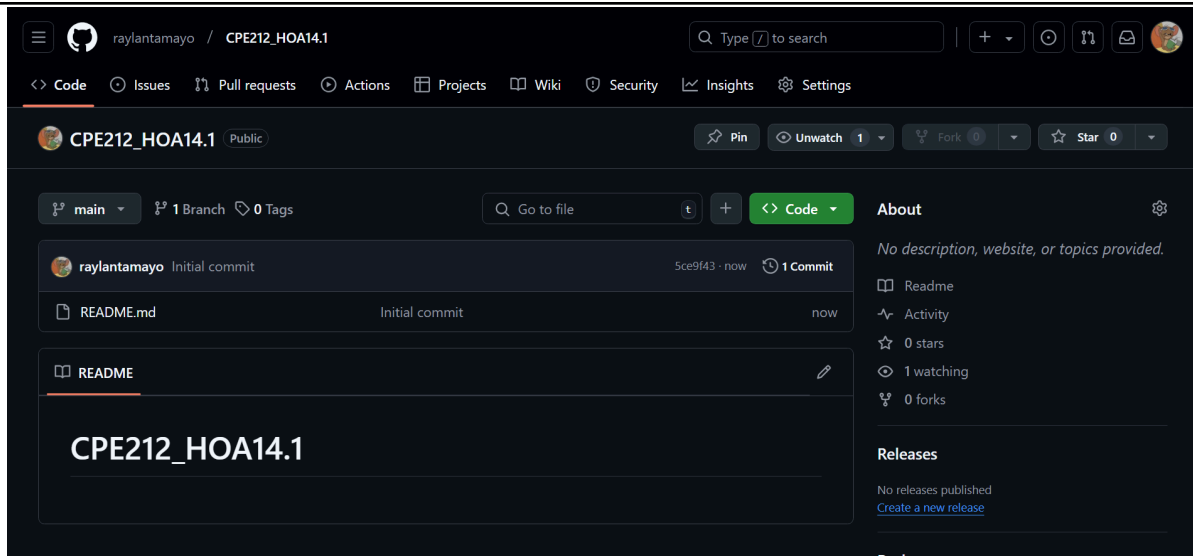


<b>Name: Tamayo, Ray Lan A.</b>	<b>Date Performed: 12/10/2024</b>
<b>Course/Section: CPE31S21</b>	<b>Date Submitted: 12/10/2024</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY: First 2024-2025</b>
<b>Activity 14: OpenStack Installation (Keystone, Glance, Nova)</b>	
<b>1. Objectives</b>	
Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).	
<b>2. Intended Learning Outcomes</b>	
<ol style="list-style-type: none"> <li>1. Analyze the advantages and disadvantages of cloud services</li> <li>2. Evaluate different Cloud deployment and service models</li> <li>3. Create a workflow to install and configure OpenStack base services using Ansible as documentation and execution.</li> </ol>	
<b>3. Resources</b>	
<p>Oracle VirtualBox (Hypervisor)</p> <p>1x Ubuntu VM or Centos VM</p>	
<b>4. Tasks</b>	
<ol style="list-style-type: none"> <li>1. Create a new repository for this activity.</li> <li>2. Create a playbook that converts the steps in the following items in <a href="https://docs.openstack.org/install-guide/">https://docs.openstack.org/install-guide/</a> <ol style="list-style-type: none"> <li>a. Keystone (Identity Service)</li> <li>b. Glance (Imaging Service)</li> <li>c. Nova (Compute Service)</li> <li>d. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file.</li> <li>e. Add, commit and push it to your GitHub repo.</li> </ol> </li> </ol>	
<b>5. Output</b> (screenshots and explanations)	
<p><b>Task 1: Create a File</b></p> <ol style="list-style-type: none"> <li>1. Create a new repository for this Hands-On Activity.</li> </ol>	



```
tamayo@workstation: ~  
tamayo@workstation:~$ git clone git@github.com:raylantanamayo/CPE212_HOA14.1.git  
Cloning into 'CPE212_HOA14.1'...  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
Receiving objects: 100% (3/3), done.  
tamayo@workstation:~$
```

2. Create the ansible.cfg and inventory file (must include one Ubuntu)

```
tamayo@workstation: ~/CPE212_HOA14.1  
GNU nano 7.2 ansible.cfg *  
[defaults]  
  
inventory = inventory  
host_key_checking = False  
  
deprecation_warnings = False  
  
remote_user = tamayo  
private_key_file = ~/.ssh/
```

```
GNU nano 7.2
[Keystone]
192.168.56.102

[Glance]
192.168.56.102

[Nova]
192.168.56.102
```

## Task 2: Create Playbook for Installing OpenStack

1. Create a playbook and name it install\_openstack.yml.

```
tamayo@workstation: ~/CPE212_HOA14.1
GNU nano 7.2      install_openstack.yml
---
- hosts: all
  become: true
  pre_tasks:
    - name: Install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
        changed_when: false

- hosts: Keystone
  become: true
  roles:
    - role: Keystone

- hosts: Glance
  become: true
  roles:
    - role: Glance

- hosts: Nova
  become: true
  roles:
    - role: Nova
```

## CODE EXPLANATION

It instructs Ansible to run tasks on all hosts, become a privileged user, and execute a pre-task. The pre-task installs updates on Ubuntu using the 'apt' module, specifying a distribution upgrade while suppressing changes indication for brevity.

```
---
- hosts: all
  become: true
  pre_tasks:

    - name: Install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
        changed_when: false
```

### Task 3: Create Roles

1. Create a new directory and name it “roles”. Enter the roles directory and create new directories: Keystone, Nova, and Glance. For each directory, create a directory and name it tasks.

#### For Keystone

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ mkdir Keystone
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Keystone
tamayo@workstation:~/CPE212_H0A14.1/roles/Keystone$ mkdir tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Keystone$ cd tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Keystone/tasks$
```

#### For Nova

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ mkdir Nova
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Nova
tamayo@workstation:~/CPE212_H0A14.1/roles/Nova$ mkdir tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Nova$ cd tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Nova/tasks$
```

### For Glance

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ mkdir Glance
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Glance
tamayo@workstation:~/CPE212_H0A14.1/roles/Glance$ mkdir tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Glance$ cd tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Glance/tasks$
```

### Tree for roles

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ tree
.
├── Glance
│   └── tasks
├── Keystone
│   └── tasks
└── Nova
    └── tasks

7 directories, 0 files
```

2. In each of the tasks for the three directory (Keystone, Nova, Glance), create another file and name it main.yml

### For Keystone

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Keystone/tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Keystone/tasks$ sudo nano main.yml
tamayo@workstation:~/CPE212_H0A14.1/roles/Keystone/tasks$
```

### For Nova

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Nova/tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Nova/tasks$ sudo nano main.yml
```

### For Glance

```
tamayo@workstation:~/CPE212_H0A14.1/roles$ cd Glance/tasks
tamayo@workstation:~/CPE212_H0A14.1/roles/Glance/tasks$ sudo nano main.yml
```

## Tree for roles

```
tamayo@workstation:~/CPE212_HOA14.1/roles$ tree
.
├── Glance
│   └── tasks
│       └── main.yml
├── Keystone
│   └── tasks
│       └── main.yml
└── Nova
    └── tasks
        └── main.yml

7 directories, 3 files
```

3. Copy the code to the main.yml of the each subdirectory.

## For Keystone

```
tamayo@workstation: ~/CPE212_HOA14.1/roles/Keysto
GNU nano 7.2 main.yml *
##Installing the mod_wsgi
- name: Install the prerequisites for mod_wsgi
  apt:
    name:
      - apache2
      - apache2-utils
      - libexpat1
      - ssl-cert
      - python3
      - libapache2-mod-wsgi
    when: ansible_distribution == "Ubuntu"

- name:
  copy:
    dest: /var/www/html/test_script.py
    content: |
      def application(environ,start_response):
        status = '200 OK'
        html = '\n' \
              '\n' \
              ' mod_wsgi is working \n' \
              '\n' \
              '\n'
        response_header = [('Content-type','text/html')]
        start_response(status,response_header)
        return [html]

- name: Create a seperate apache config to serve our python script over HTTP
  copy:
    dest: /etc/apache2/conf-available/wsgi.conf
    content: |
```

```
WSGIScriptAlias /test_wsgi /var/www/html/test_script.py
#- name: restart apache server
# shell: sudo systemctl restart apache2

- name: install the keystone package
  apt:
    name: keystone
    when: ansible_distribution == "Ubuntu"
```

```
- name: initializing the fernet repositories (1)
  shell: keystone-manage fernet_setup --keystone-user keystone --keystone-group keystone
  when: ansible_distribution == "Ubuntu"

- name: initializing the fernet repositories (2)
  shell: keystone-manage credential_setup --keystone-user keystone --keystone-group keystone

#- name: bootstrap the identity service (1)
# shell: keystone-manage bootstrap --bootstrap-password 1234 --bootstrap-admin-url http://co

- name: configure apache http server
  copy:
    dest: /etc/apache2/apache2.conf
    content: |
      ServerName controller
  when: ansible_distribution == "Ubuntu"

- name: configuring administrative account by setting the proper environmental variables (1)
  shell: export OS_USERNAME=admin

- name: configuring administrative account by setting the proper environmental variables (2)
  shell: export OS_PASSWORD=1234

- name: configuring administrative account by setting the proper environmental variables (3)
  shell: export OS_PROJECT_NAME=admin

- name: configuring administrative account by setting the proper environmental variables(4)
  shell: OS_USER_DOMAIN_NAME=Default

- name: configuring administrative account by setting the proper environmental variables(5)
```

```
  shell: OS_PROJECT_DOMAIN_NAME=Default

- name: configuring administrative account by setting the proper environmental variables(6)
  shell: OS_AUTH_URL=http://controller:5000/v3

- name: configuring administrative account by setting the proper environmental variables (7)
  shell: OS_IDENTITY_API_VERSION=3

- block:
  - name: Verifying if already active and running the keystone.
    shell: keystone-manage --help
    register: keystone_service

  - debug:
    msg="{{ keystone_service }}"
```

## For Nova

```
tamayo@workstation: ~/CPE212_HOA14.1/roles/Nova/tas
GNU nano 7.2 main.yml *
#Nova

- name: install the packages
  apt:
    name: nova-compute
    when: ansible_distribution == "Ubuntu"

- name: configuring RabbitMQ message queue access
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [DEFAULT]
      transport_url = rabbit://openstack:1234@controller
      my_ip = 192.168.56.103

- name: configuring identity service access (1)
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [api]
      auth_strategy = keystone

- name: configuring identity service access (2)
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [keystone_authtoken]
      www_authenticate_uri = http://controller:5000/
      auth_url = http://controller:5000/
      memcached_servers = controller:11211
```



```

    auth_type = password
    project_domain_name: Default
    user_domain_name = Default
    project_name = service
    username = nova
    password = 1234

- name: enable and configure remote console access
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [vnc]
      enabled = true
      server_listen = 0.0.0.0
      server_proxyclient_address = $my_ip
      novncproxy_base_url = http://controller:6080/vnc_auto.html

- name: configure the location of the image service API
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [glance]
      api_servers = http://controller:9292

- name: configure the lock path
  copy:
    dest: /etc/nova/nova.conf
    content: |

```

```

      [solo_currency]
      lock_path = /var/lib/nova/tmp

- name: configure the placement API
  copy:
    dest: /etc/nova/nova.conf
    content: |
      [placement]
      region_name = RegionOne
      project_domain_name = Default
      project_name = service
      auth_type = password
      user_domain_name = Default
      auth_url = http://controller:5000/v3
      username = placement
      password = 1234

- name: configuring to make the computer node to support hardware acceleration
  copy:
    dest: /etc/nova/nova-compute.conf
    content: |
      [libvirt]
      virt_type = qemu

- name: restarting the computer service
  shell: service nova-compute restart

- block:

```

```
- name: Verifying if already running and active the nova-compute.  
  shell: systemctl status nova-compute  
  register: novacompute_service  
  
- debug:  
  msg="{[ novacompute_service ]}"
```

## For Glance

```
tamayo@workstation: ~/CPE212_HOA14.1/roles/Glance/tasks  
GNU nano 7.2 main.yml *  
##install and config  
  
- name: install and configure components of glance  
  apt:  
    name: glance  
    when: ansible_distribution == "Ubuntu"  
  
##editing the [database] /etc/glance/glance-api.conf  
  
- name: configuring database access  
  copy:  
    dest: /etc/glance/glance-api.conf  
    content: |  
      [database]  
      connection = mysql+pymysql://glance:1234@controller/glance  
      [keystone_authtoken]  
      www_authenticate_uri = http://controller:5000  
      auth_url = http://controller:5000  
      memcached_servers = controller:11211  
      auth_type = password  
      project_domain_name = Default  
      user_domain_name = Default  
      project_name = service  
      username = glance  
      password = 1234  
      [paste_deploy]  
      flavor = keystone
```

```

- name: configuring the local file system store and location of image files
  copy:
    dest: /etc/glance/glance-api.conf
    content: |
      [glance_store]
      stores = file, http
      default_store = file
      filesystem_store_datadir = /var/lib/glance/images/

- name: configuring the access to keystone
  copy:
    dest: /etc/glance/glance-api.conf
    content: |
      [oslo_limit]
      auth_url = http://controller:5000
      auth_type = password
      user_domain_id = default
      username = MY_SERVICE
      system_scope = all
      password = 1234
      endpoint_id = ENDPOINT_ID
      region_name = RegionOne

- name: enable per-tenant quotas
  copy:
    dest: /etc/glance/glance-api.conf
    content: |
      [DEFAULT]
      use_keystone_quotas = True

- name: restart the image services
  shell: service glance-api restart
  when: ansible_distribution == "Ubuntu"

- block:
  - name: Verifying if already installed Glance.
    shell: glance --version
    register: glance_version

    - debug:
      msg="{{ glance_version }}"

  - block:
    - name: Verifying if already active and running the Glance.
      shell: systemctl status glance-api
      register: glance_service

    - debug:
      msg="{{ glance_service }}"

```

#### Task 4: Run and Verify

1. Run the command `ansible-playbook --ask-become-pass install_openstack.yml` to completely install the OpenStack.

```

PLAY [all] *****

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

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

PLAY [Keystone] *****

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

TASK [Keystone : Install the prerequisites for mod_wsgi] *****
changed: [192.168.56.102]

TASK [Keystone : copy] *****
ok: [192.168.56.102]

TASK [Keystone : Create a seperate apache config to serve our python script over HTTP] ***
ok: [192.168.56.102]

TASK [Keystone : install the keystone package] *****
changed: [192.168.56.102]

TASK [Keystone : initializing the fernet repositories (1)] *****
changed: [192.168.56.102]

TASK [Keystone : initializing the fernet repositories (2)] *****
changed: [192.168.56.102]

```

```

changed: [192.168.56.102]

TASK [Keystone : configure apache http server] *****
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables (1)
] ***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables (2)
] ***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables (3)
] ***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables(4)]
***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables(5)]
***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables(6)]
***
changed: [192.168.56.102]

TASK [Keystone : configuring administrative account by setting the proper environmental variables (7)
] ***
changed: [192.168.56.102]

TASK [Keystone : Verifying if already active and running the keystone.] *****
changed: [192.168.56.102]

```

```

TASK [Keystone : debug] *****
ok: [192.168.56.102] => {
  "msg": {
    "changed": true,
    "cmd": "keystone-manage --help",
    "delta": "0:00:00.486644",
    "end": "2023-11-30 18:12:13.120187",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-30 18:12:12.633543",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "usage: keystone-manage [bootstrap|credential_migrate|credential_rotate|credential_
setup|db_sync|db_version|doctor|domain_config_upload|fernet_rotate|fernet_setup|mapping_populate|mapp
ing_purge|mapping_engine|saml_idp_metadata|token_flush]\n\noptional arguments:\n  -h, --help
show this help message and exit\n  --config-dir DIR      Path to a config directory to pull '*.con
f' files\n                        from. This file set is sorted, so as to provide a\n                        predictable parse order if individual options are\n                        over-ridden. The s
et is parsed after the file(s)\n                        specified via previous --config-file, argumen
ts hence\n                        over-ridden options in the directory take precedence.\n  --config-f
ile PATH      Path to a config file to use. Multiple config files\n                        can be speci
fied, with values in later files taking\n                        precedence. Defaults to None.\n  --d
ebug, -d      If set to true, the logging level will be set to DEBUG\n                        in
stead of the default INFO level.\n  --log-config-append PATH, --log-config PATH, --log-config PATH\n                        The name of a logging configuration file. This file is\n
appended to any existing logging configuration files.\n                        For details about lo
gging configuration files, see the\n                        Python logging module documentation. Note
that when\n                        logging configuration files are used then all logging\n
configuration is set in the configuration file and\n                        other loggi
ng configuration options are ignored (for\n                        example, logging_context_format_st
ring).\n  --log-date-format DATE_FORMAT\n                        Defines the format string for %(asct
ime)s in log\n                        records. Default: None . This option is ignored if\n
log_config_append is set.\n  --log-dir LOG_DIR, --logdir LOG_DIR\n
(Optional) The base directory used for relative\n

```

```

    "domain_config_upload",
    "        Upload the domain specific configuration files to the",
    "        database.",
    "    fernet_rotate
    "        Rotate Fernet encryption keys. This assumes you have",
    "        already run keystone-manage fernet_setup. A new",
    "        primary key is placed into rotation, which is used for",
    "        new tokens. The old primary key is demoted to",
    "        secondary, which can then still be used for validating",
    "        tokens. Excess secondary keys (beyond [fernet_tokens]",
    "        max_active_keys) are revoked. Revoked keys are",
    "        permanently deleted. A new staged key will be created",
    "        and used to validate tokens. The next time key",
    "        rotation takes place, the staged key will be put into",
    "        rotation as the primary key. Rotating keys too",
    "        frequently, or with [fernet_tokens] max_active_keys",
    "        set too low, will cause tokens to become invalid prior",
    "        to their expiration.",
    "    fernet_setup
    "        Setup a key repository for Fernet tokens. This also",
    "        creates a primary key used for both creating and",
    "        validating Fernet tokens. To improve security, you",
    "        should rotate your keys (using keystone-manage",
    "        fernet_rotate, for example).",
    "    mapping_populate
    "        Pre-populate entries from domain-specific backends.",
    "        Running this command is not required. It should only",
    "        be run right after the LDAP was configured, when many",
    "        new users were added, or when \"mapping_purge\" is run.",
    "        This command will take a while to run. It is perfectly",
    "        fine for it to run more than several minutes.",
    "    mapping_purge
    "        Purge the mapping table.",
    "    mapping_engine
    "        Execute mapping engine locally.",
    "    saml_idp_metadata
    "        Generate Identity Provider metadata.",
    "    token_flush
    "        Flush expired tokens from the backend."
  ]
}

```

Ubuntu Software

```
PLAY [Glance] *****
TASK [Gathering Facts] *****
ok: [192.168.56.102]

TASK [Glance : install and configure components of glance] *****
changed: [192.168.56.102]

TASK [Glance : configuring database access] *****
changed: [192.168.56.102]

TASK [Glance : configuring the local file system store and location of image files] ***
changed: [192.168.56.102]

TASK [Glance : configuring the access to keystone] *****
changed: [192.168.56.102]

TASK [Glance : enable per-tenant quotas] *****
changed: [192.168.56.102]

TASK [Glance : restart the image services] *****
changed: [192.168.56.102]

TASK [Glance : Verifying if already installed Glance.] *****
changed: [192.168.56.102]

TASK [Glance : debug] *****
ok: [192.168.56.102] => {
  "msg": {
    "changed": true,
    "cmd": "glance --version",
    "delta": "0:00:00.256231",
    "failed": false,
    "rc": 0,
    "start": "2013-01-30 18:12:40.131418",
    "stderr_lines": 0,
    "stdout_lines": 1
  }
}
```

```

ok: [192.168.56.102] => {
  "msg": {
    "changed": true,
    "cmd": "glance --version",
    "delta": "0:00:00.256231",
    "end": "2023-11-30 18:12:40.131418",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-30 18:12:39.875187",
    "stderr": "2.9.1",
    "stderr_lines": [
      "2.9.1"
    ],
    "stdout": "",
    "stdout_lines": []
  }
}

```

TASK [Glance : Verifying if already active and running the Glance.] \*\*\*\*\*  
 changed: [192.168.56.102]

TASK [Glance : debug] \*\*\*\*\*

```

ok: [192.168.56.102] => {
  "msg": {
    "changed": true,
    "cmd": "systemctl status glance-api",
    "delta": "0:00:00.054612",
    "end": "2023-11-30 18:12:40.414857",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-30 18:12:40.360245",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "● glance-api.service - OpenStack Image Service API\n Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor preset: enabled)\n Active: active (running) since Thu 2023-11-30 18:12:39 PST; 796ms ago\n Main PID: 14471 (glance-api)\n Tasks: 3 (limit: 4656)\n CGroup: /system.slice/glance-api.service\n          └─14471 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n          └─14516 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n          └─14517 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/util.py:55: DeprecationWarning: Using function/method 'Healthcheck.Factory()' is deprecated: The healthcheck middleware must now be configured as an application, not as a filter\n Nov 30 18:12:40 server1 glance-api[14471]: val = callable(*args, **kw)",
    "stdout_lines": [
      "● glance-api.service - OpenStack Image Service API",
      "Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor preset: enabled)",
      "Active: active (running) since Thu 2023-11-30 18:12:39 PST; 796ms ago",
      "Main PID: 14471 (glance-api)",
      "Tasks: 3 (limit: 4656)",
      "CGroup: /system.slice/glance-api.service",

```

```

    "start": "2023-11-30 18:12:40.360245",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "● glance-api.service - OpenStack Image Service API\n Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor preset: enabled)\n Active: active (running) since Thu 2023-11-30 18:12:39 PST; 796ms ago\n Main PID: 14471 (glance-api)\n Tasks: 3 (limit: 4656)\n CGroup: /system.slice/glance-api.service\n          └─14471 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n          └─14516 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n          └─14517 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance-api.conf --log-file=/var/log/glance/glance-api.log\n Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require separately.\n Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.parse('x=' + s).load(False)\n Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy/util.py:55: DeprecationWarning: Using function/method 'Healthcheck.Factory()' is deprecated: The healthcheck middleware must now be configured as an application, not as a filter\n Nov 30 18:12:40 server1 glance-api[14471]: val = callable(*args, **kw)",
    "stdout_lines": [
      "● glance-api.service - OpenStack Image Service API",
      "Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor preset: enabled)",
      "Active: active (running) since Thu 2023-11-30 18:12:39 PST; 796ms ago",
      "Main PID: 14471 (glance-api)",
      "Tasks: 3 (limit: 4656)",
      "CGroup: /system.slice/glance-api.service",

```



```

    " Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor preset: enable)",
    "   Active: active (running) since Thu 2023-11-30 18:12:39 PST; 796ms ago",
    " Main PID: 14471 (glance-api)",
    "   Tasks: 3 (limit: 4656)",
    "    CGroup: /system.slice/glance-api.service,
              └─14471 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance
-api.conf --log-file=/var/log/glance/glance-api.log",
              └─14516 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance
-api.conf --log-file=/var/log/glance/glance-api.log",
              └─14517 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glance/glance
-api.conf --log-file=/var/log/glance/glance-api.log",
    " ",
    "Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy
/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require s
eparately.",
    " + s).load(False)",
    "Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy
/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require s
eparately.",
    " + s).load(False)",
    "Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy
/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require s
eparately.",
    " + s).load(False)",
    "Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy
/loadwsgi.py:22: DeprecationWarning: Parameters to load are deprecated. Call .resolve and .require s
eparately.",
    " + s).load(False)",
    "Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/paste/deploy
/utl.py:55: DeprecationWarning: Using function/method 'Healthcheck_factory()' is deprecated: The hea

```

```

PLAY [Nova] *****

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

TASK [Nova : install the packages] *****
changed: [192.168.56.102]

TASK [Nova : configuring RabbitMQ message queue access] *****
changed: [192.168.56.102]

TASK [Nova : configuring identity service access (1)] *****
changed: [192.168.56.102]

TASK [Nova : configuring identity service access (2)] *****
changed: [192.168.56.102]

TASK [Nova : enable and configure remote console access] *****
changed: [192.168.56.102]

TASK [Nova : configure the location of the image service API] *****
changed: [192.168.56.102]

TASK [Nova : configure the lock path] *****
changed: [192.168.56.102]

TASK [Nova : configure the placement API] *****
changed: [192.168.56.102]

Terminal
TASK [Nova : configuring to make the computer node to support hardware acceleration] ***
changed: [192.168.56.102]

TASK [Nova : restarting the computer service] *****
changed: [192.168.56.102]

```



```
TASK [Nova : Verifying if already running and active the nova-compute.] *****
changed: [192.168.56.102]

TASK [Nova : debug] *****
ok: [192.168.56.102] => {
  "msg": {
    "changed": true,
    "cmd": "systemctl status nova-compute",
    "delta": "0:00:00.054407",
    "end": "2023-11-30 18:13:45.985907",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-30 18:13:45.931500",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "● nova-compute.service - OpenStack Compute\n Loaded: loaded (/lib/systemd/system\n/nova-compute.service; enabled; vendor preset: enabled)\n Active: active (running) since Thu 2023-11-30 18:13:45 PST; 251ms ago\n Main PID: 24896 (nova-compute)\n Tasks: 1 (limit: 4656)\n CGroup: /system.slice/nova-compute.service\n      └─24896 /usr/bin/python2 /usr/bin/nova-compute --conf\nig-file=/etc/nova/nova.conf --config-file=/etc/nova/nova-compute.conf --log-file=/var/log/nova/nova-c\nompute.log\nNov 30 18:13:45 server1 systemd[1]: Started OpenStack Compute.",
    "stdout_lines": [
      "● nova-compute.service - OpenStack Compute",
      "Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor preset: ena\nbled)",
      "Active: active (running) since Thu 2023-11-30 18:13:45 PST; 251ms ago",
      "Main PID: 24896 (nova-compute)",
      "Tasks: 1 (limit: 4656)",
      "CGroup: /system.slice/nova-compute.service",
      "      └─24896 /usr/bin/python2 /usr/bin/nova-compute --config-file=/etc/nova/nova.c\nonf --config-file=/etc/nova/nova-compute.conf --log-file=/var/log/nova/nova-compute.log",
      "",
      "Nov 30 18:13:45 server1 systemd[1]: Started OpenStack Compute."
    ]
  }
}

PLAY RECAP *****
192.168.56.102 : ok=43 changed=32 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

2. Show the screenshot of the Keystone, Nova, and Glance that are working.

## Keystone

```
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

keystone/bionic-updates,bionic-updates,bionic-security,bionic-security,now 2:13.0.4-0ubuntu1 all [installed]
python-keystone/bionic-updates,bionic-updates,bionic-security,bionic-security,now 2:13.0.4-0ubuntu1 all [installed,automatic]
python-keystoneauth1/bionic,bionic,now 3.4.0-0ubuntu1 all [installed,automatic]
python-keystoneclient/bionic,bionic,now 1:3.15.0-0ubuntu1 all [installed,automatic]
python-keystonemiddleware/bionic,bionic,now 4.21.0-0ubuntu1 all [installed,automatic]
```

## Nova

```
● nova-compute.service - OpenStack Compute
   Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor pre
   Active: active (running) since Thu 2023-11-30 18:13:45 PST; 10min ago
   Main PID: 24896 (nova-compute)
     Tasks: 1 (limit: 4656)
    CGroup: /system.slice/nova-compute.service
            └─24896 /usr/bin/python2 /usr/bin/nova-compute --config-file=/etc/no

Nov 30 18:13:45 server1 systemd[1]: Started OpenStack Compute.
lines 1-9/9 (END)
```

## Glance

```
● glance-api.service - OpenStack Image Service API
   Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor pres
   Active: active (running) since Thu 2023-11-30 18:12:39 PST; 7min ago
   Main PID: 14471 (glance-api)
     Tasks: 3 (limit: 4656)
    CGroup: /system.slice/glance-api.service
            └─14471 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glan
              └─14516 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glan
                └─14517 /usr/bin/python2 /usr/bin/glance-api --config-file=/etc/glan

Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/pas
Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.pa
Nov 30 18:12:39 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/pas
Nov 30 18:12:39 server1 glance-api[14471]: return pkg_resources.EntryPoint.pa
Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/pas
Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.pa
Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/pas
Nov 30 18:12:40 server1 glance-api[14471]: return pkg_resources.EntryPoint.pa
Nov 30 18:12:40 server1 glance-api[14471]: /usr/lib/python2.7/dist-packages/pas
Nov 30 18:12:40 server1 glance-api[14471]: val = callable(*args, **kw)
lines 1-20/20 (END)
```

### 3. Upload it in the github

```
tamayo@workstation:~/CPE212_HOA14.1$ git add *
tamayo@workstation:~/CPE212_HOA14.1$ git commit -m "OpenStack Installation"
[main 52f6a17] OpenStack Installation
 6 files changed, 344 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 install_openstack.yml
 create mode 100644 inventory
 create mode 100644 roles/Glance/tasks/main.yml
 create mode 100644 roles/Keystone/tasks/main.yml
 create mode 100644 roles/Nova/tasks/main.yml
tamayo@workstation:~/CPE212_HOA14.1$ git push origin
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 2 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (15/15), 3.91 KiB | 3.91 MiB/s, done.
Total 15 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:raylantamayo/CPE212_HOA14.1.git
   5ce9f43..52f6a17  main -> main
tamayo@workstation:~/CPE212_HOA14.1$
```

GitHub Link: [https://github.com/raylantamayo/CPE212\\_HOA14.1.git](https://github.com/raylantamayo/CPE212_HOA14.1.git)

**Reflections:**

Answer the following:

1. Describe Keystone, Glance and Nova services

In OpenStack, Keystone is the identity service, managing user authentication. Glance is the image service, handling virtual machine images. Nova is the compute service, orchestrating the creation and management of virtual machines. Together, they form the backbone of OpenStack, ensuring secure access, efficient image storage, and seamless virtual machine operation for cloud computing.

**Conclusions:**

In this activity, I was able to encounter the Keystone, Nova, and Glance. Compared to the last activity, which also focuses on OpenStack, it is more complicated to understand the entire concept of the three. In the journey of exploring cloud services, we've unveiled a realm of possibilities and challenges. Embracing the advantages of flexibility and scalability, we also faced the intricacies of security concerns and potential downtime. By delving into diverse cloud deployment and service models, we gained a nuanced understanding of tailoring solutions to specific needs. Notably, orchestrating an OpenStack installation through Ansible illuminated the power of automation and meticulous documentation. The lesson echoes: the cloud's potential is boundless, but success lies in navigating its terrain wisely. It's not just about technology; it's about a mindful fusion of innovation, adaptability, and strategic decision-making.