

<b>Name:</b> Kazuki A. Ogata	<b>Date Performed:</b> December 16, 2023
<b>Course/Section:</b> CPE 232 - CPE31S5	<b>Date Submitted:</b> December 16, 2023
<b>Instructor:</b> Engr. Roman Richard	<b>Semester and SY:</b> 1st semester S.Y 2023-2024

### 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

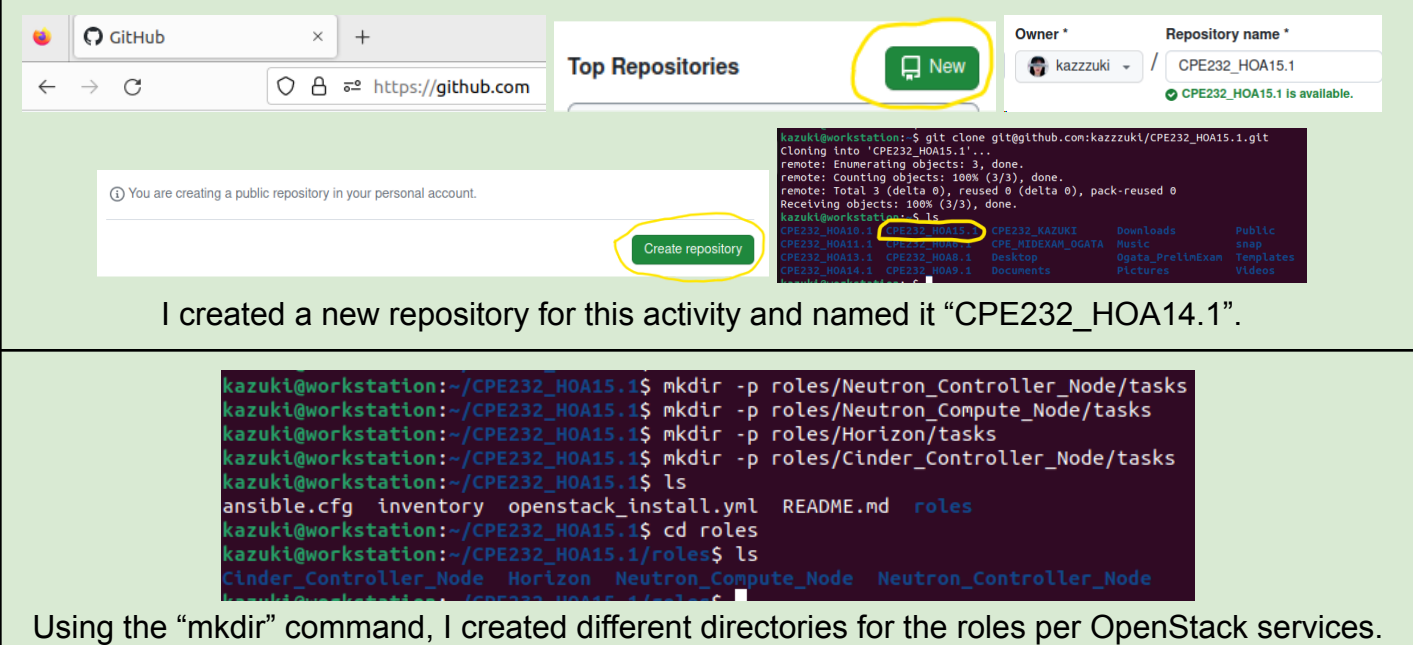
Oracle VirtualBox (Hypervisor)

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
3. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file.
4. Add, commit and push it to your GitHub repo.

#### 5. Output (screenshots and explanations)



The screenshot shows a GitHub web interface where a new repository named "CPE232\_HOA15.1" is being created. The "Create repository" button is highlighted. Below this, a terminal window displays the command to clone the repository and the subsequent directory structure created using the 'mkdir' command. The terminal output shows the following commands and results:

```
kazuki@workstation:~$ git clone git@github.com:kazzzuki/CPE232_HOA15.1.git
Cloning into 'CPE232_HOA15.1'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
kazuki@workstation:~$ ls
CPE232_HOA15.1  CPE232_HOAZ15.1  CPE232_KAZUKI  Downloads  Public
CPE232_HOA11.1  CPE232_HOAZ11.1  CPE_HIDEXAM_OGATA  Music  snap
CPE232_HOA13.1  CPE232_HOAZ13.1  Desktop  Ogata_PrelimExam  Templates
CPE232_HOA14.1  CPE232_HOAZ14.1  Documents  Pictures  Videos
```

I created a new repository for this activity and named it "CPE232\_HOA14.1".

```
kazuki@workstation:~/CPE232_HOA15.1$ mkdir -p roles/Neutron_Controller_Node/tasks
kazuki@workstation:~/CPE232_HOA15.1$ mkdir -p roles/Neutron_Compute_Node/tasks
kazuki@workstation:~/CPE232_HOA15.1$ mkdir -p roles/Horizon/tasks
kazuki@workstation:~/CPE232_HOA15.1$ mkdir -p roles/Cinder_Controller_Node/tasks
kazuki@workstation:~/CPE232_HOA15.1$ ls
ansible.cfg  inventory  openstack_install.yml  README.md  roles
kazuki@workstation:~/CPE232_HOA15.1$ cd roles
kazuki@workstation:~/CPE232_HOA15.1/roles$ ls
Cinder_Controller_Node  Horizon  Neutron_Compute_Node  Neutron_Controller_Node
```

Using the "mkdir" command, I created different directories for the roles per OpenStack services.

```
kazuki@workstation: ~/CPE232_HOA15.1
kazuki@workstation:~/CPE232_HOA15.1$
kazuki@workstation:~/CPE232_HOA15.1$ cat /home/kazuki/CPE232_HOA15.1/roles/Neutron_Controller_Node/tasks/main.yml

- name: Create Neutron database
  mysql_db:
    name: neutron
    login_unix_socket: /var/run/mysqld/mysqld.sock

- name: Grant privileges on Neutron database
  mysql_user:
    name: neutron
    password: NEUTRON_PASS
    priv: "neutron.*:ALL"
    state: present
    host: "{{ item }}"
    login_unix_socket: /var/run/mysqld/mysqld.sock
  loop:
    - localhost
    - "%"

- name: Source the admin credentials
  shell: ". /home/kazuki/admin-openrc"

- name: Create Neutron user
  command: "openstack user create --domain default --password-prompt neutron "
  environment:
    OS_USERNAME: admin
    OS_PASSWORD: NEUTRON_PASS
    OS_PROJECT_NAME: admin
    OS_AUTH_URL: http://192.168.56.131/v3
  async: 3600
  poll: 0

- name: Add admin role to neutron user
  command: "openstack role add --project service --user neutron admin"
  environment:
    OS_USERNAME: admin
    OS_PASSWORD: NEUTRON_PASS
    OS_PROJECT_NAME: admin
    OS_AUTH_URL: http://192.168.56.131/v3
  async: 3600
  poll: 0

- name: Create neutron service entity
  command: 'openstack service create --name neutron --description "OpenStack Networking" network '
  async: 3600
  poll: 0

- name: Create networking service API endpoints (public)
  command: 'openstack endpoint create --region RegionOne network public http://controller:9696 '
  async: 3600
  poll: 0

- name: Create network service API endpoints (internal)
  command: 'openstack endpoint create --region RegionOne network internal http://controller:9696 '
  async: 3600
  poll: 0

- name: Create network service API endpoints (admin)
  command: 'openstack endpoint create --region RegionOne network admin http://controller:9696 '
  async: 3600
  poll: 0

- name: Install network components
  apt:
    name: "{{ item }}"
    state: present
  loop:
    - neutron-server
    - neutron-plugin-ml2
    - neutron-linuxbridge-agent
    - neutron-dhcp-agent
    - neutron-metadata-agent
```

```

- name: Edit /etc/neutron/neutron.conf file
  lineinfile:
    path: /etc/neutron/neutron.conf
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^connection =', line: 'connection = mysql+pymysql://neutron:NEUTRON_DBPASS@controller/neutron' }
    - { regexp: '^core_plugin =', line: 'core_plugin = ml2' }
    - { regexp: '^service_plugins =', line: 'service_plugins =' }
    - { regexp: '^transport_url =', line: 'transport_url = rabbit://openstack:RABBIT_PASS@controller' }
    - { regexp: '^auth_strategy =', line: 'auth_strategy = keystone' }
    - { regexp: '^www_authenticate_uri =', line: 'www_authenticate_uri = http://controller:5000' }
    - { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
    - { regexp: '^memcached_servers =', line: 'memcached_servers = controller:11211' }
    - { regexp: '^auth_type =', line: 'auth_type = password' }
    - { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
    - { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
    - { regexp: '^project_name =', line: 'project_name = service' }
    - { regexp: '^username =', line: 'username = neutron' }
    - { regexp: '^password =', line: 'password = NEUTRON_PASS' }
    - { regexp: '^notify_nova_on_port_status_changes =', line: 'notify_nova_on_port_status_changes = true' }
    - { regexp: '^notify_nova_on_port_data_changes =', line: 'notify_nova_on_port_data_changes = true' }
    - { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
    - { regexp: '^auth_type =', line: 'auth_type = password' }
    - { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
    - { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
    - { regexp: '^region_name =', line: 'region_name = RegionOne' }
    - { regexp: '^project_name =', line: 'project_name = service' }
    - { regexp: '^username =', line: 'username = nova' }
    - { regexp: '^password =', line: 'password = NOVA_PASS' }
    - { regexp: '^lock_path =', line: 'lock_path = /var/lib/neutron/tmp' }

```

```

- name: Edit /etc/neutron/plugins/ml2/ml2_conf.ini file
  lineinfile:
    path: /etc/neutron/plugins/ml2/ml2_conf.ini
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^type_drivers =', line: 'type_drivers = flat,vlan' }
    - { regexp: '^tenant_network_types =', line: 'tenant_network_types =' }
    - { regexp: '^mechanism_drivers =', line: 'mechanism_drivers = linuxbridge' }
    - { regexp: '^extension_drivers =', line: 'extension_drivers = port_security' }
    - { regexp: '^flat_networks =', line: 'flat_networks = provider' }
    - { regexp: '^enable_ipset =', line: 'enable_ipset = true' }

- name: Edit /etc/neutron/plugins/ml2/linuxbridge_agent.ini file
  lineinfile:
    path: /etc/neutron/plugins/ml2/linuxbridge_agent.ini
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^physical_interface_mappings =', line: 'physical_interface_mappings = provider:PROVIDER_INTERFACE_NAME' }
    - { regexp: '^enable_vxlan =', line: 'enable_vxlan = false' }
    - { regexp: '^enable_security_group =', line: 'enable_security_group = true' }
    - { regexp: '^firewall_driver =', line: 'firewall_driver = neutron.agent.linux.iptables_firewall.IptablesFirewallDriver' }

- name: Edit /etc/neutron/dhcp_agent.ini file
  ansible.builtin.lineinfile:
    path: /etc/neutron/dhcp_agent.ini
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^interface_driver =', line: 'interface_driver = linuxbridge' }
    - { regexp: '^dhcp_driver =', line: 'dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq' }
    - { regexp: '^enable_isolated_metadata =', line: 'enable_isolated_metadata = true' }

```

```

- name: Edit /etc/neutron/metadata_agent.ini file
  lineinfile:
    path: /etc/neutron/metadata_agent.ini
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^nova_metadata_host =', line: 'nova_metadata_host = controller' }
    - { regexp: '^metadata_proxy_shared_secret =', line: 'metadata_proxy_shared_secret = METADATA_SECRET' }

```

```

- name: Edit /etc/nova/nova.conf file
  lineinfile:
    path: /etc/nova/nova.conf
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
    - { regexp: '^auth_type =', line: 'auth_type = password' }
    - { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
    - { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
    - { regexp: '^region_name =', line: 'region_name = RegionOne' }
    - { regexp: '^project_name =', line: 'project_name = service' }
    - { regexp: '^username =', line: 'username = neutron' }
    - { regexp: '^password =', line: 'password = NEUTRON_PASS' }
    - { regexp: '^service_metadata_proxy =', line: 'service_metadata_proxy = true' }
    - { regexp: '^metadata_proxy_shared_secret =', line: 'metadata_proxy_shared_secret = METADATA_SECRET' }

- name: Populate the Neutron database
  command: "su -s /bin/sh -c 'neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_conf.ini upgrade head' neutron"
  async: 3600
  poll: 0

- name: Restart Compute API service
  service:
    name: nova-api
    state: restarted

- name: Restart Networking services
  service:
    name: "{{ item }}"
    state: restarted
  loop:
    - neutron-server
    - neutron-linuxbridge-agent
    - neutron-dhcp-agent
    - neutron-metadata-agent

```

This is my playbook for the controller node of Neutron Node. I followed the steps on the given document of OpenStack. I started by creating a database for neutron, granted privileges on it, source the admin, configure the neutron services, edit the files, installed, network components, and lastly, restarting different services like neutron-server, neutron-linuxbridge-agent, neutron-dhcp-agent, and neutron-metadata-agent.

```

kazuki@workstation: ~/CPE232_HOA15.1
kazuki@workstation:~/CPE232_HOA15.1$
kazuki@workstation:~/CPE232_HOA15.1$
kazuki@workstation:~/CPE232_HOA15.1$ cat /home/kazuki/CPE232_HOA15.1/roles/Neutron_Compute_Node/tasks/main.yml

- name: Install components
  apt:
    name: neutron-linuxbridge-agent
    state: present

- name: Edit /etc/neutron/neutron.conf file
  lineinfile:
    path: /etc/neutron/neutron.conf
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^connection =', line: '# connection =' }
    - { regexp: '^transport_url =', line: 'transport_url = rabbit://openstack:RABBIT_PASS@controller' }
    - { regexp: '^auth_strategy =', line: 'auth_strategy = keystone' }
    - { regexp: '^www_authenticate_uri =', line: 'www_authenticate_uri = http://controller:5000' }
    - { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
    - { regexp: '^memcached_servers =', line: 'memcached_servers = controller:11211' }
    - { regexp: '^auth_type =', line: 'auth_type = password' }
    - { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
    - { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
    - { regexp: '^project_name =', line: 'project_name = service' }
    - { regexp: '^username =', line: 'username = neutron' }
    - { regexp: '^password =', line: 'password = NEUTRON_PASS' }
    - { regexp: '^lock_path =', line: 'lock_path = /var/lib/neutron/tmp' }

- name: Edit /etc/neutron/plugins/ml2/linuxbridge_agent.ini file
  lineinfile:
    path: /etc/neutron/plugins/ml2/linuxbridge_agent.ini
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^physical_interface_mappings =', line: 'physical_interface_mappings = provider:PROVIDER_INTERFACE_NAME' }
    - { regexp: '^enable_vxlan =', line: 'enable_vxlan = false' }
    - { regexp: '^enable_security_group =', line: 'enable_security_group = true' }
    - { regexp: '^firewall_driver =', line: 'firewall_driver = neutron.agent.linux.iptables_firewall.IptablesFirewallDriver' }
    - { regexp: '^net.bridge.bridge-nf-call-iptables =', line: 'net.bridge.bridge-nf-call-iptables = 1' }
    - { regexp: '^net.bridge.bridge-nf-call-ip6tables =', line: 'net.bridge.bridge-nf-call-ip6tables = 1' }

- name: Edit /etc/nova/nova.conf file
  lineinfile:
    path: /etc/nova/nova.conf
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
    - { regexp: '^auth_type =', line: 'auth_type = password' }
    - { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
    - { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
    - { regexp: '^region_name =', line: 'region_name = RegionOne' }
    - { regexp: '^project_name =', line: 'project_name = service' }
    - { regexp: '^username =', line: 'username = neutron' }
    - { regexp: '^password =', line: 'password = NEUTRON_PASS' }

- name: Restart compute service
  service:
    name: nova-compute
    state: restarted

- name: Restart linux bridge agent
  service:
    name: neutron-linuxbridge-agent
    state: restarted

```

This is my playbook for the compute node of Neutron service. I followed every step on the given document of OpenStack including the network configuration. I started by installing a component, editing files, and then restarting the compute service and linux bridge agent service,

```

kazuki@workstation: ~/CPE232_HOA15.1
kazuki@workstation:~/CPE232_HOA15.1$
kazuki@workstation:~/CPE232_HOA15.1$ cat /home/kazuki/CPE232_HOA15.1/roles/Horizon/tasks/main.yml

- name: Install component
  apt:
    name: openstack-dashboard
    state: present

- name: Edit /etc/openstack-dashboard/local_settings.py file
  lineinfile:
    path: /etc/openstack-dashboard/local_settings.py
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^OPENSTACK_HOST = ', line: 'OPENSTACK_HOST = "controller"' }
    - { regexp: '^ALLOWED_HOSTS = ', line: 'ALLOWED_HOSTS = ["one.example.com", "two.example.com"]' }
    - { regexp: '^SESSION_ENGINE = ', line: 'SESSION_ENGINE = "django.contrib.sessions.backends.cache"' }
    - { regexp: '^CACHES = ', line: 'CACHES = {"default": {"BACKEND": "django.core.cache.backends.memcached.MemcachedCache", "LOCATIO
N": "controller:11211"}}' }
    - { regexp: '^OPENSTACK_KEYSTONE_URL = ', line: 'OPENSTACK_KEYSTONE_URL = "http://s/identity/v3" % OPENSTACK_HOST' }
    - { regexp: '^OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = ', line: 'OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True' }
    - { regexp: '^OPENSTACK_API_VERSIONS = ', line: 'OPENSTACK_API_VERSIONS = {"identity": 3, "image": 2, "volume": 3}' }
    - { regexp: '^OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = ', line: 'OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = "Default"' }
    - { regexp: '^OPENSTACK_KEYSTONE_DEFAULT_ROLE = ', line: 'OPENSTACK_KEYSTONE_DEFAULT_ROLE = "user"' }
    - { regexp: '^OPENSTACK_NEUTRON_NETWORK = ', line: '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}' }
    - { regexp: '^TIME_ZONE = ', line: 'TIME_ZONE = "UTC"' }

- name: Edit /etc/apache2/conf-available/openstack-dashboard.conf file
  lineinfile:
    path: /etc/apache2/conf-available/openstack-dashboard.conf
    line: 'WSGIApplicationGroup %{GLOBAL}'
    insertafter: '^<VirtualHost'

- name: Reload apache2
  systemd:
    name: apache2.service
    state: reloaded

```

This is my playbook for the installation and configuration of Horizon service. I followed the steps provided on the given document of OpenStack, it is not too long compared to other services for this activity. This playbook only consists of installation of the component, editing the file, and reloading apache2 service.

```
kazuki@workstation:~/CPE232_HOA15.1$
```

```
kazuki@workstation:~/CPE232_HOA15.1$ cat /home/kazuki/CPE232_HOA15.1/roles/Cinder_Controller_Node/tasks/main.yml
```

```
name: Create Cinder database
mysql_db:
  name: cinder
  login_unix_socket: /var/run/mysqld/mysqld.sock
```

```
name: Grant privileges on Cinder database
mysql_user:
  name: cinder
  password: CINDER_PASS
  priv: "cinder.*:ALL"
  state: present
  host: "{{ item }}"
  login_unix_socket: /var/run/mysqld/mysqld.sock
loop:
  - localhost
  - "%"

```

```
name: Source the admin credentials
shell: ". /home/kazuki/admin-openrc"
```

```
name: Create Cinder user
command: " openstack user create --domain default --password-prompt cinder "
environment:
  OS_USERNAME: admin
  OS_PASSWORD: CINDER_PASS
  OS_PROJECT_NAME: admin
  OS_AUTH_URL: http://192.168.56.131/v3
async: 3600
poll: 0
```

```
- name: Add admin role to cinder user
  command: "openstack role add --project service --user cinder admin"
  environment:
    OS_USERNAME: admin
    OS_PASSWORD: CINDER_PASS
    OS_PROJECT_NAME: admin
    OS_AUTH_URL: http://192.168.56.131/v3
  async: 3600
  poll: 0
```

```
- name: Create cinderv3 service entity
  command: ' openstack service create --name cinderv3 --description "OpenStack Block Storage" volumev3 '
  async: 3600
  poll: 0
```

```
- name: Create Block Storage service API endpoints (public)
  command: ' openstack endpoint create --region RegionOne volumev3 public http://controller:8776/v3/%(project_id)s '
  async: 3600
  poll: 0
```

```
- name: Create Block Storage service API endpoints (internal)
  command: ' openstack endpoint create --region RegionOne network internal http://controller:8776/v3/%(project_id)s '
  async: 3600
  poll: 0
```

```
- name: Create Block Storage service API endpoints (admin)
  command: ' openstack endpoint create --region RegionOne network admin http://controller:8776/v3/%(project_id)s '
  async: 3600
  poll: 0
```

```
- name: Install packages
apt:
  name:
    - cinder-api
    - cinder-scheduler
  state: present
```

```
- name: Edit cinder.conf file
  ansible.builtin.lineinfile:
    path: /etc/cinder/cinder.conf
    regexp: "{{ item.regexp }}"
    line: "{{ item.line }}"
  loop:
    - { regexp: '^connection =', line: 'connection = mysql+pymysql://cinder:CINDER_DBPASS@controller/cinder' }
    - { regexp: '^transport url =', line: 'transport url = rabbit://openstack:RABBIT_PASS@controller' }
```



```

- { regexp: '^transport_url =', line: 'transport_url = rabbit://openstack:KABBIT_PASS@controller' }
- { regexp: '^auth_strategy =', line: 'auth_strategy = keystone' }
- { regexp: '^www_authenticate_uri =', line: 'www_authenticate_uri = http://controller:5000' }
- { regexp: '^auth_url =', line: 'auth_url = http://controller:5000' }
- { regexp: '^memcached_servers =', line: 'memcached_servers = controller:11211' }
- { regexp: '^project_domain_name =', line: 'project_domain_name = default' }
- { regexp: '^user_domain_name =', line: 'user_domain_name = default' }
- { regexp: '^project_name =', line: 'project_name = service' }
- { regexp: '^username =', line: 'username = cinder' }
- { regexp: '^password =', line: 'password = CINDER_PASS' }
- { regexp: '^my_ip =', line: 'my_ip = 10.0.0.11' }
- { regexp: '^lock_path =', line: 'lock_path = /var/lib/cinder/tmp' }

- name: Populate the Neutron database
  command: ' su -s /bin/sh -c "cinder-manage db sync" cinder '
  async: 3600
  poll: 0

- name: Edit /etc/nova/nova.conf file
  lineinfile:
    path: /etc/nova/nova.conf
    insertafter: '^#\[DEFAULT\]'
    line: 'os_region_name = RegionOne'

- name: Restart compute service
  service:
    name: nova-api
    state: restarted

- name: Restart block storage services
  service:
    name: cinder-scheduler
    state: restarted

- name: Restart block storage services
  service:
    name: apache2
    state: restarted

```

This is my playbook for the installation and configuration of Cinder service. I followed every steps on the given document of OpenStack.

```
[Neutron_Controller_Node]
```

```
192.168.56.131  ansible_connection=ssh
```

```
[Neutron_Compute_Node]
```

```
192.168.56.131  ansible_connection=ssh
```

```
[Horizon]
```

```
192.168.56.131  ansible_connection=ssh
```

```
[Cinder_Controller_Node]
```

```
192.168.56.131  ansible_connection=ssh
```

```

---
- hosts: all
  become: true
  tasks:
    - name: Update
      apt:
        update_cache: yes
        state: present

- hosts: Neutron_Controller_Node
  become: true
  roles:
    - Neutron_Controller_Node

- hosts: Neutron_Compute_Node
  become: true
  roles:
    - Neutron_Compute_Node

- hosts: Horizon
  become: true
  roles:
    - Horizon

- hosts: Cinder_Controller_Node
  become: true
  roles:
    - Cinder_Controller_Node

```

On the given document of OpenStack, there is a controller and compute node for Neutron service so I created separated groups in my inventory. The image on the right is my main playbook, it consists of 1 task which is for updating the target host, and 4 roles for Neutron's controller and compute nodes, Horizon, and Cinder.



```

kazuki@workstation: ~/CPE232_HOA15.1
kazuki@workstation:~/CPE232_HOA15.1$ ansible-playbook --ask-become-pass openstack_install.yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [192.168.56.131]

TASK [Update] *****
changed: [192.168.56.131]

PLAY [Neutron_Controller_Node] *****

TASK [Gathering Facts] *****
ok: [192.168.56.131]

TASK [Neutron_Controller_Node : Create Neutron database] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Grant privileges on Neutron database] *****
changed: [192.168.56.131] => (item=localhost)
changed: [192.168.56.131] => (item=%)

TASK [Neutron_Controller_Node : Source the admin credentials] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Create Neutron user] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Add admin role to neutron user] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Create neutron service entity] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Create networking service API endpoints (public)] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Create network service API endpoints (internal)] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Install network components] *****
ok: [192.168.56.131] => (item=neutron-server)
ok: [192.168.56.131] => (item=neutron-plugin-ml2)
changed: [192.168.56.131] => (item=neutron-linuxbridge-agent)
changed: [192.168.56.131] => (item=neutron-dhcp-agent)
ok: [192.168.56.131] => (item=neutron-metadata-agent)

TASK [Neutron_Controller_Node : Edit /etc/neutron/neutron.conf file] *****
changed: [192.168.56.131] => (item={'regex': '^connection =', 'line': 'connection = mysql+pymysql://neutron:NEUTRON_DBPASS@controller/neutron'})
ok: [192.168.56.131] => (item={'regex': '^core_plugin =', 'line': 'core_plugin = ml2'})
changed: [192.168.56.131] => (item={'regex': '^service_plugins =', 'line': 'service_plugins ='})
changed: [192.168.56.131] => (item={'regex': '^transport_url =', 'line': 'transport_url = rabbit://openstack:RABBIT_PASS@controller'})
changed: [192.168.56.131] => (item={'regex': '^auth_strategy =', 'line': 'auth_strategy = keystone'})
changed: [192.168.56.131] => (item={'regex': '^www_authenticate_url =', 'line': 'www_authenticate_url = http://controller:5000'})
changed: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
changed: [192.168.56.131] => (item={'regex': '^memcached_servers =', 'line': 'memcached_servers = controller:11211'})
changed: [192.168.56.131] => (item={'regex': '^auth_type =', 'line': 'auth_type = password'})
changed: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
changed: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = neutron'})
changed: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = NEUTRON_PASS'})
changed: [192.168.56.131] => (item={'regex': '^notify_nova_on_port_status_changes =', 'line': 'notify_nova_on_port_status_changes = true'})
changed: [192.168.56.131] => (item={'regex': '^notify_nova_on_port_data_changes =', 'line': 'notify_nova_on_port_data_changes = true'})
ok: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
ok: [192.168.56.131] => (item={'regex': '^auth_type =', 'line': 'auth_type = password'})
ok: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^region_name =', 'line': 'region_name = RegionOne'})

ok: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
changed: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = nova'})
changed: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = NOVA_PASS'})
changed: [192.168.56.131] => (item={'regex': '^lock_path =', 'line': 'lock_path = /var/lib/neutron/tmp'})

```

```
TASK [Neutron_Controller_Node : Edit /etc/neutron/plugins/ml2/ml2_conf.ini file] *****
changed: [192.168.56.131] => (item={'regex': '^type_drivers =', 'line': 'type_drivers = flat,vlan'})
changed: [192.168.56.131] => (item={'regex': '^tenant_network_types =', 'line': 'tenant_network_types ='})
changed: [192.168.56.131] => (item={'regex': '^mechanism_drivers =', 'line': 'mechanism_drivers = linuxbridge'})
changed: [192.168.56.131] => (item={'regex': '^extension_drivers =', 'line': 'extension_drivers = port_security'})
changed: [192.168.56.131] => (item={'regex': '^flat_networks =', 'line': 'flat_networks = provider'})
changed: [192.168.56.131] => (item={'regex': '^enable_ipset =', 'line': 'enable_ipset = true'})

TASK [Neutron_Controller_Node : Edit /etc/neutron/plugins/ml2/linuxbridge_agent.ini file] *****
changed: [192.168.56.131] => (item={'regex': '^physical_interface_mappings =', 'line': 'physical_interface_mappings = provider:PROVIDER_INTERFACE_NAME'})
changed: [192.168.56.131] => (item={'regex': '^enable_vxlan =', 'line': 'enable_vxlan = false'})
changed: [192.168.56.131] => (item={'regex': '^enable_security_group =', 'line': 'enable_security_group = true'})
changed: [192.168.56.131] => (item={'regex': '^firewall_driver =', 'line': 'firewall_driver = neutron.agent.linux.iptables_firewall.IptablesFirewallDriver'})

TASK [Neutron_Controller_Node : Edit /etc/neutron/dhcp_agent.ini file] *****
changed: [192.168.56.131] => (item={'regex': '^interface_driver =', 'line': 'interface_driver = linuxbridge'})
changed: [192.168.56.131] => (item={'regex': '^dhcp_driver =', 'line': 'dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq'})
changed: [192.168.56.131] => (item={'regex': '^enable_isolated_metadata =', 'line': 'enable_isolated_metadata = true'})

TASK [Neutron_Controller_Node : Edit /etc/neutron/metadata_agent.ini file] *****
changed: [192.168.56.131] => (item={'regex': '^nova_metadata_host =', 'line': 'nova_metadata_host = controller'})
changed: [192.168.56.131] => (item={'regex': '^metadata_proxy_shared_secret =', 'line': 'metadata_proxy_shared_secret = METADATA_SECRET'})

TASK [Neutron_Controller_Node : Edit /etc/nova/nova.conf file] *****
changed: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
ok: [192.168.56.131] => (item={'regex': '^auth_type =', 'line': 'auth_type = password'})
changed: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^region_name =', 'line': 'region_name = RegionOne'})
ok: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
changed: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = neutron'})
changed: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = NEUTRON_PASS'})
ok: [192.168.56.131] => (item={'regex': '^service_metadata_proxy =', 'line': 'service_metadata_proxy = true'})
ok: [192.168.56.131] => (item={'regex': '^metadata_proxy_shared_secret =', 'line': 'metadata_proxy_shared_secret = METADATA_SECRET'})

TASK [Neutron_Controller_Node : Populate the Neutron database] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Restart Compute API service] *****
changed: [192.168.56.131]

TASK [Neutron_Controller_Node : Restart Networking services] *****
changed: [192.168.56.131] => (item=neutron-server)
changed: [192.168.56.131] => (item=neutron-linuxbridge-agent)
changed: [192.168.56.131] => (item=neutron-dhcp-agent)
changed: [192.168.56.131] => (item=neutron-metadata-agent)

PLAY [Neutron_Compute_Node] *****

TASK [Gathering Facts] *****
ok: [192.168.56.131]

TASK [Neutron_Compute_Node : Install components] *****
ok: [192.168.56.131]

TASK [Neutron_Compute_Node : Edit /etc/neutron/neutron.conf file] *****
changed: [192.168.56.131] => (item={'regex': '^connection =', 'line': '# connection ='})
ok: [192.168.56.131] => (item={'regex': '^transport_url =', 'line': 'transport_url = rabbit://openstack:RABBIT_PASS@controller'})
ok: [192.168.56.131] => (item={'regex': '^auth_strategy =', 'line': 'auth_strategy = keystone'})
ok: [192.168.56.131] => (item={'regex': '^www_authenticate_uri =', 'line': 'www_authenticate_uri = http://controller:5000'})
ok: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
ok: [192.168.56.131] => (item={'regex': '^memcached_servers =', 'line': 'memcached_servers = controller:11211'})
ok: [192.168.56.131] => (item={'regex': '^auth_type =', 'line': 'auth_type = password'})
ok: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
changed: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = neutron'})
changed: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = NEUTRON_PASS'})
ok: [192.168.56.131] => (item={'regex': '^lock_path =', 'line': 'lock_path = /var/lib/neutron/tmp'})

TASK [Neutron_Compute_Node : Edit /etc/neutron/plugins/ml2/linuxbridge_agent.ini file] *****
ok: [192.168.56.131] => (item={'regex': '^physical_interface_mappings =', 'line': 'physical_interface_mappings = provider:PROVIDER_INTERFACE_NAME'})
ok: [192.168.56.131] => (item={'regex': '^enable_vxlan =', 'line': 'enable_vxlan = false'})
ok: [192.168.56.131] => (item={'regex': '^enable_security_group =', 'line': 'enable_security_group = true'})
ok: [192.168.56.131] => (item={'regex': '^firewall_driver =', 'line': 'firewall_driver = neutron.agent.linux.iptables_firewall.IptablesFirewallDriver'})
```

```
changed: [192.168.56.131] => (item={'regex': '^net.bridge.bridge-nf-call-iptables =', 'line': 'net.bridge.bridge-nf-call-iptables = 1'})
changed: [192.168.56.131] => (item={'regex': '^net.bridge.bridge-nf-call-ip6tables =', 'line': 'net.bridge.bridge-nf-call-ip6tables = 1'})

TASK [Neutron_Compute_Node : Edit /etc/nova/nova.conf file] *****
ok: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
ok: [192.168.56.131] => (item={'regex': '^auth_type =', 'line': 'auth_type = password'})
ok: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
ok: [192.168.56.131] => (item={'regex': '^region_name =', 'line': 'region_name = RegionOne'})
ok: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
ok: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = neutron'})
ok: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = NEUTRON_PASS'})

TASK [Neutron_Compute_Node : Restart compute service] *****
changed: [192.168.56.131]

TASK [Neutron_Compute_Node : Restart linux bridge agent] *****
changed: [192.168.56.131]
```

```
PLAY [Horizon] *****

TASK [Gathering Facts] *****
ok: [192.168.56.131]

TASK [Horizon : Install component] *****
changed: [192.168.56.131]

TASK [Horizon : Edit /etc/openstack-dashboard/local_settings.py file] *****
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_HOST =', 'line': 'OPENSTACK_HOST = "controller"'})
changed: [192.168.56.131] => (item={'regex': '^ALLOWED_HOSTS =', 'line': 'ALLOWED_HOSTS = ["one.example.com", "two.example.com"]'})
changed: [192.168.56.131] => (item={'regex': '^SESSION_ENGINE =', 'line': 'SESSION_ENGINE = "django.contrib.sessions.backends.cache"'})
changed: [192.168.56.131] => (item={'regex': '^CACHES =', 'line': 'CACHES = {"default": {"BACKEND": "django.core.cache.backends.memcached.MemcachedCache", "LOCATION": "controller:11211"}}'})
ok: [192.168.56.131] => (item={'regex': '^OPENSTACK_KEYSTONE_URL =', 'line': 'OPENSTACK_KEYSTONE_URL = "http://%s/identity/v3" % OPENSTACK_HOST'})
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT =', 'line': 'OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True'})
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_API_VERSIONS =', 'line': 'OPENSTACK_API_VERSIONS = {"identity": 3, "image": 2, "volume": 3}'})
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_KEYSTONE_DEFAULT_DOMAIN =', 'line': 'OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = "Default"'})
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_KEYSTONE_DEFAULT_ROLE =', 'line': 'OPENSTACK_KEYSTONE_DEFAULT_ROLE = "user"'})
changed: [192.168.56.131] => (item={'regex': '^OPENSTACK_NEUTRON_NETWORK =', 'line': '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}'})
ok: [192.168.56.131] => (item={'regex': '^TIME_ZONE =', 'line': 'TIME_ZONE = "UTC"'})

TASK [Horizon : Edit /etc/apache2/conf-available/openstack-dashboard.conf file] *****
ok: [192.168.56.131]

TASK [Horizon : Reload apache2] *****
changed: [192.168.56.131]
```

```
PLAY [Cinder_Controller_Node] *****

TASK [Gathering Facts] *****
ok: [192.168.56.131]

TASK [Cinder_Controller_Node : Create Cinder database] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Grant privileges on Cinder database] *****
changed: [192.168.56.131] => (item=localhost)
changed: [192.168.56.131] => (item=%)

TASK [Cinder_Controller_Node : Source the admin credentials] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Create Cinder user] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Add admin role to cinder user] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Create cinderv3 service entity] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Create Block Storage service API endpoints (public)] *****
changed: [192.168.56.131]
```

```

TASK [Cinder_Controller_Node : Create Block Storage service API endpoints (internal)] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Create Block Storage service API endpoints (admin)] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Install packages] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Edit cinder.conf file] *****
changed: [192.168.56.131] => (item={'regex': '^connection =', 'line': 'connection = mysql+pymysql://cinder:CINDER_DBPASS@controller/cinder'})
changed: [192.168.56.131] => (item={'regex': '^transport_url =', 'line': 'transport_url = rabbit://openstack:RABBIT_PASS@controller'})
ok: [192.168.56.131] => (item={'regex': '^auth_strategy =', 'line': 'auth_strategy = keystone'})
changed: [192.168.56.131] => (item={'regex': '^www_authenticate_url =', 'line': 'www_authenticate_url = http://controller:5000'})
changed: [192.168.56.131] => (item={'regex': '^auth_url =', 'line': 'auth_url = http://controller:5000'})
changed: [192.168.56.131] => (item={'regex': '^memcached_servers =', 'line': 'memcached_servers = controller:11211'})
changed: [192.168.56.131] => (item={'regex': '^project_domain_name =', 'line': 'project_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^user_domain_name =', 'line': 'user_domain_name = default'})
changed: [192.168.56.131] => (item={'regex': '^project_name =', 'line': 'project_name = service'})
changed: [192.168.56.131] => (item={'regex': '^username =', 'line': 'username = cinder'})
changed: [192.168.56.131] => (item={'regex': '^password =', 'line': 'password = CINDER_PASS'})
changed: [192.168.56.131] => (item={'regex': '^my_ip =', 'line': 'my_ip = 10.0.0.11'})
changed: [192.168.56.131] => (item={'regex': '^lock_path =', 'line': 'lock_path = /var/lib/cinder/tmp'})

TASK [Cinder_Controller_Node : Populate the Neutron database] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Edit /etc/nova/nova.conf file] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Restart compute service] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Restart block storage services] *****
changed: [192.168.56.131]

TASK [Cinder_Controller_Node : Restart block storage services] *****
changed: [192.168.56.131]

PLAY RECAP *****
192.168.56.131      : ok=51   changed=28   unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

```

This is the output of my main playbook. As we can see, there is no error encountered. There are 0 unreachable, 0 failed, 0 skipped, and 0 ignored, meaning the executing of my playbook was successful. Below are the proofs that the execution of my playbook really successfully does all the tasks I created for installing different services of OpenStack.

### DATABASES PROOF FOR NEUTRON AND CINDER:

```

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| cinder   |
| glance   |
| information_schema |
| keystone |
| mysql     |
| neutron   |
| nova      |
| nova_api  |
| nova_cell0 |
| performance_schema |
| sys       |
+-----+
11 rows in set (0.010 sec)

```



## NEUTRON\_CONTROLLER\_NODE PROOF:

```
MariaDB [(none)]> SHOW GRANTS FOR 'neutron'@'localhost';
+-----+
| Grants for neutron@localhost |
+-----+
| GRANT USAGE ON *.* TO 'neutron'@'localhost' IDENTIFIED BY PASSWORD '*4A2A29F92D5B7E938289C95542CDA1C5B747AF01' |
| GRANT ALL PRIVILEGES ON 'neutron'.* TO 'neutron'@'localhost' |
+-----+
2 rows in set (0.018 sec)

MariaDB [(none)]> SHOW GRANTS FOR 'neutron'@'%';
+-----+
| Grants for neutron@% |
+-----+
| GRANT USAGE ON *.* TO 'neutron'@'%' IDENTIFIED BY PASSWORD '*4A2A29F92D5B7E938289C95542CDA1C5B747AF01' |
| GRANT ALL PRIVILEGES ON 'neutron'.* TO 'neutron'@'%' |
+-----+
2 rows in set (0.000 sec)
```

```
kazuki@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 Sat 2023-12-16 20:18:51 +08; 4s ago
     Docs: man:neutron-server(1)
    Main PID: 4813 (neutron-server)
      Tasks: 1 (limit: 4594)
     Memory: 101.0M
        CPU: 1.434s
    CGroup: /system.slice/neutron-server.service
            └─4813 /usr/bin/python3 /usr/bin/neutron-server --config-file=/etc/neutron/neutron.conf --config-file=/etc/neutron/plug
lines 1-10/10 (END)
```

```
kazuki@server2:~$ systemctl status neutron-linuxbridge-agent
● neutron-linuxbridge-agent.service - Openstack Neutron Linux Bridge Agent
   Loaded: loaded (/lib/systemd/system/neutron-linuxbridge-agent.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-16 20:13:55 +08; 5min ago
     Process: 2132 ExecStartPre=/bin/mkdir -p /var/lock/neutron /var/log/neutron /var/lib/neutron (code=exited, status=0/SUCCESS)
     Process: 2142 ExecStartPre=/bin/chown neutron:neutron /var/lock/neutron /var/log/neutron /var/lib/neutron (code=exited, status=0/SUCCESS)
     Process: 2164 ExecStartPre=/sbin/modprobe br_netfilter (code=exited, status=0/SUCCESS)
    Main PID: 2172 (neutron-linuxbr)
      Tasks: 6 (limit: 4594)
     Memory: 255.8M
        CPU: 4.898s
    CGroup: /system.slice/neutron-linuxbridge-agent.service
            └─2172 "neutron-linuxbridge-agent (/usr/bin/python3 /usr/bin/neutron-linuxbridge-agent --config-file=/etc/neutron/neutron.conf --config-file=/etc/neutron/plugins/
            └─2252 /usr/bin/python3 /bin/privsep-helper --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/
            └─2281 /usr/bin/python3 /bin/privsep-helper --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/

Warning: some journal files were not opened due to insufficient permissions.
lines 1-16/16 (END)
```

```
kazuki@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 Sat 2023-12-16 20:13:54 +08; 6min ago
     Docs: man:neutron-dhcp-agent(1)
    Main PID: 2131 (neutron-dhcp-ag)
      Tasks: 3 (limit: 4594)
     Memory: 111.1M
        CPU: 2.152s
    CGroup: /system.slice/neutron-dhcp-agent.service
            └─2131 "neutron-dhcp-agent (/usr/bin/python3 /usr/bin/neutron-dhcp-agent --config-file=/etc/neutron/neutron.conf --conf
lines 1-12/12 (END)
```

```
kazuki@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 Sat 2023-12-16 20:13:54 +08; 7min ago
     Docs: man:neutron-metadata-agent(1)
    Main PID: 2133 (neutron-metadat)
      Tasks: 3 (limit: 4594)
     Memory: 120.2M
        CPU: 2.068s
    CGroup: /system.slice/neutron-metadata-agent.service
            └─2133 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --config-file=/etc/neutron/neutron.conf
            └─2227 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --config-file=/etc/neutron/neutron.conf
lines 1-13/13 (END)
```

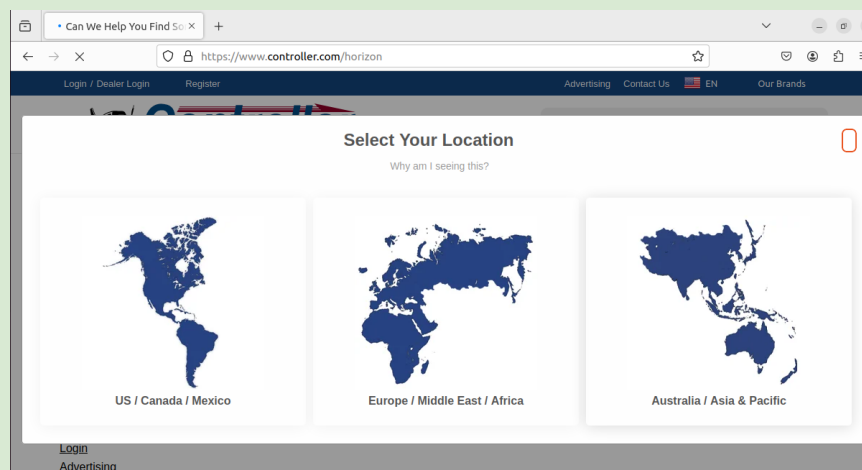
```
kazuki@server2:~$ systemctl status nova-api
● nova-api.service - OpenStack Compute API
   Loaded: loaded (/lib/systemd/system/nova-api.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-16 20:21:19 +08; 356ms ago
     Docs: man:nova-api(1)
    Main PID: 5783 (nova-api)
      Tasks: 1 (limit: 4594)
     Memory: 20.2M
        CPU: 143ms
    CGroup: /system.slice/nova-api.service
            └─5783 /usr/bin/python3 /usr/bin/nova-api --config-file=/etc/nova/nova.conf --log-file=/var/log/nova/nova-api.log
```

### NEUTRON\_COMPUTE\_NODE PROOF:

```
kazuki@server2:~$ systemctl status nova-compute
● nova-compute.service - OpenStack Compute
   Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-16 20:29:10 +08; 2s ago
     Main PID: 8738 (nova-compute)
        Tasks: 1 (limit: 4594)
       Memory: 67.5M
          CPU: 831ms
         CGroup: /system.slice/nova-compute.service
                └─8738 /usr/bin/python3 /usr/bin/nova-compute --config-file=/etc/nova/nova.conf --config-file=/etc/nova/nova-compute.conf
```

```
kazuki@server2:~$ systemctl status neutron-linuxbridge-agent
● neutron-linuxbridge-agent.service - Openstack Neutron Linux Bridge Agent
   Loaded: loaded (/lib/systemd/system/neutron-linuxbridge-agent.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-16 20:13:55 +08; 14min ago
     Process: 2132 ExecStartPre=/bin/mkdir -p /var/lock/neutron /var/log/neutron /var/lib/neutron (code=exited, status=0/SUCCESS)
     Process: 2142 ExecStartPre=/bin/chown neutron:neutron /var/lock/neutron /var/log/neutron /var/lib/neutron (code=exited, status=0/SUCCESS)
     Process: 2164 ExecStartPre=/sbin/modprobe br_netfilter (code=exited, status=0/SUCCESS)
   Main PID: 2172 (neutron-linuxbr)
      Tasks: 6 (limit: 4594)
     Memory: 256.3M
        CPU: 5.133s
   CGroup: /system.slice/neutron-linuxbridge-agent.service
           └─2172 "neutron-linuxbridge-agent (/usr/bin/python3 /usr/bin/neutron-linuxbridge-agent --config-file=/etc/neutron/neutr
           └─2252 /usr/bin/python3 /bin/privsep-helper --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/
           └─2281 /usr/bin/python3 /bin/privsep-helper --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/
```

## HORIZON PROOF:



```
kazuki@server2:~$ systemctl status apache2.service
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-16 20:13:31 +08; 21min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 985 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 1169 (apache2)
    Tasks: 105 (limit: 4594)
   Memory: 111.3M
      CPU: 1.418s
   CGroup: /system.slice/apache2.service
           └─1169 /usr/sbin/apache2 -k start
           └─1203 "wsqgl:cinder-wsqt" -k start
           └─1209 "wsqgl:cinder-wsqt" -k start
           └─1210 "wsqgl:cinder-wsqt" -k start
           └─1211 "wsqgl:cinder-wsqt" -k start
           └─1212 "wsqgl:cinder-wsqt" -k start
           └─1213 "wsqgl:horizon" -k start
           └─1214 "wsqgl:horizon" -k start
           └─1215 "wsqgl:horizon" -k start
           └─1216 "wsqgl:keystone-pu" -k start
           └─1217 "wsqgl:keystone-pu" -k start
           └─1218 "wsqgl:keystone-pu" -k start
           └─1219 "wsqgl:keystone-pu" -k start
           └─1222 "wsqgl:keystone-pu" -k start
           └─1223 "wsqgl:placement-a" -k start
           └─1224 "wsqgl:placement-a" -k start
           └─1225 "wsqgl:placement-a" -k start
           └─1385 "wsqgl:placement-a" -k start
           └─1400 "wsqgl:placement-a" -k start
           └─1415 /usr/sbin/apache2 -k start
           └─1416 /usr/sbin/apache2 -k start
           └─1472 /usr/sbin/apache2 -k start
           └─1473 /usr/sbin/apache2 -k start
           └─1519 /usr/sbin/apache2 -k start
```





```
kazuki@workstation:~/CPE232_HOA15.1$ git add .
kazuki@workstation:~/CPE232_HOA15.1$ git commit -m "HOA15"
[main 9a25852] HOA15
7 files changed, 446 insertions(+)
create mode 100644 ansible.cfg
create mode 100644 inventory
create mode 100644 openstack_install.yml
create mode 100644 roles/Cinder_Controller_Node/tasks/main.yml
create mode 100644 roles/Horizon/tasks/main.yml
create mode 100644 roles/Neutron_Compute_Node/tasks/main.yml
create mode 100644 roles/Neutron_Controller_Node/tasks/main.yml
kazuki@workstation:~/CPE232_HOA15.1$ git push
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 2 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (18/18), 4.30 KiB | 881.00 KiB/s, done.
Total 18 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To github.com:kazzzuki/CPE232_HOA15.1.git
a377515..9a25852 main -> main
```

I used the command “git add .” to add all the created files to my github repository then commit it using the “git commit” command and then lastly push it using the “git push” command.

**GITHUB REPOSITORY LINK:** [https://github.com/kazzzuki/CPE232\\_HOA15.1](https://github.com/kazzzuki/CPE232_HOA15.1)

### Reflections:

Answer the following:

1. Describe Neutron, Horizon and Cinder services

- Based on the provided document of OpenStack services, Neutron is the networking service, Horizon is the dashboard, and Cinder is the Block Storage service. The Horizon and Cinder are components that are advised to be installed along with the minimal deployment of Openstack services. Neutron provided network connectivity between the component services of the OpenStack cloud infrastructure, it is also the one who is incharge of the creation and management of network resources like virtual networks. Horizon is a GUI that allows us or the users to interact and manage different OpenStack services. Lastly, Cinder is the one responsible for the management of block storage for virtual machines in the OpenStack clouds.

### Conclusions:

In this activity, I successfully created a workflow in installing OpenStack using Ansible as IaC. This workflow involves a step by step approach such as the installation and configuration of different services of OpenStack like Neutron, Horizon, and Cinder. I learned about these different cloud services and their advantages and disadvantages. For Neutron, it is flexible especially in configuring and managing networking resources and it is also capable in complex architectures to support scalable cloud environments. But its configurations and terminologies are too difficult for beginners like me. For Horizon, this provides a user-friendly GUI that allows users to interact and manage different resources from OpenStack without using any commands or codes. But it is limited for advanced configurations and it relies on web browsers and there might be some compatibility issues. Lastly, for Cinder, it provides scalability and flexibility for block storage that allows users to attach and detach volumes and it can be integrated with Nova compute service. But its concept might be complex for users, especially the beginners and it only supports limited storage types.