Name: Jozshua Amiel Alonzo	**Date Performed:** December 4, 2022
Course/Section: CPE31S23	**Date Submitted:**
Instructor: Dr. Jonathan Taylar	**Semester and SY:** 2022-2023

Activity 13: OpenStack Prerequisite Installation

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

Github Link:

https://github.com/jozshua/HO

A13_Alonzo.git

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. NTP
    b. OpenStack packages
    c. SQL Database
    d. Message Queue
    e. Memcached
    f. Etcd
    g. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in Inventory file.
    h. Add, commit and push it to your GitHub repo.

5. Output (screenshots and explanations)

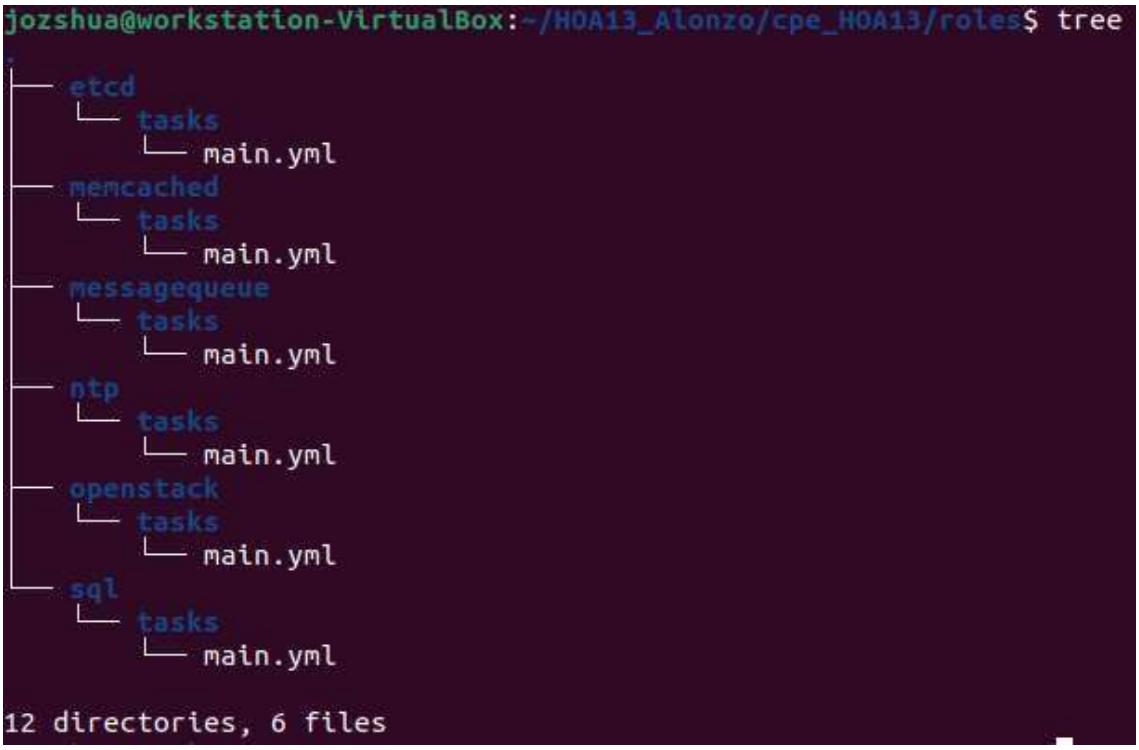
Created the new repository for this activity.

```
  GNU nano 6.2                                inventory
[controller]
192.168.56.102

[compute]
192.168.56.102
```

```
  GNU nano 6.2                                site.yml
---
- hosts: all
  become: true
  pre_tasks:

  - name: install updates (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

- hosts: controller
  become: true
  roles:
    - ntp
    - openstack
    - sql

- hosts: compute
  become: true
  roles:
    - messagequeue
    - memcached
    - etcd
```

Create site.yml file and put the different server type with the different packages for installation.

Creating roles for different installation packages:



I created the six different directories and created the tasks and main.yml file in each.

For NTP

```
  GNU nano 6.2                                  main.yml
- name: Install NTP (Ubuntu)
  apt:
    name:
      - chrony
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Edit chrony file (Ubuntu)
  copy:
    dest: /etc/chrony/chrony.conf
    content: |
      server NTP_SERVER iburst
      allow 10.0.0.0/24
    mode: 0755
  when: ansible_distribution == "Ubuntu"

- name: Enable NTP service (Ubuntu)
  service:
    name: chrony
    enabled: yes
  when: ansible_distribution == "Ubuntu"
```

```
- name: Restart NTP service
  service:
    name: chrony
    state: restarted
  when: ansible_distribution == "Ubuntu"
```

For OpenStack:

```
  GNU nano 6.2                                  main.yml
- name: Install OpenStack (Ubuntu)
  apt:
    name:
      - nova-compute
      - python3-openstackclient
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"
```

For SQL:

```
  GNU nano 6.2                                main.yml
- name: Installation SQL Database (Ubuntu)
  apt:
    name:
      - mariadb-server
      - python3-pymysql
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"
- name: mariadb file (Ubuntu)
  copy:
    dest: /etc/mysql/mariadb.conf.d/99-openstack.cnf
    content: |
      [mysqld]
      bind-address = 127.0.0.1
      default-storage-engine = innodb
      innodb_file_per_table = on
      max_connections = 4096
      collation-server = utf8_general_ci
      character-set-server = utf8
    mode: 0755
  when: ansible_distribution == "Ubuntu"
```

```
- name: Stop database service (Ubuntu)
  service:
    name: mysql
    state: stopped
  when: ansible_distribution == "Ubuntu"

- name: Start database service (Ubuntu)
  service:
    name: mysql
    state: started
  when: ansible_distribution == "Ubuntu"

- name: Enable database service
  service:
    name: mysql
    enabled: yes
  when: ansible_distribution == "Ubuntu"
```

For messagequeue:

```
  GNU nano 6.2                                main.yml
- name: Install message queue (Ubuntu)
  apt:
    name:
      - rabbitmq-server
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Add openstack user
  community.rabbitmq.rabbitmq_user:
    user: openstack
    password: RABIT_PASS
    vhost: /
    configure_priv: .*
    read_priv: .*
    write_priv: .*
    state: present
  when: ansible_distribution == "Ubuntu"
```

For memcached:

```
  GNU nano 6.2                                main.yml
- name: Install memcached (Ubuntu)
  apt:
    name:
      - memcached
      - python3-memcache
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Edit memcached file (Ubuntu)
  copy:
    dest: /etc/memcached.conf
    content: |
      memcached_user = memcache
      memcached_port = 11211
      memcached_listen_ip = 127.0.0.1
      memcached_connections = 1024
      memcached_log_file = /var/log/memcached.log
      memcached_log_verbosity = ""
      memcached_max_item_size = 1m
      memcached_threads = 4
    mode: 0644
    owner: root
    group: root
  when: ansible_distribution == "Ubuntu"
```

```
- name: Stop memcached (Ubuntu)
  service:
    name: memcached
    state: stopped
  when: ansible_distribution == "Ubuntu"

- name: Start memcached (Ubuntu)
  service:
    name: memcached
    state: started
  when: ansible_distribution == "Ubuntu"

- name: Enable memcached (Ubuntu)
  service:
    name: memcached
    enabled: yes
  when: ansible_distribution == "Ubuntu"
```

For etcd:

```
  GNU nano 6.2                                  main.yml
- name: Install etcd (Ubuntu)
  apt:
    name:
      - etcd
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Edit the etcd file (Ubuntu)
  copy:
    dest: /etc/default/ectd
    content: |
      ETCD_NAME="controller"
      ETCD_DATA_DIR="/var/lib/etcd"
      ETCD_INITIAL_CLUSTER_STATE="new"
      ETCD_INITIAL_CLUSTER_TOKEN="etcd-cluster-01"
      ETCD_INITIAL_CLUSTER="controller=http://127.0.0.1:2380"
      ETCD_INITIAL_ADVERTISE_PEER_URLS="http://127.0.0.1:2380"
      ETCD_ADVERTISE_CLIENT_URLS="http://127.0.0.1:2379"
      ETCD_LISTEN_PEER_URLS="http://127.0.0.1:2379"
    mode: 0755
  when: ansible_distribution == "Ubuntu"
```

```
- name: Stop etcd service (Ubuntu)
  service:
    name: etcd
    state: stopped
  when: ansible_distribution == "Ubuntu"

- name: Start etcd service (Ubuntu)
  service:
    name: etcd
    state: started
  when: ansible_distribution == "Ubuntu"

- name: Enable etcd service
  service:
    name: etcd
    enabled: yes
  when: ansible_distribution == "Ubuntu"
```

Executing the playbook:

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]

TASK [install updates (Ubuntu)] ************************************************
*
ok: [192.168.56.102]

PLAY [controller] **************************************************************
*

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]

TASK [ntp : Install NTP (Ubuntu)] **********************************************
*
ok: [192.168.56.102]

TASK [ntp : Edit chrony file (Ubuntu)] *****************************************
*
ok: [192.168.56.102]
```

```
TASK [ntp : Enable NTP service (Ubuntu)] ***************************************
*
ok: [192.168.56.102]

TASK [ntp : Restart NTP service] ***********************************************
*
changed: [192.168.56.102]

TASK [openstack : Install OpenStack (Ubuntu)] **********************************
*
ok: [192.168.56.102]

TASK [sql : Installation SQL Database (Ubuntu)] ********************************
*
ok: [192.168.56.102]

TASK [sql : mariadb file (Ubuntu)] *********************************************
*
ok: [192.168.56.102]

TASK [sql : Stop database service (Ubuntu)] ************************************
*
changed: [192.168.56.102]

TASK [sql : Start database service (Ubuntu)] ***********************************
*
changed: [192.168.56.102]
```

```
TASK [sql : Enable database service] ***************************************
*
ok: [192.168.56.102]

PLAY [compute] *************************************************************
*

TASK [Gathering Facts] *****************************************************
*
ok: [192.168.56.102]

TASK [messagequeue : Install message queue (Ubuntu)] ***********************
*
ok: [192.168.56.102]

TASK [messagequeue : Add openstack user] ***********************************
*
ok: [192.168.56.102]

TASK [memcached : Install memcached (Ubuntu)] ******************************
*
ok: [192.168.56.102]

TASK [memcached : Edit memcached file (Ubuntu)] ****************************
*
ok: [192.168.56.102]
```

```
TASK [memcached : Stop memcached (Ubuntu)] *********************************
*
changed: [192.168.56.102]

TASK [memcached : Start memcached (Ubuntu)] ********************************
*
changed: [192.168.56.102]

TASK [memcached : Enable memcached (Ubuntu)] *******************************
*
ok: [192.168.56.102]

TASK [etcd : Install etcd (Ubuntu)] ****************************************
*
changed: [192.168.56.102]

TASK [etcd : Edit the etcd file (Ubuntu)] **********************************
*
changed: [192.168.56.102]

TASK [etcd : Stop etcd service (Ubuntu)] ***********************************
*
changed: [192.168.56.102]

TASK [etcd : Start etcd service (Ubuntu)] **********************************
*
changed: [192.168.56.102]
```

```
TASK [etcd : Enable etcd service] ******************************************
*
ok: [192.168.56.102]

PLAY RECAP *****************************************************************
*
192.168.56.102             : ok=26   changed=9    unreachable=0    failed=0
skipped=0    rescued=0    ignored=0

jozshua@workstation-VirtualBox:~/HOA13_Alonzo/cpe_HOA13$
```

From executing this playbook there are a total of 26 successfully executed tasks for the two different plays and 9 changed states. This playbook shows the installation of the six different packages from the different files.

Proof of installations:

```
jozshua@server2-VirtualBox:~$ systemctl status mysqld.service
● mariadb.service - MariaDB 10.6.11 database server
     Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor pres>
     Active: active (running) since Tue 2022-12-06 17:19:38 PST; 12min ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 797 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var>
    Process: 813 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_ST>
    Process: 843 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && >
    Process: 1439 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_>
    Process: 1441 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0>
   Main PID: 959 (mariadbd)
```

```
jozshua@server2-VirtualBox:~$ systemctl status chrony.service
● chrony.service - chrony, an NTP client/server
     Loaded: loaded (/lib/systemd/system/chrony.service; enabled; vendor prese>
     Active: active (running) since Tue 2022-12-06 17:19:10 PST; 26min ago
       Docs: man:chronyd(8)
             man:chronyc(1)
             man:chrony.conf(5)
   Main PID: 835 (chronyd)
      Tasks: 2 (limit: 1075)
     Memory: 1.0M
        CPU: 250ms
     CGroup: /system.slice/chrony.service
             ├─835 /usr/sbin/chronyd -F 1
             └─855 /usr/sbin/chronyd -F 1
```

```
jozshua@server2-VirtualBox:~$ systemctl status nova-compute.service
● nova-compute.service - OpenStack Compute
     Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor>
     Active: active (running) since Tue 2022-12-06 17:21:03 PST; 28min ago
   Main PID: 1977 (nova-compute)
      Tasks: 2 (limit: 1075)
     Memory: 46.4M
        CPU: 14.552s
     CGroup: /system.slice/nova-compute.service
             └─1977 /usr/bin/python3 /usr/bin/nova-compute --config-file=/etc/>
```

```
jozshua@server2-VirtualBox:~$ systemctl status rabbitmq-server.service
● rabbitmq-server.service - RabbitMQ Messaging Server
     Loaded: loaded (/lib/systemd/system/rabbitmq-server.service; enabled; ven>
     Active: active (running) since Tue 2022-12-06 17:21:03 PST; 30min ago
   Main PID: 800 (beam.smp)
      Tasks: 21 (limit: 1075)
     Memory: 33.1M
        CPU: 47.672s
     CGroup: /system.slice/rabbitmq-server.service
             ├─ 800 /usr/lib/erlang/erts-12.2.1/bin/beam.smp -W w -MBas ageffc>
             ├─ 969 erl_child_setup 65536
             ├─1528 inet_gethost 4
             └─1529 inet_gethost 4
```

```
jozshua@server2-VirtualBox:~$ systemctl status memcached.service
● memcached.service - memcached daemon
     Loaded: loaded (/lib/systemd/system/memcached.service; enabled; vendor pr>
     Active: active (running) since Tue 2022-12-06 17:19:08 PST; 33min ago
       Docs: man:memcached(1)
   Main PID: 798 (memcached)
      Tasks: 10 (limit: 1075)
     Memory: 1.1M
        CPU: 1.001s
     CGroup: /system.slice/memcached.service
             └─798 /usr/bin/memcached -u root
```

```
jozshua@server2-VirtualBox:~$ systemctl status etcd.service
● etcd.service - etcd - highly-available key value store
     Loaded: loaded (/lib/systemd/system/etcd.service; enabled; vendor preset:>
     Active: active (running) since Tue 2022-12-06 17:19:30 PST; 33min ago
       Docs: https://etcd.io/docs
             man:etcd
   Main PID: 796 (etcd)
      Tasks: 7 (limit: 1075)
     Memory: 9.5M
        CPU: 14.407s
     CGroup: /system.slice/etcd.service
             └─796 /usr/bin/etcd
```

The codes that we used for this activity was already committed and pushed to my github account.

Reflections:

Answer the following:

1. What are the benefits of implementing OpenStack?

The benefits of having the implementation of the OpenStack is that has a few prerequisites are needed before you can use it. Also, you just only needed an internet then you might find yourself using it on any kind of device. It also have high and reliability security so your personal information was safe. It has also a wide range access if you are accessing from the different location.

Conclusions:

In this activity, I created the new repository to commit and pushed the codes. So for the installation first I created inventory,ansible.cfg and site.yml files for the playbook. Next I created roles with 6 different directories and it has main.yml file in each for the packages. Next is I run the playbook and it was successfully executed the two play with the two different nodes. Also, I assure that the 6 different packages were successfully installed by checking their status on the terminal of the Ubuntu server. Therefore I am confidently finished this activity.