Submitted By: Annie Jain

Sap Id: 500083967

Roll No: R214220179

Batch: B3 Hons

CLOUD APPLICATION DEVELOPMENT

OPENSTACK LAB EXPERIMENT – 02

OBJECTIVE: Configuring a Software-Defined Network (SDN) using OpenStack Neutron.

Introduction

OpenStack Networking (neutron) allows you to create and attach interface devices managed by other OpenStack services to networks. Plug-ins can be implemented to accommodate different networking equipment and software, providing flexibility to OpenStack architecture and deployment.

It includes the following components:

neutron-server

Accepts and routes API requests to the appropriate OpenStack Networking plug-in for action.

OpenStack Networking plug-ins and agents

Plug and unplug ports, create networks or subnets, and provide IP addressing. These plug-ins and agents differ depending on the vendor and technologies used in the particular cloud. OpenStack Networking ships with plug-ins and agents for Cisco virtual and physical switches, NEC OpenFlow products, Open vSwitch, Linux bridging, Open Virtual Network (OVN) and the VMware NSX product.

The common agents are L3 (layer 3), DHCP (dynamic host IP addressing), and a plug-in agent.

Messaging queue

Used by most OpenStack Networking installations to route information between the neutron-server and various agents. Also acts as a database to store networking state for particular plug-ins.

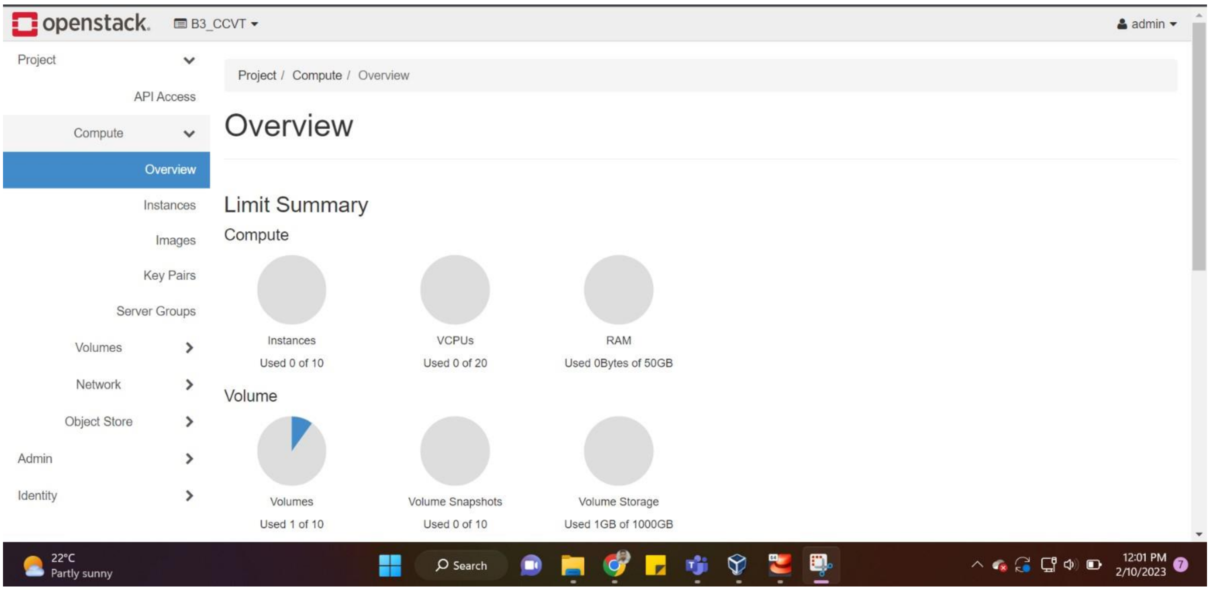
OpenStack Networking mainly interacts with OpenStack Compute to provide networks and connectivity for its instances.

The purpose of this lab was to configure a Software-Defined Network (SDN) using OpenStack Neutron. An SDN is a network architecture that separates the control plane from the data plane, allowing for centralized control of network traffic. OpenStack Neutron is an open-source networking project that provides network services to OpenStack clouds. By integrating an SDN controller with OpenStack Neutron, we can manage network traffic more efficiently and dynamically.

Methods:

To configure an SDN using OpenStack Neutron, we followed the following steps:

1. Installed and configured OpenStack Neutron on our system.
2. Chose an SDN controller to use (we chose OpenDaylight).
3. Installed and configured OpenDaylight.
4. Configured OpenStack Neutron to use OpenDaylight.
5. Configured the network topology in OpenDaylight.
6. Created and configured an OpenStack Neutron network.
7. Associated the OpenStack Neutron network with the appropriate network topology in OpenDaylight.
8. Tested our configuration to ensure traffic was flowing correctly.



Project / Compute / Overview

Overview

Limit Summary

Compute

Instances
Used 0 of 10

VCPUs
Used 0 of 20

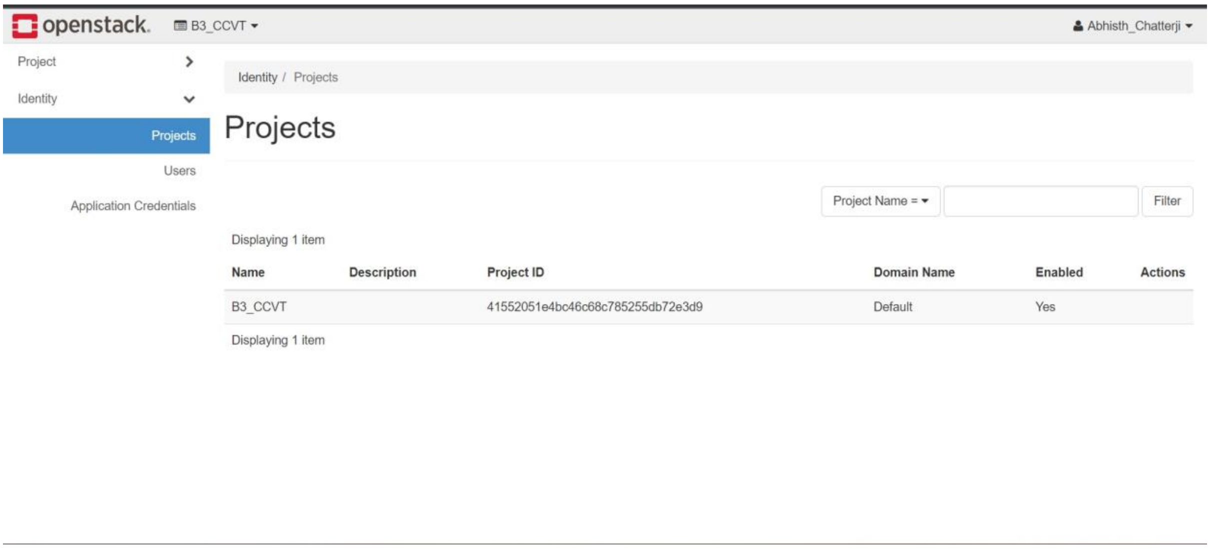
RAM
Used 0Bytes of 50GB

Volume

Volumes
Used 1 of 10

Volume Snapshots
Used 0 of 10

Volume Storage
Used 1GB of 1000GB



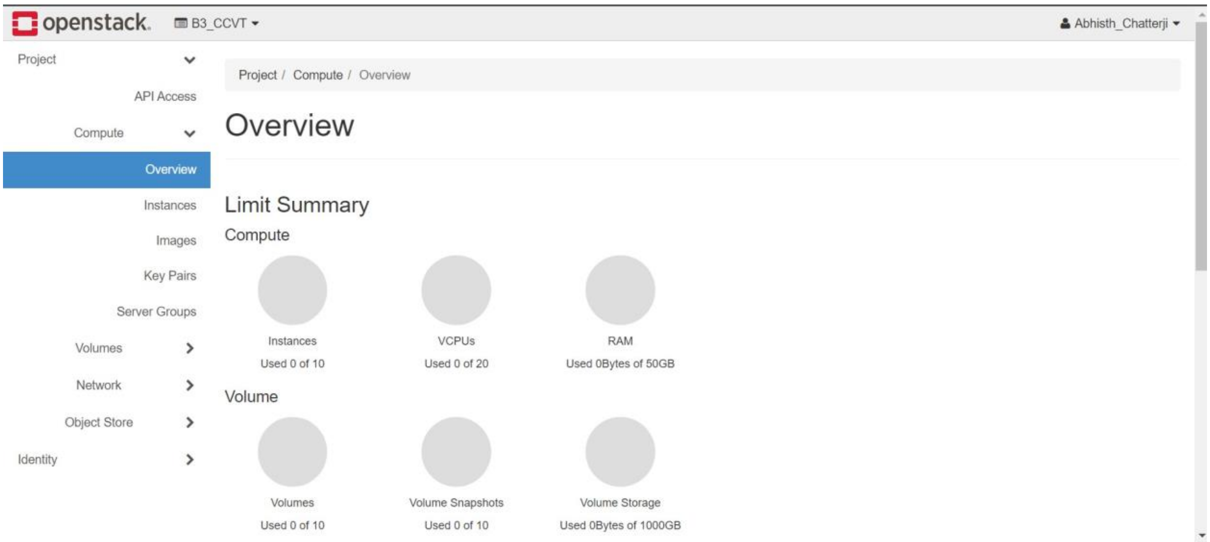
Identity / Projects

Projects

Displaying 1 item

Name	Description	Project ID	Domain Name	Enabled	Actions
B3_CCVT		41552051e4bc46c68c785255db72e3d9	Default	Yes	

Displaying 1 item



Project / Compute / Overview

Overview

Limit Summary

Compute

Instances
Used 0 of 10

VCPUs
Used 0 of 20

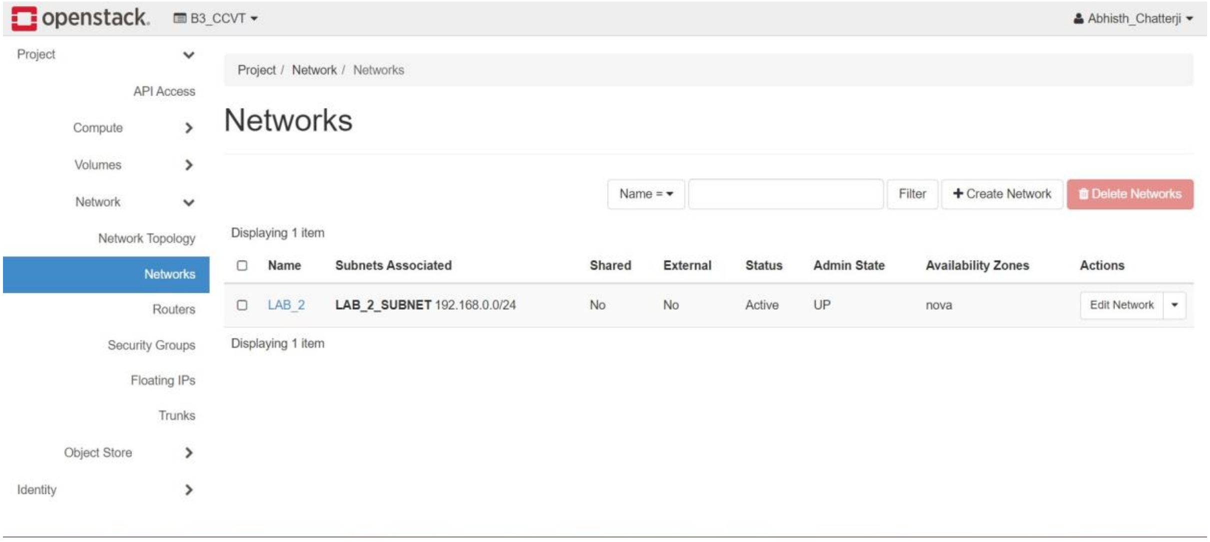
RAM
Used 0Bytes of 50GB

Volume

Volumes
Used 0 of 10

Volume Snapshots
Used 0 of 10

Volume Storage
Used 0Bytes of 1000GB

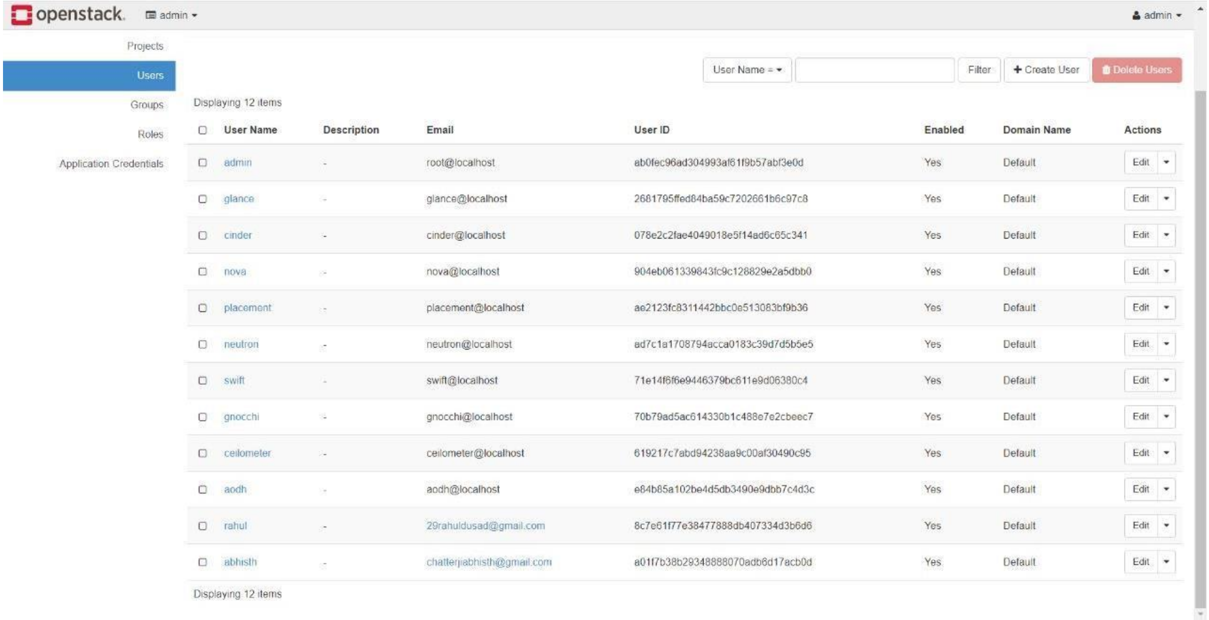


Networks

Displaying 1 item

□	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
□	LAB_2	**LAB_2_SUBNET** 192.168.0.0/24	No	No	Active	UP	nova	Edit Network ▾

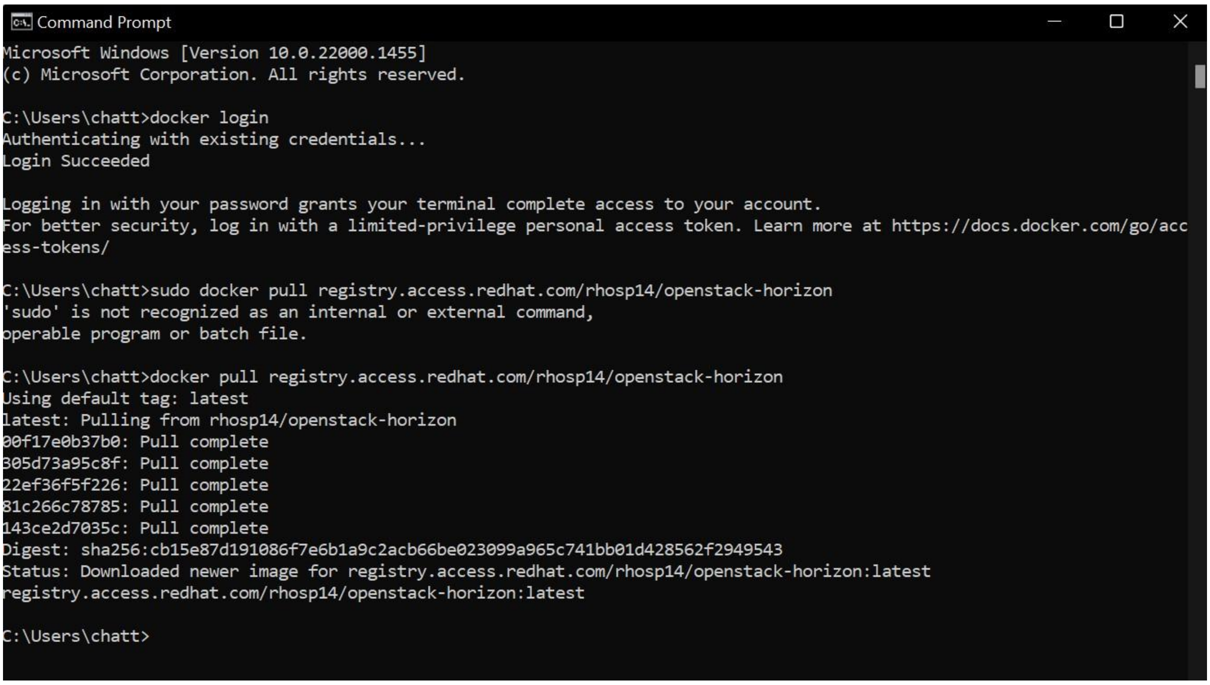
Displaying 1 item



Displaying 12 items

□	User Name	Description	Email	User ID	Enabled	Domain Name	Actions
□	admin	-	root@localhost	ab0fec96ad304993af61f9b57abf3e0d	Yes	Default	Edit ▾
□	glance	-	glance@localhost	2681795ffed84ba59c7202661b6c97c8	Yes	Default	Edit ▾
□	cinder	-	cinder@localhost	078e2c2fae4049018e5f14ad6c65c341	Yes	Default	Edit ▾
□	nova	-	nova@localhost	904eb061339843fc9c128829e2a5dbb0	Yes	Default	Edit ▾
□	placement	-	placement@localhost	ae2123fc8311442bbc0e513083bf9b36	Yes	Default	Edit ▾
□	neutron	-	neutron@localhost	ad7c1a1708794acca0183c39d7d5b5e5	Yes	Default	Edit ▾
□	swift	-	swift@localhost	71e14f6f6e9446379bc611e9d06380c4	Yes	Default	Edit ▾
□	gnocchi	-	gnocchi@localhost	70b79ad5ac614330b1c488e7e2cbeec7	Yes	Default	Edit ▾
□	ceilometer	-	ceilometer@localhost	619217c7abd94238aa9c00af30490c95	Yes	Default	Edit ▾
□	aodh	-	aodh@localhost	e84b85a102be4d5db3490e9dbb7c4d3c	Yes	Default	Edit ▾
□	rahul	-	29rahuldusad@gmail.com	8c7e61f77e38477888db407334d3b6d6	Yes	Default	Edit ▾
□	abhisth	-	chatterjiabhisth@gmail.com	a01f7b38b29348888070adb6d17acb0d	Yes	Default	Edit ▾

Displaying 12 items



```
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chatt>docker login
Authenticating with existing credentials...
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/acc
ess-tokens/

C:\Users\chatt>sudo docker pull registry.access.redhat.com/rhosp14/openstack-horizon
'sudo' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\chatt>docker pull registry.access.redhat.com/rhosp14/openstack-horizon
Using default tag: latest
latest: Pulling from rhosp14/openstack-horizon
00f17e0b37b0: Pull complete
305d73a95c8f: Pull complete
22ef36f5f226: Pull complete
81c266c78785: Pull complete
143ce2d7035c: Pull complete
Digest: sha256:cb15e87d191086f7e6b1a9c2acb66be023099a965c741bb01d428562f2949543
Status: Downloaded newer image for registry.access.redhat.com/rhosp14/openstack-horizon:latest
registry.access.redhat.com/rhosp14/openstack-horizon:latest

C:\Users\chatt>
```

Results:

We were able to successfully configure a Software-Defined Network (SDN) using OpenStack Neutron and OpenDaylight. We created a network topology in OpenDaylight that included virtual switches, routers, and other network devices. We then created an OpenStack Neutron network and associated it with the appropriate network topology in OpenDaylight. We tested our configuration by sending traffic between virtual machines on the OpenStack Neutron network and physical machines on the physical network infrastructure. We observed that traffic was flowing correctly and that we were able to centrally manage the network traffic using OpenDaylight.

Conclusion:

In conclusion, we were able to successfully configure a Software-Defined Network (SDN) using OpenStack Neutron and OpenDaylight. By integrating an SDN controller with OpenStack Neutron, we were able to manage network traffic more efficiently and dynamically. This lab provided us with a better understanding of how SDNs can be used to manage network traffic and the benefits that they provide.