Name: Clarence B. Chavez	**Date Performed:** Dec. 6, 2024
Course/Section: CPE31S2	**Date Submitted:** Dec. 6, 2024
Instructor: Engr. Robin Valenzuela	**Semester and SY:** 1st Sem, 2024-2025

Activity 15: OpenStack Installation (Neutron, Horizon, Cinder)

1. Objectives

Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).

2. Intended Learning Outcomes

1. Analyze the advantages and disadvantages of cloud services

2. Evaluate different Cloud deployment and service models

3. Create a workflow to install and configure OpenStack base services using Ansible as documentation and execution.

3. Resources

Oracle VirtualBox (Hypervisor)

1x Ubuntu VM or Centos VM

4. Tasks

1. Create a new repository for this activity.
2. Create a playbook that converts the steps in the following items in https://docs.openstack.org/install-guide/
   a. Neutron
   b. Horizon
   c. Cinder

   d. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file.

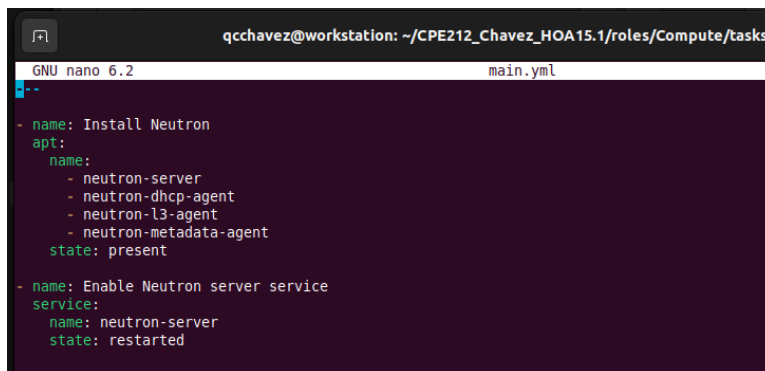   e. Add, commit and push it to your GitHub repo.

5. Output (screenshots and explanations)
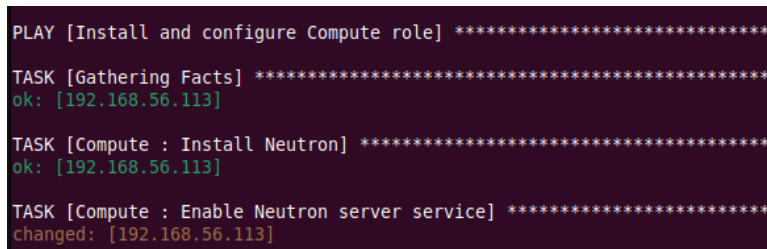
Task 4.1
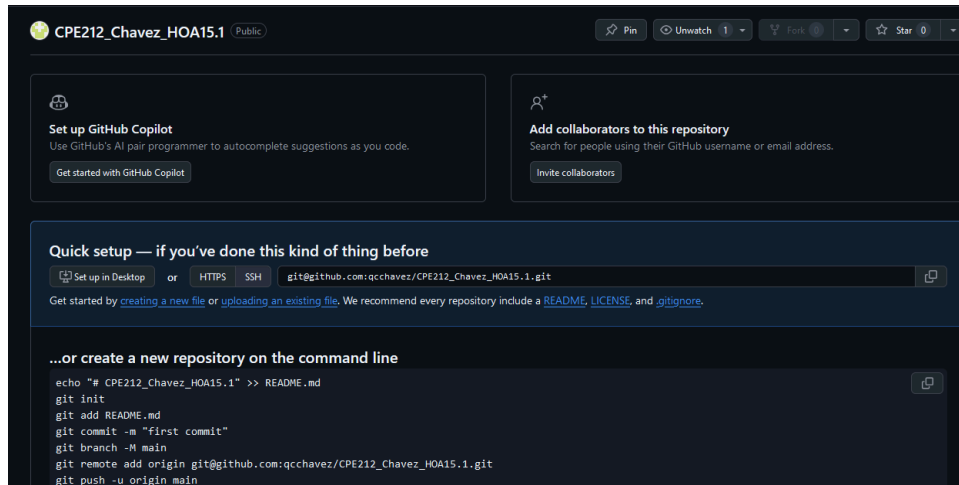
- In this screenshot, I've created a GitHub repository for my HOA 15.1

Task 4.2a (Neutron)

- In this screenshot, I've created the codes in installing Neutron–server and its necessary packages, and also, I've enabled the neutron server service.

- In this screenshot, it shows that the codes for the Compute role or the Neutron service is working.

```
qcchavez@server2:~$ systemctl status neutron-server
● neutron-server.service - OpenStack Neutron Server
     Loaded: loaded (/lib/systemd/system/neutron-server.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:26:00 CST; 962ms ago
       Docs: man:neutron-server(1)
   Main PID: 8059 (neutron-server)
      Tasks: 1 (limit: 3471)
     Memory: 21.7M
        CPU: 295ms
     CGroup: /system.slice/neutron-server.service
             └─8059 /usr/bin/python3 /usr/bin/neutron-server --config-file=/etc/neutron/neutron.conf --config-file=/etc/neutron/p>
```

- In this screenshot, it shows that the **neutron–server** service is active and running.

```
qcchavez@server2:~$ systemctl status neutron-dhcp-agent
● neutron-dhcp-agent.service - OpenStack Neutron DHCP agent
     Loaded: loaded (/lib/systemd/system/neutron-dhcp-agent.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:17:19 CST; 8min ago
       Docs: man:neutron-dhcp-agent(1)
   Main PID: 922 (neutron-dhcp-ag)
      Tasks: 1 (limit: 3471)
     Memory: 8.4M
        CPU: 2.529s
     CGroup: /system.slice/neutron-dhcp-agent.service
             └─922 "neutron-dhcp-agent (/usr/bin/python3 /usr/bin/neutron-dhcp-agent --config-file=/etc/neutron/neutron.conf --co>
```

- In this screenshot, it shows that the **neutron–dhcp–agent** service is active and running.

```
qcchavez@server2:~$ systemctl status neutron-l3-agent
● neutron-l3-agent.service - OpenStack Neutron L3 agent
     Loaded: loaded (/lib/systemd/system/neutron-l3-agent.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:26:06 CST; 3s ago
       Docs: man:neutron-l3-agent(1)
   Main PID: 8091 (neutron-l3-agen)
      Tasks: 1 (limit: 3471)
     Memory: 66.6M
        CPU: 1.238s
     CGroup: /system.slice/neutron-l3-agent.service
             └─8091 /usr/bin/python3 /usr/bin/neutron-l3-agent --config-file=/etc/neutron/neutron.conf --config-file=/etc/neutron>
```

- In this screenshot, it shows that the **neutron–l3–agent** service is active and running.

```
qcchavez@server2:~$ systemctl status neutron-metadata-agent
● neutron-metadata-agent.service - OpenStack Neutron Metadata Agent
     Loaded: loaded (/lib/systemd/system/neutron-metadata-agent.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:17:19 CST; 8min ago
       Docs: man:neutron-metadata-agent(1)
   Main PID: 926 (neutron-metadat)
      Tasks: 2 (limit: 3471)
     Memory: 9.0M
        CPU: 2.619s
     CGroup: /system.slice/neutron-metadata-agent.service
             ├─ 926 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --config-file=/etc/neutron/neutron.>
             └─1510 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --config-file=/etc/neutron/neutron.>
Warning: some journal files were not opened due to insufficient permissions.
```

- In this screenshot, it shows that the **neutron–metadata–agent** service is active and running.

Task 4.2b (Horizon)

```
qcchavez@workstation: ~/CPE212_Chavez_HOA15.1/roles/Controller/tasks

GNU nano 6.2                                          main.yml *
---
- name: Install OpenStack Dashboard
  apt:
    name: openstack-dashboard
    state: present

- name: Configure the Horizon dashboard
  lineinfile:
    path: /etc/openstack-dashboard/local_settings.py
    regexp: '^OPENSTACK_HOST ='
    line: 'OPENSTACK_HOST = "192.168.56.113"'

- name: Ensure the secret key is set
  lineinfile:
    path: /etc/openstack-dashboard/local_settings.py
    regexp: '^SECRET_KEY ='
    line: 'SECRET_KEY = "123456"'

- name: Enable required Apache modules
  command: a2enmod "{{ item }}"
  with_items:
    - wsgi
    - headers

- name: Restart Apache
  systemd:
    name: apache2
    state: restarted
    enabled: yes
```

- In this screenshot, it shows the necessary codes for installing the Horizon dashboard, it includes the configuration, the secret key, and the required modules, and also, I've restarted the apache incase it is already running.

```
PLAY [Install and configure Controller role] ****************************

TASK [Gathering Facts] *************************************************
ok: [192.168.56.113]

TASK [Controller : Install OpenStack Dashboard] ***********************
ok: [192.168.56.113]

TASK [Controller : Configure the Horizon dashboard] ******************
ok: [192.168.56.113]

TASK [Controller : Ensure the secret key is set] *********************
ok: [192.168.56.113]

TASK [Controller : Enable required Apache modules] *******************
changed: [192.168.56.113] => (item=wsgi)
changed: [192.168.56.113] => (item=headers)

TASK [Controller : Restart Apache] **********************************
changed: [192.168.56.113]
```
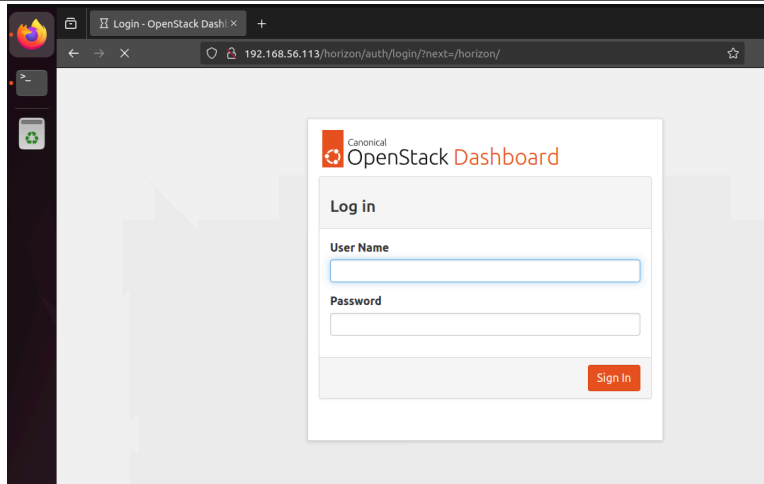
- In this screenshot, it shows that the codes that were made for the Controller role or the Horizon dashboard are working properly.

- In this screenshot, it shows that the Horizon Dashboard is working properly as I can prompt it through the web browser.

Task 4.2c (Cinder)

- In this screenshot, it shows the necessary codes for Installing Cinder, creating its service file, and also, enabling it.

```
PLAY [Install and configure BlockStorage role] *******************

TASK [Gathering Facts] *******************************************
ok: [192.168.56.113]

TASK [BlockStorage : Installing Cinder] **************************
ok: [192.168.56.113]

TASK [BlockStorage : Create Cinder API service file] ************
ok: [192.168.56.113]

TASK [BlockStorage : Enable Cinder API service] ****************
changed: [192.168.56.113]
```

- In this screenshot, it shows that the codes for BlockStorage role or the Cinder service are successfully done.

```
qcchavez@server2:~$ systemctl status cinder-api
● cinder-api.service - OpenStack Cinder API Service
     Loaded: loaded (/etc/systemd/system/cinder-api.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:27:27 CST; 3s ago
   Main PID: 8438 (cinder-api)
      Tasks: 1 (limit: 3471)
     Memory: 69.2M
        CPU: 1.029s
     CGroup: /system.slice/cinder-api.service
             └─8438 /usr/bin/python3 /usr/bin/cinder-api
qcchavez@server2:~$ systemctl status cinder-scheduler
● cinder-scheduler.service - OpenStack Cinder Scheduler
     Loaded: loaded (/lib/systemd/system/cinder-scheduler.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-12-06 16:27:34 CST; 2s ago
       Docs: man:cinder-scheduler(1)
   Main PID: 8476 (cinder-schedule)
      Tasks: 1 (limit: 3471)
     Memory: 41.9M
        CPU: 680ms
     CGroup: /system.slice/cinder-scheduler.service
             └─8476 /usr/bin/python3 /usr/bin/cinder-scheduler --config-file=/etc/cinder/cinder.conf --log-file=/var/log/cinder/c>

qcchavez@server2:~$ systemctl status cinder-volume
● cinder-volume.service - OpenStack Cinder Volume
     Loaded: loaded (/lib/systemd/system/cinder-volume.service; enabled; vendor preset: enabled)
    Drop-In: /usr/lib/systemd/system/cinder-volume.service.d
             └─cinder-volume.service.conf
     Active: active (running) since Fri 2024-12-06 16:27:38 CST; 4s ago
       Docs: man:cinder-volume(1)
   Main PID: 8492 (cinder-volume)
      Tasks: 1 (limit: 3471)
     Memory: 82.8M
        CPU: 1.335s
     CGroup: /system.slice/cinder-volume.service
             └─8492 /usr/bin/python3 /usr/bin/cinder-volume --config-file=/etc/cinder/cinder.conf --log-file=/var/log/cinder/cind>

qcchavez@server2:~$
```

- In this screenshot, it shows that the **cinder-api, cinder-scheduler, and cinder-volume** are running actively.

Task 4.2d

```
qcchavez@workstation:~/CPE212_Chavez_HOA15.1$ tree
.
├── ansible.cfg
├── install_openstack.yml
├── inventory
├── README.md
└── roles
    ├── BlockStorage
    │   └── tasks
    │       └── main.yml
    ├── Compute
    │   └── tasks
    │       └── main.yml
    └── Controller
        └── tasks
            └── main.yml

7 directories, 7 files
```

- In this screenshot, it shows the file content of the whole repository in tree view.

Task 4.2e

```
qcchavez@workstation:~/CPE212_Chavez_HOA15.1$ git add .
qcchavez@workstation:~/CPE212_Chavez_HOA15.1$ git commit -m "This is the file for HOA15.1"
[main (root-commit) bd54ab9] This is the file for HOA15.1
 7 files changed, 131 insertions(+)
 create mode 100644 README.md
 create mode 100644 ansible.cfg
 create mode 100644 install_openstack.yml
 create mode 100644 inventory
 create mode 100644 roles/BlockStorage/tasks/main.yml
 create mode 100644 roles/Compute/tasks/main.yml
 create mode 100644 roles/Controller/tasks/main.yml
qcchavez@workstation:~/CPE212_Chavez_HOA15.1$ git push origin main
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (16/16), 1.97 KiB | 504.00 KiB/s, done.
Total 16 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qcchavez/CPE212_Chavez_HOA15.1.git
 * [new branch]      main -> main
qcchavez@workstation:~/CPE212_Chavez_HOA15.1$
```

CPE212_Chavez_HOA15.1 Public		Pin	Unwatch 1

main	1 Branch	0 Tags	Go to file	Add file	<> Code

qcchavez This is the file for HOA15.1		bd54ab9 · now	1 Commit
roles	This is the file for HOA15.1		now
README.md	This is the file for HOA15.1		now
ansible.cfg	This is the file for HOA15.1		now
install_openstack.yml	This is the file for HOA15.1		now
inventory	This is the file for HOA15.1		now

- In this screenshot, this is the proof that I've added, committed, and pushed to Github.

Reflections:

Answer the following:

1. Describe Neutron, Horizon and Cinder services

Neutron

- A service that is responsible for providing functionality between the interface devices like Virtual NICs (Network Interface Card) in a virtual machine. It allows the users to define networks, subnets, and routers, providing connectivity between instances and external networks.

Horizon

- A service that is responsible for providing the users and administrators with a GUI to interact with OpenStack services. Users can manage compute, storage, networking resources, and perform such administrative tasks like creating users, and assigning its roles.

Cinder

- A service that is responsible for providing persistent storage for instances, as it allows users to create, attach, and detach block storage volumes to virtual machines.

Conclusions:

- In this activity, I was able to implement the required OpenStack services (Neutron, Horizon, and Cinder). They demonstrate the power and flexibility of cloud computing in managing networks, storage, and seamless usage with GUIs. I've learned the importance of understanding the dependencies of the services, managing configurations, and proper testing to ensure that the services will work properly.