Name: Gabriel Soriano	Date Performed: December 08, 2022
Course/Section: CPE31S23	Date Submitted: December 09, 2022
Instructor: Engr. Taylar	Semester and SY: 1st sem - SY 2022-2023

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)

```
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ ls
ansible.cfg  get-pip.py  get-pip.py.1  inventory  nhcS.yml  README.md
qgsoriano1@cloudshell:~/HOA15.1_Soriano$
```

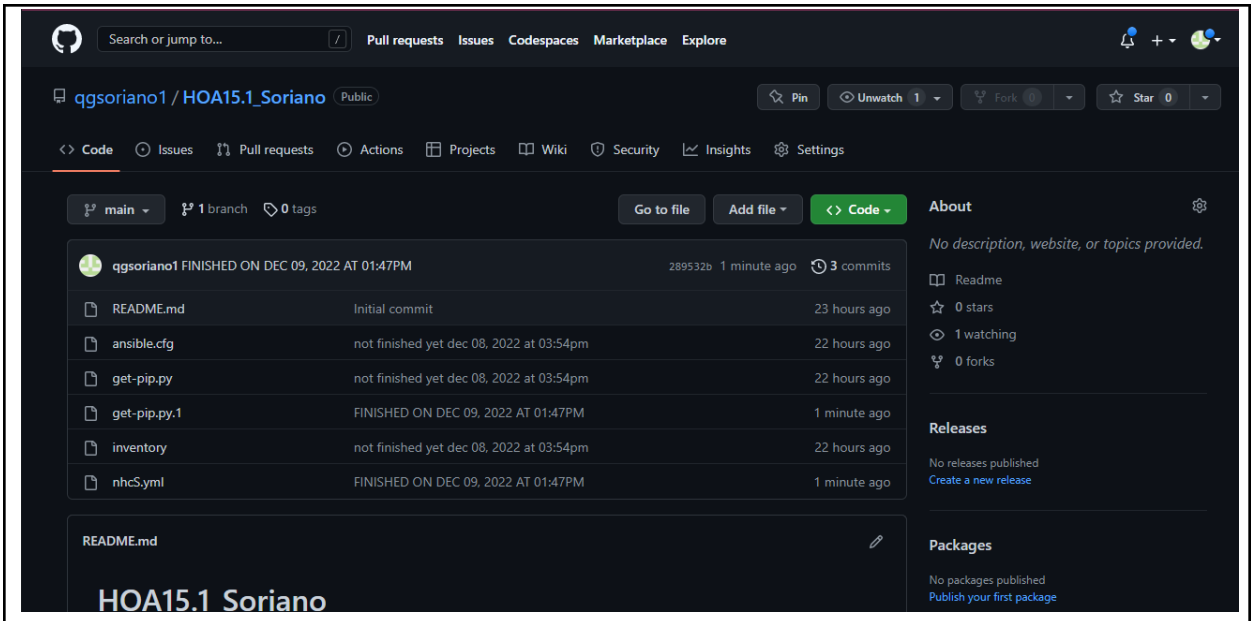
- **This shows the creation of a new repository in github and a successful git cloning, and making it as a directory in the cli.**

```
PLAY RECAP *********************************************************************************************************************
localhost                  : ok=3    changed=0    unreachable=0    failed=1    skipped=2    rescued=0    ignored=0

qgsoriano1@cloudshell:~/HOA15.1_Soriano$ nano nhcS.yml
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ ansible-playbook --ask-become-pass nhcS.yml
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning:
 Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be r
emoved in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:

PLAY [all] *********************************************************************************************************************

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

TASK [Update repository index (CentOS)] ****************************************************************************************
skipping: [localhost]

TASK [Install updates (Ubuntu)] ************************************************************************************************
skipping: [localhost]

TASK [Install neutron] *********************************************************************************************************
ok: [localhost]

TASK [Install horizon] *********************************************************************************************************
ok: [localhost]

TASK [Install cinder] **********************************************************************************************************
changed: [localhost]

PLAY RECAP *********************************************************************************************************************
localhost                  : ok=4    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0

qgsoriano1@cloudshell:~/HOA15.1_Soriano$
```

- **This shows the successful run of the playbook file for the trial of achieving all of the requirements. Turns out that this is successful. There were a lot of errors encountered while running this trial, and all of these errors were overcomed.**

```
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   nhcS.yml

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        get-pip.py.1

no changes added to commit (use "git add" and/or "git commit -a")
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ git add *
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ git commit -m "FINISHED ON DEC 09, 2022 AT 01:47PM"
[main 289532b] FINISHED ON DEC 09, 2022 AT 01:47PM
 2 files changed, 32208 insertions(+), 1 deletion(-)
 create mode 100644 get-pip.py.1
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ git push origin mian
error: src refspec mian does not match any
error: failed to push some refs to 'github.com:qgsoriano1/HOA15.1_Soriano.git'
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 350 bytes | 350.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:qgsoriano1/HOA15.1_Soriano.git
   c352fbe..289532b  main -> main
qgsoriano1@cloudshell:~/HOA15.1_Soriano$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
qgsoriano1@cloudshell:~/HOA15.1_Soriano$
```

- **This shows the successful git add, committing and pushing of the files created and edited for the success of this activity.**



- **This shows the proof of the successful uploading of all of the edited files from the cli to the github repository.**

Answer the following:

1. Describe Neutron, Horizon and Cinder services
   - The OpenStack Networking service (neutron) provides an API that allows users to build rich networking topologies, set up and define network connectivity, configure advanced network policies and address in the cloud. Horizon is a web-based graphical interface that cloud administrators and users can access to manage OpenStack compute, storage and networking services. Cinder virtualizes the management of block storage devices and provides end users with a self service API to request and consume those resources without requiring any knowledge of where their storage is actually deployed or on what type of device.

Conclusions:

While performing this activity, it was not that hard to perform because there are two other more activities that we have done before this. It was kind of like the same thing, but just different targets. As usual, there are also errors encountered during this activity, specifically those library errors. They are also solved by the same set of

commands, and they are still effective. As long as this activity concerns, all of the requirements are quite achieved.