprecated, and it will be removed in the future

Services are running under systemd unit files.
For more information see:
https://docs.openstack.org/devstack/latest/systemd.html

DevStack Version: train
Change: 16d11d27f375b8c027bbc3a1db1885e90ce6c604 Merge "Option "lock_path" from group "DEFAULT"
OS Version: Ubuntu 18.04 bionic

2019-06-04 12:19:19.207 | stack.sh completed in 884 seconds.

```

This confirms that all went well and that we can proceed to access OpenStack via a web browser.

Step 6: Accessing OpenStack on a web browser

To access OpenStack via a web browser browse your Ubuntu's IP address as shown.


<https://server-ip/dashboard> This directs you to a login page as shown.



**openstack®**

Log in

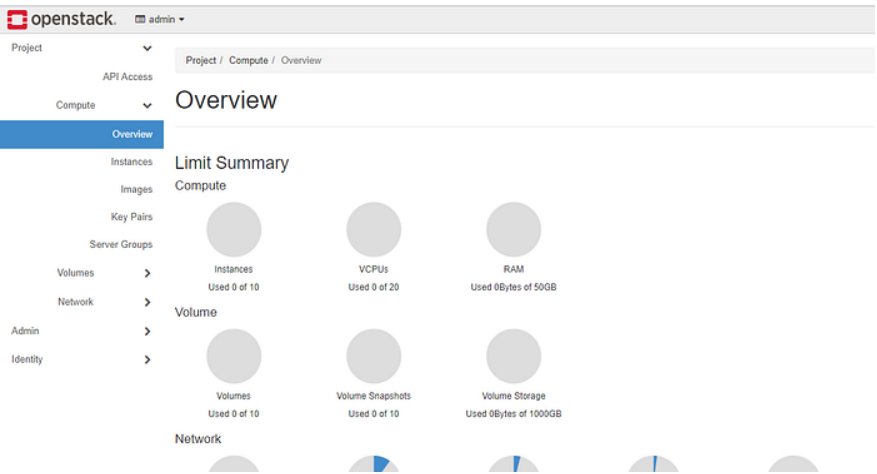
User Name

Password  
 

[Sign In](#)

Enter the credentials and hit “Sign In” You should be able to see the Management console

dashboard as shown below.



**Ex.No: 02**

Configure Swift object storage

**Aim:**

To Configure Swift object storage

**Procedure:****Create and manage object containers**

OpenStack Object Storage (swift) is used for redundant, scalable data storage using clusters of standardized servers to store petabytes of accessible data. It is a long-term storage system for large amounts of static data which can be retrieved and updated.

OpenStack Object Storage provides a distributed, API-accessible storage platform that can be integrated directly into an application or used to store any type of file, including VM images, backups, archives, or media files. In the OpenStack dashboard, you can only manage containers and objects.

In OpenStack Object Storage, containers provide storage for objects in a manner similar to a Windows folder or Linux file directory, though they cannot be nested. An object in OpenStack consists of the file to be stored in the container and any accompanying metadata.

**Create a container**

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Click Container.
5. In the Create Container dialog box, enter a name for the container, and then click Create.

You have successfully created a container.

**Note**

To delete a container, click the More button and select Delete Container.

**Upload an object**

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Select the container in which you want to store your object.
5. Click the Upload File icon.

The Upload File To Container: <name> dialog box appears. <name> is the name of the container to which you are uploading the object.

6. Enter a name for the object.
7. Browse to and select the file that you want to upload.
8. Click Upload File.

You have successfully uploaded an object to the container.

**Note**

To delete an object, click the More button and select Delete Object.

**Manage an object****To edit an object**

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Select the container in which you want to store your object.
5. Click the menu button and choose Edit from the dropdown list.

The Edit Object dialog box is displayed.

6. Browse to and select the file that you want to upload.
7. Click Update Object.

Note

To delete an object, click the menu button and select Delete Object.

**To copy an object from one container to another**

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Select the container in which you want to store your object.
5. Click the menu button and choose Copy from the dropdown list.
6. In the Copy Object launch dialog box, enter the following values:
  - Destination Container: Choose the destination container from the list.
  - Path: Specify a path in which the new copy should be stored inside of the selected container.
  - Destination object name: Enter a name for the object in the new container.
7. Click Copy Object.

**To create a metadata-only object without a file**

You can create a new object in container without a file available and can upload the file later when it is ready. This temporary object acts a place-holder for a new object, and enables the user to share object metadata and URL info in advance.

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Select the container in which you want to store your object.
5. Click Upload Object.

The Upload Object To Container: <name> dialog box is displayed.

<name> is the name of the container to which you are uploading the object.

6. Enter a name for the object.
7. Click Update Object.

**To create a pseudo-folder**

Pseudo-folders are similar to folders in your desktop operating system. They are virtual collections defined by a common prefix on the object's name.

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Object Store tab and click Containers category.
4. Select the container in which you want to store your object.
5. Click Create Pseudo-folder.

The Create Pseudo-Folder in Container <name> dialog box is displayed. <name> is the name of the container to which you are uploading the object.

6. Enter a name for the pseudo-folder.

A slash (/) character is used as the delimiter for pseudo-folders in Object Storage.

7. Click Create.



### Ex.No: 03

Construct a cinder block node

#### Aim:

To Construct a cinder block node

This section describes how to install and configure the Block Storage service, code-named cinder, on the controller node. This service requires at least one additional storage node that provides volumes to instances.

#### Prerequisites

Before you install and configure the Block Storage service, you must create a database, service credentials, and API endpoints.

1. To create the database, complete these steps:
  1. Use the database access client to connect to the database server as the root user:
    1. ☐
    2. mysql
    3. Create the cinder database:
    4. MariaDB [(none)]> CREATE DATABASE cinder;
    5. Grant proper access to the cinder database:
    6. MariaDB [(none)]> GRANT ALL PRIVILEGES ON cinder.\* TO 'cinder'@'localhost' \
    7. IDENTIFIED BY 'CINDER\_DBPASS';
    8. MariaDB [(none)]> GRANT ALL PRIVILEGES ON cinder.\* TO 'cinder'@'%' \
    9. IDENTIFIED BY 'CINDER\_DBPASS';

Replace CINDER\_DBPASS with a suitable password.

10. Exit the database access client.

- ☐ Source the admin credentials to gain access to admin-only CLI commands:
- ☐ . admin-openrc
- ☐ To create the service credentials, complete these steps:
  1. Create a cinder user:
- ☐ openstack user create --domain default --password-prompt cinder

User Password:

Repeat User Password:

+-----+-----+		
Field	Value	
+-----+-----+		
domain_id	default	
enabled	True	
id	9d7e33de3e1a498390353819bc7d245d	
name	cinder	
options	{ }	
password_expires_at	None	
+-----+-----+		

- ☐ Add the admin role to the cinder user:
- ☐ openstack role add --project service --user cinder admin

#### Note

This command provides no output.

- ☐ Create the cinderv3 service entity:
  3. ☐
  4. openstack service create --name cinderv3 \
  5. --description "OpenStack Block Storage" volumev3
  - 6.

7.	+-----+		
8.	Field	Value	
9.	+-----+		
10.	description	OpenStack Block Storage	
11.	enabled	True	
12.	id	ab3bbbf780845a1a283490d281e7fda	
13.	name	cinderv3	
14.	type	volumev3	
15.	+-----+		

#### Note

Beginning with the Xena release, the Block Storage services require only one service entity. For prior releases, please consult the documentation for that specific release.

□ Create the Block Storage service API endpoints:

openstack endpoint create --region RegionOne \  
 volumev3 public http://controller:8776/v3/%(project\_id)s

+-----+		
Field	Value	
+-----+		
enabled	True	
id	03fa2c90153546c295bf30ca86b1344b	
interface	public	
region	RegionOne	
region_id	RegionOne	
service_id	ab3bbbf780845a1a283490d281e7fda	
service_name	cinderv3	
service_type	volumev3	
url	http://controller:8776/v3/%(project_id)s	
+-----+		

openstack endpoint create --region RegionOne \  
 volumev3 internal http://controller:8776/v3/%(project\_id)s

+-----+		
Field	Value	
+-----+		
enabled	True	
id	94f684395d1b41068c70e4ecb11364b2	
interface	internal	
region	RegionOne	
region_id	RegionOne	
service_id	ab3bbbf780845a1a283490d281e7fda	
service_name	cinderv3	
service_type	volumev3	
url	http://controller:8776/v3/%(project_id)s	
+-----+		

4. openstack endpoint create --region RegionOne \  
 5. volumev3 admin http://controller:8776/v3/%(project\_id)s

6.	+-----+		
7.	+-----+		
8.	Field	Value	
9.	+-----+		
10.	enabled	True	
11.	id	4511c28a0f9840c78bach25f10f62c98	

```

12. | interface | admin |
13. | region | RegionOne |
14. | region_id | RegionOne |
15. | service_id | ab3bbbf780845a1a283490d281e7fda |
16. | service_name | cinderv3 |
17. | service_type | volumev3 |
18. | url | http://controller:8776/v3/%(project_id)s |
19. +-----+-----+
20.

```

Install and configure components

1. Install the packages:

☐ apt install cinder-api cinder-scheduler

☐ Edit the /etc/cinder/cinder.conf file and complete the following actions:

1. In the [database] section, configure database access:

2. [database]

3. # ...

4. connection = mysql+pymysql://cinder:CINDER\_DBPASS@controller/cinder

Replace CINDER\_DBPASS with the password you chose for the Block Storage database.

5. In the [DEFAULT] section, configure RabbitMQ message queue access:

6. [DEFAULT]

7. # ...

8. transport\_url = rabbit://openstack:RABBIT\_PASS@controller

Replace RABBIT\_PASS with the password you chose for the openstack account in RabbitMQ.

9. In the [DEFAULT] and [keystone\_authtoken] sections, configure Identity service access:

10. [DEFAULT]

11. # ...

12. auth\_strategy = keystone

13.

14. [keystone\_authtoken]

15. # ...

16. www\_authenticate\_uri = http://controller:5000

17. auth\_url = http://controller:5000

18. memcached\_servers = controller:11211

19. auth\_type = password

20. project\_domain\_name = default

21. user\_domain\_name = default

22. project\_name = service

23. username = cinder

24. password = CINDER\_PASS

Replace CINDER\_PASS with the password you chose for the cinder user in the Identity service.

Note

Comment out or remove any other options in the [keystone\_authtoken] section.

25. In the [DEFAULT] section, configure the my\_ip option to use the management interface IP address of the controller node:

26. [DEFAULT]

27. # ...

28. my\_ip = 10.0.0.11

☐ In the [oslo\_concurrency] section, configure the lock path:

[oslo\_concurrency]

# ...

lock\_path = /var/lib/cinder/tmp

☐ Populate the Block Storage database:

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

5.

Note

Ignore any deprecation messages in this output.

Configure Compute to use Block Storage

1. Edit the `/etc/nova/nova.conf` file and add the following to it:
2. `[cinder]`
3. `os_region_name = RegionOne`

Finalize installation

1. Restart the Compute API service:

☐ `service nova-api restart`

☐ Restart the Block Storage services:

`service cinder-scheduler restart`

`service apache2 restart`

## Ex.No: 04

Build a horizon node – Monitor node

### Aim:

To Build a horizon node – Monitor node

### Procedure :

This section describes how to install and configure the dashboard on the controller node. The only core service required by the dashboard is the Identity service. You can use the dashboard in combination with other services, such as Image service, Compute, and Networking. You can also use the dashboard in environments with stand-alone services such as Object Storage.

#### Note

This section assumes proper installation, configuration, and operation of the Identity service using the Apache HTTP server and Memcached service.

### Install and configure components

#### Note

Default configuration files vary by distribution. You might need to add these sections and options rather than modifying existing sections and options. Also, an ellipsis (...) in the configuration snippets indicates potential default configuration options that you should retain.

1. Install the packages:
  1. apt install openstack-dashboard
  - 2.
  3. Edit the /etc/openstack-dashboard/local\_settings.py file and complete the following actions:
    - Configure the dashboard to use OpenStack services on the controller node:
    - OPENSTACK\_HOST = "controller"
    - In the Dashboard configuration section, allow your hosts to access Dashboard:
    - ALLOWED\_HOSTS = ['one.example.com', 'two.example.com']

#### Note

- Do not edit the ALLOWED\_HOSTS parameter under the Ubuntu configuration section.
- ALLOWED\_HOSTS can also be ['\*'] to accept all hosts. This may be useful for development work, but is potentially insecure and should not be used in production. See the [Django documentation](#) for further information.
- Configure the memcached session storage service:
- SESSION\_ENGINE = 'django.contrib.sessions.backends.cache'
- 
- CACHES = {
- 'default': {
- 'BACKEND': 'django.core.cache.backends.memcached.MemcachedCache',
- 'LOCATION': 'controller:11211',
- }
- }

#### Note

Comment out any other session storage configuration.

- Enable the Identity API version 3:
- OPENSTACK\_KEYSTONE\_URL = "http://%s/identity/v3" %

## OPENSTACK\_HOST

### Note

In case your keystone run at 5000 port then you would mentioned keystone port here as well

i.e. OPENSTACK\_KEYSTONE\_URL = "<http://s:5000/identity/v3>" % OPENSTACK\_HOST

- Enable support for domains:
- OPENSTACK\_KEYSTONE\_MULTIDOMAIN\_SUPPORT = True
- Configure API versions:
- OPENSTACK\_API\_VERSIONS = {
  - "identity": 3,
  - "image": 2,
  - "volume": 3,
- }
- Configure Default as the default domain for users that you create via the dashboard:
- OPENSTACK\_KEYSTONE\_DEFAULT\_DOMAIN = "Default"
- Configure user as the default role for users that you create via the dashboard:
- OPENSTACK\_KEYSTONE\_DEFAULT\_ROLE = "user"
- If you chose networking option 1, disable support for layer-3 networking services:
- OPENSTACK\_NEUTRON\_NETWORK = {
  - ...
  - 'enable\_router': False,
  - 'enable\_quotas': False,
  - 'enable\_ipv6': False,
  - 'enable\_distributed\_router': False,
  - 'enable\_ha\_router': False,
  - 'enable\_fip\_topology\_check': False,
- }
- Optionally, configure the time zone:
- TIME\_ZONE = "TIME\_ZONE"

Replace TIME\_ZONE with an appropriate time zone identifier. For more information, see the [list of time zones](#).

4. Add the following line to /etc/apache2/conf-available/openstack-dashboard.conf if not included.
5. WSGIApplicationGroup %{GLOBAL}

### Finalize installation

- Reload the web server configuration:  
systemctl reload apache2.service

**Ex.No: 05**

Launching an instance- Register an account at openstack, Create SSH Key, validate network.

**Aim:**

**To Launch an instance-** Register an account at openstack, Create SSH Key, validate network.

**Procedure:**

Instances are virtual machines that run inside the cloud. You can launch an instance from the following sources:

- Images uploaded to the Image service.
- Image that you have copied to a persistent volume. The instance launches from the volume, which is provided by the cinder-volume API through iSCSI.
- Instance snapshot that you took.

Launch an instance

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Compute tab and click Instances category.

The dashboard shows the instances with its name, its private and floating IP addresses, size, status, task, power state, and so on.

4. Click Launch Instance.
5. In the Launch Instance dialog box, specify the following values:

Details tab

Instance Name

Assign a name to the virtual machine.

Note

The name you assign here becomes the initial host name of the server. If the name is longer than 63 characters, the Compute service truncates it automatically to ensure dnsmasq works correctly.

After the server is built, if you change the server name in the API or change the host name directly, the names are not updated in the dashboard.

Server names are not guaranteed to be unique when created so you could have two instances with the same host name.

Description

You can assign a brief description of the virtual machine.

Availability Zone

By default, this value is set to the availability zone given by the cloud provider (for example, us-west or apac-south). For some cases, it could be nova.

Count

To launch multiple instances, enter a value greater than 1. The default is 1.

Source tab

Instance Boot Source

Your options are:

Boot from image

If you choose this option, a new field for Image Name displays. You can select the image from the list.

Boot from snapshot

If you choose this option, a new field for Instance Snapshot displays. You can select the snapshot from the list.

Boot from volume

If you choose this option, a new field for Volume displays. You can select the volume from the list.

Boot from image (creates a new volume)

With this option, you can boot from an image and create a volume by entering the Device Size and Device Name for your volume. Click the Delete Volume on Instance Delete option to delete the volume on deleting the instance.

Boot from volume snapshot (creates a new volume)

Using this option, you can boot from a volume snapshot and create a new volume by choosing Volume Snapshot from a list and adding a Device Name for your volume. Click the Delete Volume on Instance Delete option to delete the volume on deleting the instance.

Image Name

This field changes based on your previous selection. If you have chosen to launch an instance using an image, the Image Name field displays. Select the image name from the dropdown list.

Instance Snapshot

This field changes based on your previous selection. If you have chosen to launch an instance using a snapshot, the Instance Snapshot field displays. Select the snapshot name from the dropdown list.

Volume

This field changes based on your previous selection. If you have chosen to launch an instance using a volume, the Volume field displays. Select the volume name from the dropdown list. If you want to delete the volume on instance delete, check the Delete Volume on Instance Delete option.

Flavor tab

Flavor

Specify the size of the instance to launch.

Note

The flavor is selected based on the size of the image selected for launching an instance. For example, while creating an image, if you have entered the value in the Minimum RAM (MB) field as 2048, then on selecting the image, the default flavor is m1.small.

Networks tab

Selected Networks

To add a network to the instance, click the + in the Available field.

Network Ports tab

Ports

Activate the ports that you want to assign to the instance.

Security Groups tab

Security Groups

Activate the security groups that you want to assign to the instance.

Security groups are a kind of cloud firewall that define which incoming network traffic is forwarded to instances.

If you have not created any security groups, you can assign only the default security group to the instance.

Key Pair tab

Key Pair

Specify a key pair.

If the image uses a static root password or a static key set (neither is recommended), you do not need to provide a key pair to launch the instance.

Configuration tab

Customization Script Source

Specify a customization script that runs after your instance launches.

Metadata tab

Available Metadata

Add Metadata items to your instance.

6. Click Launch Instance.

The instance starts on a compute node in the cloud.

Note



If you did not provide a key pair, security groups, or rules, users can access the instance only from inside the cloud through VNC. Even pinging the instance is not possible without an ICMP rule configured.

You can also launch an instance from the Images or Volumes category when you launch an instance from an image or a volume respectively.

When you launch an instance from an image, OpenStack creates a local copy of the image on the compute node where the instance starts.

For details on creating images, see [Creating images manually](#) in the *OpenStack Virtual Machine Image Guide*.

When you launch an instance from a volume, note the following steps:

- To select the volume from which to launch, launch an instance from an arbitrary image on the volume. The arbitrary image that you select does not boot. Instead, it is replaced by the image on the volume that you choose in the next steps.

To boot a Xen image from a volume, the image you launch in must be the same type, fully virtualized or paravirtualized, as the one on the volume.

- Select the volume or volume snapshot from which to boot. Enter a device name. Enter vda for KVM images or xvda for Xen images.

#### Note

When running QEMU without support for the hardware virtualization, set `cpu_mode="none"` alongside `virt_type=qemu` in `/etc/nova/nova-compute.conf` to solve the following error:

libvirtError: unsupported configuration: CPU mode 'host-model'

for ``x86\_64`` qemu domain on ``x86\_64`` host is not supported by hypervisor

Connect to your instance by using SSH

To use SSH to connect to your instance, use the downloaded keypair file.

#### Note

The user name is ubuntu for the Ubuntu cloud images on TryStack.

1. Copy the IP address for your instance.
2. Use the ssh command to make a secure connection to the instance. For example:  
2. `ssh -i MyKey.pem ubuntu@10.0.0.2`
- 3.
4. At the prompt, type yes.

It is also possible to SSH into an instance without an SSH keypair, if the administrator has enabled root password injection. For more information about root password injection, see [Injecting the administrator password](#) in the *OpenStack Administrator Guide*.

#### Track usage for instances

You can track usage for instances for each project. You can track costs per month by showing meters like number of vCPUs, disks, RAM, and uptime for all your instances.

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Compute tab and click Overview category.
4. To query the instance usage for a month, select a month and click Submit.
5. To download a summary, click Download CSV Summary.

#### Create an instance snapshot

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Compute tab and click the Instances category.
4. Select the instance from which to create a snapshot.
5. In the actions column, click Create Snapshot.
6. In the Create Snapshot dialog box, enter a name for the snapshot, and click Create Snapshot.

The Images category shows the instance snapshot.

To launch an instance from the snapshot, select the snapshot and click Launch. Proceed with launching an instance.

#### Manage an instance

1. Log in to the dashboard.
2. Select the appropriate project from the drop down menu at the top left.
3. On the Project tab, open the Compute tab and click Instances category.
4. Select an instance.
5. In the menu list in the actions column, select the state.

You can resize or rebuild an instance. You can also choose to view the instance console log, edit instance or the security groups. Depending on the current state of the instance, you can pause, resume, suspend, soft or hard reboot, or terminate it.

## Ex.No: 06

Sharing project environment among multiple users.

### Aim:

To Share project environment among multiple users.

### Procedure:

As an administrator, you manage projects, users, and roles. Projects are organizational units in the cloud to which you can assign users. Projects are also known as *projects* or *accounts*. Users can be members of one or more projects. Roles define which actions users can perform. You assign roles to user-project pairs.

You can define actions for OpenStack service roles in the `/etc/PROJECT/policy.json` files. For example, define actions for Compute service roles in the `/etc/nova/policy.json` file.

You can manage projects, users, and roles independently from each other.

During cloud set up, the operator defines at least one project, user, and role.

You can add, update, and delete projects and users, assign users to one or more projects, and change or remove the assignment. To enable or temporarily disable a project or user, update that project or user. You can also change quotas at the project level.

Before you can delete a user account, you must remove the user account from its primary project.

Before you can run client commands, you must download and source an OpenStack RC file.

See [Download and source the OpenStack RC file](#).

#### Projects

A project is a group of zero or more users. In Compute, a project owns virtual machines. In Object Storage, a project owns containers. Users can be associated with more than one project. Each project and user pairing can have a role associated with it.

#### List projects

List all projects with their ID, name, and whether they are enabled or disabled:

```
$ openstack project list
```

```
+-----+-----+
| ID                | Name                |
+-----+-----+
| f7ac731cc11f40efbc03a9f9e1d1d21f | admin              |
| c150ab41f0d9443f8874e32e725a4cc8 | alt_demo           |
| a9debfe41a6d4d09a677da737b907d5e | demo               |
| 9208739195a34c628c58c95d157917d7 | invisible_to_admin |
| 3943a53dc92a49b2827fae94363851e1 | service            |
| 80cab5e1f02045abad92a2864cfd76cb | test_project       |
+-----+-----+
```

#### Create a project

Create a project named new-project:

```
$ openstack project create --description 'my new project' new-project \
--domain default
```

```
+-----+-----+
| Field    | Value                |
+-----+-----+
| description | my new project      |
| domain_id  | e601210181f54843b51b3edff41d4980 |
| enabled    | True                 |
| id         | 1a4a0618b306462c9830f876b0bd6af2 |
| is_domain  | False                |
| name       | new-project         |
```

```
| parent_id | e601210181f54843b51b3edff41d4980 |
```

```
+-----+-----+
```

### Update a project¶

Specify the project ID to update a project. You can update the name, description, and enabled status of a project.

- To temporarily disable a project:
- \$ openstack project set PROJECT\_ID --disable
- To enable a disabled project:
- \$ openstack project set PROJECT\_ID --enable
- To update the name of a project:
- \$ openstack project set PROJECT\_ID --name project-new
- To verify your changes, show information for the updated project:
- \$ openstack project show PROJECT\_ID

```
• +-----+-----+
• | Field      | Value                               |
• +-----+-----+
• | description | my new project                     |
• | enabled     | True                               |
• | id          | 0b0b995694234521bf93c792ed44247f |
• | name        | new-project                        |
• | properties  |                                     |
• +-----+-----+
```

### Delete a project¶

Specify the project ID to delete a project:

```
$ openstack project delete PROJECT_ID
```

### Users¶

#### List users¶

List all users:

```
$ openstack user list
```

```
+-----+-----+
| ID              | Name |
+-----+-----+
| 352b37f5c89144d4ad0534139266d51f | admin |
| 86c0de739bcb4802b8dc786921355813 | demo  |
| 32ec34aae8ea432e8af560a1cec0e881 | glance |
| 7047fcb7908e420cb36e13bbd72c972c | nova  |
+-----+-----+
```

### Create a user¶

To create a user, you must specify a name. Optionally, you can specify a project ID, password, and email address. It is recommended that you include the project ID and password because the user cannot log in to the dashboard without this information.

Create the new-user user:

```
$ openstack user create --project new-project --password PASSWORD new-user
```

```
+-----+-----+
| Field      | Value                               |
+-----+-----+
| email      | None                               |
| enabled    | True                               |
| id         | 6322872d9c7e445dbbb49c1f9ca28adc |
| name       | new-user                          |
| project_id | 0b0b995694234521bf93c792ed44247f |
| username   | new-user                          |
+-----+-----+
```

### Update a user¶

You can update the name, email address, and enabled status for a user.

- To temporarily disable a user account:

- \$ openstack user set USER\_NAME --disable

If you disable a user account, the user cannot log in to the dashboard. However, data for the user account is maintained, so you can enable the user at any time.

- To enable a disabled user account:
- \$ openstack user set USER\_NAME --enable
- To change the name and description for a user account:
- \$ openstack user set USER\_NAME --name user-new --email new-user@example.com
- User has been updated.

Delete a user

Delete a specified user account:

\$ openstack user delete USER\_NAME

Roles and role assignments

List available roles

List the available roles:

\$ openstack role list

ID	Name
71ccc37d41c8491c975ae72676db687f	Member
149f50a1fe684bfa88dae76a48d26ef7	ResellerAdmin
9fe2ff9ee4384b1894a90878d3e92bab	_member_
6ecf391421604da985db2f141e46a7c8	admin
deb4fffd123c4d02a907c2c74559dccf	anotherrole

Create a role

Users can be members of multiple projects. To assign users to multiple projects, define a role and assign that role to a user-project pair.

Create the new-role role:

\$ openstack role create new-role

Field	Value
domain_id	None
id	a34425c884c74c8881496dc2c2e84ffc
name	new-role

Note

If you are using identity v3, you may need to use the --domain option with a specific domain name.

Assign a role

To assign a user to a project, you must assign the role to a user-project pair. To do this, you need the user, role, and project IDs.

1. List users and note the user ID you want to assign to the role:
2. \$ openstack user list

3. +-----+
4. | ID | Name |
5. +-----+
6. | 6ab5800949644c3e8fb86aaeab8275c8 | admin |
7. | dfc484b9094f4390b9c51aba49a6df34 | demo |
8. | 55389ff02f5e40cf85a053cc1cacb20c | alt\_demo |
9. | bc52bcfd882f4d388485451c4a29f8e0 | nova |
10. | 255388ffa6e54ec991f584cb03085e77 | glance |
11. | 48b6e6dec364428da89ba67b654fac03 | cinder |
12. | c094dd5a8e1d4010832c249d39541316 | neutron |
13. | 6322872d9c7e445dbbb49c1f9ca28adc | new-user |

```

14. +-----+-----+
15. List role IDs and note the role ID you want to assign:
16. $ openstack role list
17. +-----+-----+
18. | ID                               | Name          |
19. +-----+-----+
20. | 71ccc37d41c8491c975ae72676db687f | Member        |
21. | 149f50a1fe684bfa88dae76a48d26ef7 | ResellerAdmin |
22. | 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
23. | 6ecf391421604da985db2f141e46a7c8 | admin         |
24. | deb4fffd123c4d02a907c2c74559dccf | anotherrole   |
25. | bef1f95537914b1295da6aa038ef4de6 | new-role      |
26. +-----+-----+
27. List projects and note the project ID you want to assign to the role:
28. $ openstack project list
29. +-----+-----+
30. | ID                               | Name          |
31. +-----+-----+
32. | 0b0b995694234521bf93c792ed44247f | new-project   |
33. | 29c09e68e6f741afa952a837e29c700b | admin         |
34. | 3a7ab11d3be74d3c9df3ede538840966 | invisible_to_admin |
35. | 71a2c23bab884c609774c2db6fcee3d0 | service       |
36. | 87e48a8394e34d13afc2646bc85a0d8c | alt_demo      |
37. | fef7ae86615f4bf5a37c1196d09bcb95 | demo          |
38. +-----+-----+
39. Assign a role to a user-project pair:
40. $ openstack role add --user USER_NAME --project TENANT_ID ROLE_NAME
For example, assign the new-role role to the demo and test-project pair:
$ openstack role add --user demo --project test-project new-role
41. Verify the role assignment:
42. $ openstack role assignment list --user USER_NAME \
43. --project PROJECT_ID --names
44. +-----+-----+-----+-----+
45. | ID                               | Name          | Project | User |
46. +-----+-----+-----+-----+
47. | a34425c884c74c8881496dc2c2e84ffc | new-role      | demo    | demo |
48. | 04a7e3192c0745a2b1e3d2baf5a3ee0f | Member        | demo    | demo |
49. | 62bcf3e27eef4f648eb72d1f9920f6e5 | anotherrole   | demo    | demo |
50. +-----+-----+-----+-----+

```

#### Note

Before the Newton release, users would run the `openstack role list --user USER_NAME --project TENANT_ID` command to verify the role assignment.

View role details [¶](#)

View details for a specified role:

`$ openstack role show ROLE_NAME`

```

+-----+-----+
| Field | Value          |
+-----+-----+
| domain_id | None          |
| id        | a34425c884c74c8881496dc2c2e84ffc |
| name      | new-role      |
+-----+-----+

```

Remove a role [¶](#)

Remove a role from a user-project pair:

1. Run the `openstack role remove` command:

2. `$ openstack role remove --user USER_NAME --project TENANT_ID ROLE_NAME`
  3. Verify the role removal:
  4. `$ openstack role list --user USER_NAME --project TENANT_ID`
- If the role was removed, the command output omits the removed role.