

Name: Buenvenida, Ken Benedict D.	Date Performed: December 01, 2023
Course/Section: CpE232 - CpE31S4	Date Submitted: December 03, 2023
Instructor: Engr. Jonathan Taylar	Semester and SY: 1st Semester 2023-2024
Activity 14: OpenStack Installation (Keystone, Glance, Nova)	
1. Objectives	
Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).	
2. Intended Learning Outcomes	
<ol style="list-style-type: none"> 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	
<p>Oracle VirtualBox (Hypervisor)</p> <p>1x Ubuntu VM or Centos VM</p>	
4. Tasks	
<ol style="list-style-type: none"> 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/ <ol style="list-style-type: none"> a. Keystone (Identity Service) b. Glance (Imaging Service) c. Nova (Compute Service) d. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file. e. Add, commit and push it to your GitHub repo. 	
5. Output (screenshots and explanations)	
<ol style="list-style-type: none"> 1. Create a new repository for this activity. 	


Input

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*


Owner *

 KBDBuenvenida

 /


Repository name *

HOA14


 HOA14 is available.

Great repository names are short and memorable. Need inspiration? How about [silver-fishstick](#) ?

Description (optional)

☒  Public

Anyone on the internet can see this repository. You choose who can commit.

☐  Private

You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore


.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)


 You are creating a public repository in your personal account.


Create repository


Output


HOA14


Public

 Pin

 Unwatch 1


 Fork

 Star 0

 Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.


Get started with GitHub Copilot

 Add collaborators to this repository

Search for people using their GitHub username or email address.

Invite collaborators

Quick setup — if you've done this kind of thing before

 Set up in Desktop

 or

HTTPS

SSH

git@github.com:KBDBuenvenida/HOA14.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# HOA14" >> README.ad
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:KBDBuenvenida/HOA14.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:KBDBuenvenida/HOA14.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

git clone repository

```
ken@controlNode:~$ git clone git@github.com:KBDBuenvenida/H0A14.git
Cloning into 'H0A14'...
warning: You appear to have cloned an empty repository.
ken@controlNode:~$
```

2. Create a playbook that converts the steps into the following items.

Input

```
ken@controlNode:~/H0A13$ cp ansible.cfg inventory ~/H0A14
ken@controlNode:~/H0A13$
```

Output

```
ken@controlNode:~/H0A14$ ls
ansible.cfg  inventory
```

create a playbook named 'playbook.yml'

Input

```
ken@controlNode:~/H0A14$ sudo nano playbook.yml
[sudo] password for ken:
```

Output

```
GNU nano 6.2
---
- hosts: Ubuntu
  become: true
  roles:
    - keystone
    - glance
    - nova

- hosts: CentOS
  become: true
  roles:
    - keystone
    - glance
    - nova
```

Create a directory named 'roles'

Input	<pre>ken@controlNode:~/HOA14\$ mkdir roles</pre>
Output	<pre>ken@controlNode:~/HOA14\$ ls ansible.cfg inventory playbook.yml roles</pre>

create a directory inside roles named nova keystone glance

Input & Output	<pre>ken@controlNode:~/HOA14\$ cd roles ken@controlNode:~/HOA14/roles\$ mkdir nova keystone glance ken@controlNode:~/HOA14/roles\$ ls glance keystone nova</pre>
----------------	--

create a directory inside glance named files and tasks

Input & Output	<pre>ken@controlNode:~/HOA14/roles/glance\$ mkdir files tasks ken@controlNode:~/HOA14/roles/glance\$ ls files tasks</pre>
----------------	--

create a directory inside nova named files and tasks

Input & Output	<pre>ken@controlNode:~/HOA14/roles/nova\$ mkdir files tasks ken@controlNode:~/HOA14/roles/nova\$ ls files tasks</pre>
----------------	--

create a directory inside keystone named files handlers and tasks

Input & Output

```
ken@controlNode:~/HOA14/roles/keystone$ mkdir files handlers tasks
ken@controlNode:~/HOA14/roles/keystone$ ls
files handlers tasks
```

create a .conf file for glance-api

Input

```
ken@controlNode:~/HOA14/roles/glance/files$ sudo nano glance-api.conf
```

Output

```
GNU nano 6.2 glance-api.conf
[DEFAULT]

#
# From glance.api
#

# DEPRECATED:
# Set the image owner to tenant or the authenticated user.
#
# Assign a boolean value to determine the owner of an image. When set to
# True, the owner of the image is the tenant. When set to False, the
# owner of the image will be the authenticated user issuing the request.
# Setting it to False makes the image private to the associated user and
# sharing with other users within the same tenant (or "project")
# requires explicit image sharing via image membership.
#
# Possible values:
#   * True
#   * False
#
# Related options:
#   * None
#
# (boolean value)
# This option is deprecated for removal since Rocky.
# Its value may be silently ignored in the future.
# Reason:
# The non-default setting for this option misaligns Glance with other
# OpenStack services with respect to resource ownership. Further, surveys
# indicate that this option is not used by operators. The option will be
# removed early in the 'S' development cycle following the standard OpenStack
# deprecation policy. As the option is not in wide use, no migration path is
# proposed.
owner_is_tenant = true

#
[ Read 11769 lines ]
```

create a configure.yml for glance

Input

```
ken@controlNode:~/HOA14/roles/glance/tasks$ sudo nano configure.yml
```

Output	<pre> GNU nano 6.2 configure.yml - name: Copying the config file copy: src: glance-api.conf dest: /etc/glance/glance-api.conf owner: root group: glance mode: 0640 - name: Populating the database command: su -s /bin/sh -c "glance-manage db_sync" glance failed_when: false no_log: true - name: Restarting glance-api service: name: openstack-glance-api.service state: started enabled: true </pre>
--------	---

create main.yml for glance

Input	<pre> ken@controlNode:~/H0A14/roles/glance/tasks\$ sudo nano main.yml </pre>
Output	<pre> GNU nano 6.2 - name: Installation Glance apt: name: - glance state: latest update_cache: yes when: ansible_distribution == "Ubuntu" - name: Start Glance service: name: glance-api state: restarted enabled: yes </pre>

create a main.yml inside handlers directory under keystone

Input	<pre> ken@controlNode:~/H0A14/roles/keystone/handlers\$ sudo nano main.yml </pre>
-------	---

Output	<pre> GNU nano 6.2 main.yml - name: Creating link command: ln -s /usr/share/keystone/wsgi-keystone.conf /etc/httpd/conf.d/ failed_when: false no_log: true - name: Reloading firewall command: firewall-cmd --reload </pre>
--------	---

create main.yml inside tasks directory under keystone

Input	<pre> ken@controlNode:~/HOA14/roles/keystone/tasks\$ sudo nano main.yml </pre>
Output	<pre> GNU nano 6.2 - name: Install Keystone apt: name: - keystone - apache2 - php - libapache2-mod-php state: latest update_cache: yes when: ansible_distribution == "Ubuntu" </pre>

create a nova.conf inside files directory under nova

Input	<pre> ken@controlNode:~/HOA14/roles/nova/files\$ sudo nano nova.conf </pre>
-------	---

Output	<pre> GNU nano 6.2 nova.conf [DEFAULT] # # From nova.conf # # # Availability zone for internal services. For more information, refer to the # documentation. (string value) #internal_service_availability_zone=internal # # Default availability zone for compute services. For more information, refer to # the documentation. (string value) #default_availability_zone=nova # # Default availability zone for instances. For more information, refer to the # documentation. (string value) #default_schedule_zone=<None> # # Length of generated instance admin passwords (integer value) # Minimum value: 0 #password_length=12 # # Time period to generate instance usages for. It is possible to define optional # offset to given period by appending @ character followed by a number defining # offset. For more information, refer to the documentation. (string value) #instance_usage_audit_period=month # # Start and use a daemon that can run the commands that need to be run with # root privileges. This option is usually enabled on nodes that run nova compute # processes. # (boolean value) #use_rootwrap_daemon=false </pre>
--------	---

create a main.yml inside tasks directory under nova

Input	<pre> ken@controlNode:~/H0A14/roles/nova/tasks\$ sudo nano main.yml </pre>
Output	<pre> GNU nano 6.2 - name: Installation Nova apt: name: - nova-compute - python3-openstackclient state: latest update_cache: yes when: ansible_distribution == "Ubuntu" - name: Start Nova service: name: nova-compute state: restarted enabled: yes </pre>

run the playbook.yml

Input & Output

```
ken@controlNode: /MDA1$ ansible-playbook --ask-become-pass playbook.yml
BECOME password:

PLAY [Ubuntu] *****

TASK [Gathering Facts] *****
ok: [192.168.56.103]

TASK [keystone : Install Keystone] *****
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/core.py) as it seems to be invalid: cannot import
name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/mathstuff.py) as it seems to be invalid: cannot
import name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
changed: [192.168.56.103]

TASK [glance : Installation Glance] *****
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/core.py) as it seems to be invalid: cannot import
name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/mathstuff.py) as it seems to be invalid: cannot
import name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
ok: [192.168.56.103]

TASK [glance : Start Glance] *****
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/core.py) as it seems to be invalid: cannot import
name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/mathstuff.py) as it seems to be invalid: cannot
import name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
changed: [192.168.56.103]

TASK [nova : Installation Nova] *****
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/core.py) as it seems to be invalid: cannot import
name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/mathstuff.py) as it seems to be invalid: cannot
import name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
changed: [192.168.56.103]

TASK [nova : Start Nova] *****
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/core.py) as it seems to be invalid: cannot import
name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
[WARNING]: Skipping plugin (/usr/lib/python3/dist-packages/ansible/plugins/filter/mathstuff.py) as it seems to be invalid: cannot
import name 'environmentfilter' from 'jinja2.filters' (/home/ken/.local/lib/python3.10/site-packages/jinja2/filters.py)
changed: [192.168.56.103]

PLAY [CentOS] *****
skipping: no hosts matched

PLAY RECAP *****
192.168.56.103 : ok=6 changed=4 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Checking of status

Nova

```
ken@controlNode2:~$ service nova-compute status
● nova-compute.service - OpenStack Compute
   Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor
   Active: active (running) since Sun 2023-12-03 23:48:32 PST; 12h ago
   Main PID: 2483 (nova-compute)
     Tasks: 2 (limit: 4594)
    Memory: 55.7M
       CPU: 2.727s
    CGroup: /system.slice/nova-compute.service
           └─2483 /usr/bin/python3 /usr/bin/nova-compute --config-file=/etc/n>

Dec 03 23:48:32 controlNode2 systemd[1]: Started OpenStack Compute.
Dec 03 23:48:40 controlNode2 nova-compute[2483]: Modules with known eventlet mo>
lines 1-12/12 (END)
```

<p>Glance</p>	<pre> ken@controlNode2:~\$ service glance-api status ● glance-api.service - OpenStack Image Service API Loaded: loaded (/lib/systemd/system/glance-api.service; enabled; vendor pr Active: active (running) since Sun 2023-12-03 23:48:32 PST; 12h ago Docs: man:glance-api(1) Main PID: 2472 (glance-api) Tasks: 2 (limit: 4594) Memory: 52.7M CPU: 37.817s CGroup: /system.slice/glance-api.service └─2472 /usr/bin/python3 /usr/bin/glance-api --config-file=/etc/gla 3455 /usr/bin/python3 /usr/bin/glance-api --config-file=/etc/gla Dec 03 23:48:32 controlNode2 systemd[1]: Started OpenStack Image Service API. lines 1-13/13 (END) </pre>
<p>Keystone</p>	<pre> ken@controlNode2:~\$ sudo netstat -nltp egrep '5000 35357' tcp6 0 0 :::5000 :::* LISTEN 1497/apache2 ken@controlNode2:~\$ </pre>
<p>Reflections: Answer the following:</p> <ol style="list-style-type: none"> Describe Keystone, Glance and Nova services <ul style="list-style-type: none"> Keystone is the identity and access management service for OpenStack that authenticates and authorizes users, projects, and roles. Glance is the image service for Openstack that stores and manages disk and server images for creating new VM's Nova is the compute service for OpenStack that manages creation, operation, and deletion of VM's. <p>Together, these 3 services are the foundation for building and managing a cloud computing environment.</p>	
<p>Conclusions:</p> <p>In conclusion, this Hands-on Activity has made me learn about the different usage of OpenStack services especially Keystone, Glance, and Nova. I was able to implement the usage of these services where all play in my playbook was successful.</p>	