


Name: Tendencia, Jasmin Raiza S.	Date Performed: 12/03/2023
Course/Section: CPE31S4	Date Submitted: 12/09/2023
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st/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	
<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. Neutron b. Horizon c. Cinder d. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file. e. Add, commit and push it to your GitHub repo. 	
5. Output (screenshots and explanations)	
<ol style="list-style-type: none"> 1. Create a new repository in Github. 	

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 *

 jrstendencia

Repository name *

HOA15

HOA15 is available.

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

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.](#)



This will set `main` as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.


Create repository

 HOA15 Public

 Pin

 Unwatch 

 main


 1 branch

 0 tags


Go to file

Add file

 Code

 jrstendencia Initial commit

e56b478 now 1 commit

 README.md

Initial commit

now

README.md



HOA15

```
tendencia@workstation:~$ git clone git@github.com:jrstendencia/HOA15.git
Cloning into 'HOA15'...
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.
tendencia@workstation:~$ ls
Ansible_S4          HOA10  HOA2   HOA8_Tendencia  snap
CPE232_JasminTendencia HOA11  HOA4   HOA9            Templates
CPE_MIDEXAM_TENDENCIA HOA12  HOA5   HOA9_Final      Tendencia_PrelimExam
Desktop            HOA13  HOA6   Music           Videos
Documents          HOA14  HOA7   Pictures
Downloads          HOA15  HOA8   Public
```

- 2. Setup the ansible environment by creating an inventory file to specify the target hosts (Ubuntu and CentOS) to be configured. Also, create the ansible.cfg.

ansible.cfg:

```
[defaults]

inventory = inventory
host_key_checking = false

deprecation_warnings = false

remote_user = tendencia
private_key_file = ~/.ssh/
```

inventory:

Ubuntu	192.168.56.103 ansible_python_interpreter=/usr/bin/python3 [controller] 192.168.56.103 [compute] 192.168.56.103
CentOS	192.168.56.104 ansible_python_interpreter=/usr/bin/python3 [controller_node] 192.168.56.104

- 3. Create a role for Neutron, Horizon, and Cinder installation for Ubuntu and CentOS by generating the role structure. Then, create directories need that contains either a .conf or .yml file.

Ubuntu

```
tendencia@workstation:~/H0A15/OS/ubuntu$ tree
.
├── ansible.cfg
├── inventory
├── nhc.yml
├── roles
│   ├── cinder
│   │   ├── tasks
│   │   └── main.yml
│   ├── horizon
│   │   ├── tasks
│   │   └── main.yml
│   └── neutron
│       ├── tasks
│       └── main.yml
└── 7 directories, 6 files
```

CentOS

```
tendencia@workstation:~/H0A15/OS/centos$ tree
.
├── ansible.cfg
├── inventory
├── nhc.yml
├── roles
│   ├── cinder
│   │   ├── files
│   │   │   ├── cinder.conf
│   │   │   └── nova.conf
│   │   ├── handlers
│   │   │   └── main.yml
│   │   └── tasks
│   │       ├── configure.yml
│   │       ├── install.yml
│   │       ├── main.yml
│   │       └── prereq.yml
│   ├── horizon
│   │   ├── files
│   │   │   ├── local_settings
│   │   │   └── openstack-dashboard.conf
│   │   ├── handlers
│   │   │   └── main.yml
│   │   └── tasks
│   │       ├── configure.yml
│   │       ├── install.yml
│   │       └── main.yml
│   └── neutron
│       ├── files
│       │   ├── dhcp_agent.ini
│       │   ├── linuxbridge_agent.ini
│       │   ├── metadata_agent.ini
│       │   ├── ml2_conf.ini
│       │   ├── neutron.conf
│       │   └── nova.conf
│       └── tasks
│           ├── configure.yml
│           ├── install.yml
│           ├── main.yml
│           ├── network1.yml
│           └── prereq.yml
└── 12 directories, 27 files
```

- The scripts should define the tasks for Neutron, Horizon, and Cinder installation for both Ubuntu and CentOS.

Ubuntu:

	Neutron		
	Directory	File	Ansible Script
	tasks	main.yml	<pre>--- - name: install Neutron for Ubuntu tags: ubuntu apt: name: neutron-openvswitch-agent state: latest when: ansible_distribution == "Ubuntu" - name: "Neutron- Restarting/Enabling" service: name: nova-compute state: restarted when: ansible_distribution == "Ubuntu"</pre>
	Horizon		
	Directory	File	Ansible Script
	tasks	main.yml	<pre>--- - name: install Horizon for Ubuntu tags: ubuntu apt: name: openstack-dashboard state: latest when: ansible_distribution == "Ubuntu"</pre>
	Cinder		
	Directory	File	Ansible Script

	tasks	main.yml	<pre>--- - name: install Cinder for Ubuntu tags: ubuntu apt: name: cinder-backup state: latest when: ansible_distribution == "Ubuntu" - name: "Cinder- Restarting/Enabling" service: name: cinder-backup state: restarted when: ansible_distribution == "Ubuntu"</pre>
--	-------	----------	---

CentOS:

Cinder		
Directory	File	Ansible Script
files	cinder.conf	<pre>[DEFAULT] transport_url = rabbit://openstack:rabbitpass@controller # # From cinder # # The maximum number of items that a collection resource returns in a single # response (integer value) #osapi_max_limit = 1000 # # Json file indicating user visible filter parameters for list queries. (string # value) #resource_query_filters_file = /etc/cinder/resource_filters.json # # Treat X-Forwarded-For as the canonical remote address. Only enable this if # you have a sanitizing proxy. (boolean value) #use_forwarded_for = false # # Public url to use for versions endpoint. The default is None, which will use # the request's host_url attribute to populate the URL base. If Cinder is # operating behind a proxy, you will want to change this to represent the # proxy's URL. (string value) #public_endpoint = <None> # # Backup services use same backend. (boolean value) #backup_use_same_host = false # # Compression algorithm ("none" to disable) (string value) # Possible values: # none - <No description provided></pre>

		nova.conf	<pre>[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</pre>
	handlers	main.yml	<pre>- name: Restarting nova service: name: openstack-nova-api.service failed_when: false no_log: true</pre>
	tasks	configure.yml	<pre>- name: Configuring cinder config file copy: src: cinder.conf dest: /etc/cinder/cinder.conf owner: root group: cinder mode: 0640 - name: Configuring nova config file copy: src: nova.conf dest: /etc/nova/nova.conf owner: root group: nova mode: 640 notify: Restarting nova - name: Populating the Block Storage database become_user: root command: su -s /bin/sh -c "cinder-manage db sync" cinder failed_when: false no_log: true - name: Starting and enabling cinder service service: name: - openstack-cinder-api.service - openstack-cinder-scheduler.service state: started enabled: true failed_when: false no_log: true</pre>
		install.yml	<pre>- name: Installing cinder yum: name: openstack-cinder</pre>

		main.yml	<pre> - import_tasks: prereq.yml - import_tasks: install.yml - import_tasks: configure.yml </pre>
		prereq.yml	<pre> - name: Creating Cinder database mysql_query: login_user: root login_password: mysqlpass login_unix_socket: /var/lib/mysql/mysql.sock query: - CREATE DATABASE cinder; - GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'localhost' IDENTIFIED BY 'cinderpass'; - GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'%' IDENTIFIED BY 'cinderpass'; single_transaction: yes failed_when: false no_log: true - name: Sourcing the admin credentials command: source /home/cserver/admin-openrc failed_when: false no_log: true - name: Creating cinder user become_user: root expect: command: openstack user create --domain default --password-prompt cinder responses: "User Password": cinderpass "Repeat User Password": cinderpass failed_when: false no_log: true - name: Creating cinderv2 and cinderv3 service entities shell: openstack service create --name cinderv2 --description "OpenStack Block Storage" volumev2 openstack service create --name cinderv3 --description "OpenStack Block Storage" volumev3 failed_when: false no_log: true - name: Creating a block storage service API endpoints shell: openstack endpoint create --region RegionOne volumev2 public http://controller:8776/v2/%(tenant_id)s openstack endpoint create --region RegionOne volumev2 internal http://controller:8776/v2/%(tenant_id)s openstack endpoint create --region RegionOne volumev2 admin http://controller:8776/v2/%(tenant_id)s openstack endpoint create --region RegionOne volumev3 public http://controller:8776/v3/%(tenant_id)s openstack endpoint create --region RegionOne volumev3 internal http://controller:8776/v3/%(tenant_id)s openstack endpoint create --region RegionOne volumev3 admin http://controller:8776/v3/%(tenant_id)s failed_when: false no_log: true </pre>

Horizon		
Directory	File	Ansible Script
files	local_settings	<pre> # -*- coding: utf-8 -*- # NOTE: The default values of the settings are defined in # openstack_dashboard/defaults.py. Previously most available settings # were listed in this example file, but it is no longer true. # For available settings, see openstack_dashboard/defaults.py and # the horizon setting reference found at # https://docs.openstack.org/horizon/latest/configuration/settings.html. # # Django related settings and HORIZON_CONFIG still exist here. # Keep in my mind that they will be revisit in upcoming releases. # import os from django.utils.translation import ugettext_lazy as _ from openstack_dashboard.settings import HORIZON_CONFIG DEBUG = False # This setting controls whether or not compression is enabled. Disabling # compression makes Horizon considerably slower, but makes it much easier # to debug JS and CSS changes #COMPRESS_ENABLED = not DEBUG # This setting controls whether compression happens on the fly, or offline # with 'python manage.py compress' # See https://django-compressor.readthedocs.io/en/latest/usage/#offline-compression # for more information #COMPRESS_OFFLINE = not DEBUG # If horizon is running in production (DEBUG is False), set this # with the list of host/domain names that the application can serve. # For more information see: # https://docs.djangoproject.com/en/dev/ref/settings/#allowed-hosts ALLOWED_HOSTS = ['*'] # Set SSL proxy settings: # Pass this header from the proxy after terminating the SSL, # and don't forget to strip it from the client's request. # For more information see: # https://docs.djangoproject.com/en/dev/ref/settings/#secure-proxy-ssl-header #SECURE_PROXY_SSL_HEADER = ('HTTP_X_FORWARDED_PROTO', 'https') # If Horizon is being served through SSL, then uncomment the following two Read 414 lines </pre>

		openstack-dashboard.conf	<pre> WSGIDaemonProcess dashboard WSGIProcessGroup dashboard WSGISocketPrefix run/wsgi WSGIScriptAlias /dashboard /usr/share/openstack-dashboard/openstack_dashboard/wsgi/django.wsgi Alias /dashboard/static /usr/share/openstack-dashboard/static <Directory /usr/share/openstack-dashboard/openstack_dashboard/wsgi> Options All AllowOverride All Require all granted </Directory> <Directory /usr/share/openstack-dashboard/static> Options All AllowOverride All Require all granted </Directory> </pre>
	handlers	main.yml	<pre> - name: Restarting httpd and memcached service: name: - httpd - memcached state: restarted failed_when: false no_log: true </pre>
	tasks	configure.yml	<pre> - name: Configuring dashboard local settings copy: src: local_settings dest: /etc/openstack-dashboard owner: root group: apache mode: 640 notify: Restarting httpd and memcached - name: Configuring openstack dashboard copy: src: openstack-dashboard.conf dest: /etc/httpd/conf.d/ owner: root group: root mode: 0644 notify: Restarting httpd and memcached </pre>
		install.yml	<pre> - name: Installing dashboard yum: name: openstack-dashboard </pre>
		main.yml	<pre> - import_tasks: install.yml - import_tasks: configure.yml </pre>
	Neutron		
	Directory	File	Ansible Script

	files	dhcp_agent.ini	<pre>[DEFAULT] interface_driver = linuxbridge dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq enable_isolated_metadata = true # # From oslo.log # # If set to true, the logging level will be set to DEBUG instead of the default # INFO level. (boolean value) # Note: This option can be changed without restarting. #debug = false # The name of a logging configuration file. This file is appended to any # existing logging configuration files. For details about logging configuration # files, see the Python logging module documentation. Note that when logging # configuration files are used then all logging configuration is set in the # configuration file and other logging configuration options are ignored (for # example, log-date-format). (string value) # Note: This option can be changed without restarting. # Deprecated group/name - [DEFAULT]/log_config #log_config_append = <None> # Defines the format string for %(asctime)s in log records. Default: # %(default)s . This option is ignored if log_config_append is set. (string # value) #log_date_format = %Y-%m-%d %H:%M:%S # (Optional) Name of log file to send logging output to. If no default is set, # logging will go to stderr as defined by use_stderr. This option is ignored if # log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logfile #log_file = <None> # (Optional) The base directory used for relative log_file paths. This option # is ignored if log_config_append is set. (string value)</pre>
		linuxbridge_agent.ini	<pre>[DEFAULT] # # From oslo.log # # If set to true, the logging level will be set to DEBUG instead of the default # INFO level. (boolean value) # Note: This option can be changed without restarting. #debug = false # The name of a logging configuration file. This file is appended to any # existing logging configuration files. For details about logging configuration # files, see the Python logging module documentation. Note that when logging # configuration files are used then all logging configuration is set in the # configuration file and other logging configuration options are ignored (for # example, log-date-format). (string value) # Note: This option can be changed without restarting. # Deprecated group/name - [DEFAULT]/log_config #log_config_append = <None> # Defines the format string for %(asctime)s in log records. Default: # %(default)s . This option is ignored if log_config_append is set. (string # value) #log_date_format = %Y-%m-%d %H:%M:%S # (Optional) Name of log file to send logging output to. If no default is set, # logging will go to stderr as defined by use_stderr. This option is ignored if # log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logfile #log_file = <None> # (Optional) The base directory used for relative log_file paths. This option # is ignored if log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logdir #log_dir = <None> # Uses logging handler designed to watch file system. When log file is moved or</pre>
		metadata_agent.ini	<pre>[DEFAULT] nova_metadata_host = controller metadata_proxy_shared_secret = METADATA_SECRET # # From oslo.log # # If set to true, the logging level will be set to DEBUG instead of the default # INFO level. (boolean value) # Note: This option can be changed without restarting. #debug = false # The name of a logging configuration file. This file is appended to any # existing logging configuration files. For details about logging configuration # files, see the Python logging module documentation. Note that when logging # configuration files are used then all logging configuration is set in the # configuration file and other logging configuration options are ignored (for # example, log-date-format). (string value) # Note: This option can be changed without restarting. # Deprecated group/name - [DEFAULT]/log_config #log_config_append = <None> # Defines the format string for %(asctime)s in log records. Default: # %(default)s . This option is ignored if log_config_append is set. (string # value) #log_date_format = %Y-%m-%d %H:%M:%S # (Optional) Name of log file to send logging output to. If no default is set, # logging will go to stderr as defined by use_stderr. This option is ignored if # log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logfile #log_file = <None> # (Optional) The base directory used for relative log_file paths. This option # is ignored if log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logdir</pre>

		ml2_conf.ini	<pre> [DEFAULT] # # From oslo.log # # If set to true, the logging level will be set to DEBUG instead of the default # INFO level. (boolean value) # Note: This option can be changed without restarting. #debug = false # The name of a logging configuration file. This file is appended to any # existing logging configuration files. For details about logging configuration # files, see the Python logging module documentation. Note that when logging # configuration files are used then all logging configuration is set in the # configuration file and other logging configuration options are ignored (for # example, log-date-format). (string value) # Note: This option can be changed without restarting. # Deprecated group/name - [DEFAULT]/log_config #log_config_append = <None> # Defines the format string for %(asctime)s in log records. Default: # %(default)s . This option is ignored if log_config_append is set. (string # value) #log_date_format = %Y-%m-%d %H:%M:%S # (Optional) Name of log file to send logging output to. If no default is set, # logging will go to stderr as defined by use_stderr. This option is ignored if # log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logfile #log_file = <None> # (Optional) The base directory used for relative log_file paths. This option # is ignored if log_config_append is set. (string value) # Deprecated group/name - [DEFAULT]/logdir #log_dir = <None> # Uses logging handler designed to watch file system. When log file is moved or </pre>
		neutron.conf	<pre> [DEFAULT] core_plugin = ml2 service_plugins = transport_url = rabbit://openstack:rabbitpass@controller auth_strategy = keystone notify_nova_on_port_status_changes = true notify_nova_on_port_data_changes = true # # From oslo.log # # If set to true, the logging level will be set to DEBUG instead of the default # INFO level. (boolean value) # Note: This option can be changed without restarting. #debug = false # The name of a logging configuration file. This file is appended to any # existing logging configuration files. For details about logging configuration # files, see the Python logging module documentation. Note that when logging # configuration files are used then all logging configuration is set in the # configuration file and other logging configuration options are ignored (for # example, log-date-format). (string value) # Note: This option can be changed without restarting. # Deprecated group/name - [DEFAULT]/log_config #log_config_append = <None> # Defines the format string for %(asctime)s in log records. Default: # %(default)s . This option is ignored if log_config_append is set. (string # value) #log_date_format = %Y-%m-%d %H:%M:%S # (Optional) Name of log file to send logging output to. If no default is set, # logging will go to stderr as defined by use_stderr. This option is ignored if </pre>
		nova.conf	<pre> # Show more detailed information as part of the response. Security note: # Enabling this option may expose sensitive details about the service being # monitored. Be sure to verify that it will not violate your security policies # (boolean value) #detailed=false # Additional backends that can perform health checks and report that information # back as part of a request (list value) #backends = # Check the presence of a file to determine if an application is running on a # port. Used by DisableFileHealthcheck plugin (string value) #disable_by_file_path=<None> # Check the presence of a file based on a port to determine if an application is # running on a port. Expects a "port:path" list of strings. Used by # DisableFilesPortsHealthcheck plugin (list value) #disable_by_file_paths = [hyperv] # # The hyperv feature allows you to configure the Hyper-V hypervisor # driver to be used within an OpenStack deployment. # # From nova.conf # # # Dynamic memory ratio. For more information, refer to the documentation. # (floating point value) #dynamic_memory_ratio=1.0 # # Enable instance metrics collection. For more information, refer to the # documentation. (boolean value) #enable_instance_metrics_collection=false </pre>

	tasks	configure.yml	<pre> - name: Configuring metadata_agent.ini file copy: src: metadata_agent.ini dest: /etc/neutron/metadata_agent.ini owner: root group: neutron mode: 040 - name: Configuring nova.conf file copy: src: nova.conf dest: /etc/nova/nova.conf owner: root group: nova mode: 040 - name: Creating a link command: ln -s /etc/neutron/plugins/ml2/ml2_conf.ini /etc/neutron/plugin.ini failed_when: false no_log: true - name: Populating the database become_user: root command: db -s /bin/sh -c "neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_conf.ini upgrade head" failed_when: false no_log: true - name: Restarting the nova-api service service: name: openstack-nova-api.service state: restarted failed_when: false no_log: true - name: Starting and enabling the Networking services service: name: - neutron-server.service - neutron-linuxbridge-agent - neutron-dhcp-agent - neutron-metadata-agent state: started enabled: true failed_when: false no_log: true </pre>
		install.yml	<pre> - name: Installing neutron and its dependencies yum: name: - openstack-neutron - openstack-neutron-ml2 - openstack-neutron-linuxbridge - ebtables </pre>
		main.yml	<pre> - import_tasks: prereq.yml - import_tasks: install.yml - import_tasks: network1.yml - import_tasks: configure.yml </pre>
		network1.yml	<pre> - name: Editing neutron config file copy: src: neutron.conf dest: /etc/neutron/neutron.conf owner: root group: neutron mode: 0640 - name: Editing ml2_conf.ini file copy: src: ml2_conf.ini dest: /etc/neutron/plugins/ml2/ml2_conf.ini owner: root group: neutron mode: 0640 - name: Editing neutron config file copy: src: linuxbridge_agent.ini dest: /etc/neutron/plugins/ml2/linuxbridge_agent.ini owner: root group: neutron mode: 0640 - name: Editing neutron config file copy: src: dhcp_agent.ini dest: /etc/neutron/dhcp_agent.ini owner: root group: neutron mode: 0640 </pre>

		prereq.yml	<pre> - name: Creating neutron database mysql_query: login_user: root login_password: mysqlpass login_unix_socket: /var/lib/mysql/mysql.sock query: - CREATE DATABASE neutron; - GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'localhost' IDENTIFIED BY 'neutronpass'; - GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'%' IDENTIFIED BY 'neutronpass'; single_transaction: yes failed_when: false no_log: true - name: Sourcing the admin credentials command: source /home/cserver/admin-openrc failed_when: false no_log: true - name: Creating neutron user become_user: root expect: command: openstack user create --domain default --password-prompt neutron responses: "User Password": neutronpass "Repeat User Password": neutronpass failed_when: false no_log: true - name: Adding admin roles to the neutron user command: openstack role add --project service --user neutron admin failed_when: false no_log: true - name: Creating neutron service entity command: openstack service create --name neutron --description "OpenStack Networking" network failed_when: false no_log: true - name: Creating the network service API endpoints shell: openstack endpoint create --region RegionOne network public http://controller:9696 openstack endpoint create --region RegionOne network internal http://controller:9696 openstack endpoint create --region RegionOne network admin http://controller:9696 failed_when: false no_log: true </pre>
--	--	------------	--

5. Create a playbook in the current working directory. This playbook will use the created roles for Neutron, Horizon, and Cinder.

Ubuntu		<pre> --- - hosts: all become: true tasks: - name: install updates (Ubuntu) tags: always apt: upgrade: dist update_cache: yes when: ansible_distribution == "Ubuntu" - hosts: compute become: true roles: - neutron - horizon - hosts: controller become: true roles: - cinder </pre>
--------	--	--

CentOS

```
---
- hosts: all
  become: true
  pre_tasks:
    - name: Updating and upgrading the operating system
      yum:
        name: "*"
        state: latest
        update_cache: true
- hosts: controller_node
  become: true
  roles:
    - neutron
    - horizon
    - cinder
```

6. Run the playbook to install on the target hosts using the command *ansible-playbook --ask-become-pass name-of-playbook.yml*.

Ubuntu:

```
tendencia@workstation:~/HOA15/OS/ubuntu$ ansible-playbook --ask-become-pass nhc.yml
BECOME password:

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

TASK [install updates (Ubuntu)] *****
ok: [192.168.56.103]

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

TASK [neutron : install Neutron for Ubuntu] *****
changed: [192.168.56.103]

TASK [neutron : Neutron- Restarting/Enabling] *****
changed: [192.168.56.103]

TASK [horizon : install Horizon for Ubuntu] *****
changed: [192.168.56.103]

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

TASK [cinder : install Cinder for Ubuntu] *****
changed: [192.168.56.103]

TASK [cinder : Cinder- Restarting/Enabling] *****
changed: [192.168.56.103]

PLAY RECAP *****
192.168.56.103      : ok=9   changed=5   unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

tendencia@workstation:~/HOA15/OS/ubuntu$
```

CentOS:

```
tendencia@workstation:~/H0A15/05/centos$ ansible-playbook --ask-become-pass nhc.yml  
BECOME password:
```

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

```
TASK [Gathering Facts] *****  
ok: [192.168.56.104]
```

```
TASK [Updating and upgrading the operating system] *****  
ok: [192.168.56.104]
```

```
PLAY [controller_node] *****
```

```
TASK [Gathering Facts] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Creating neutron database] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Sourcing the admin credentials] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Creating neutron user] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Adding admin roles to the neutron user] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Creating neutron service entity] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Creating the network service API endpoints] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Installing neutron and its dependencies] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Editing neutron config file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Editing ml2_conf.ini file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Editing neutron config file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Editing neutron config file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Configuring metadata_agent.ini file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Configuring nova.conf file] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Creating a link] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Populating the database] *****  
changed: [192.168.56.104]
```

```
TASK [neutron : Restarting the nova-api service] *****  
ok: [192.168.56.104]
```

```
TASK [neutron : Starting and enabling the Networking services] *****  
ok: [192.168.56.104]
```

```
TASK [horizon : Installing dashboard] *****  
changed: [192.168.56.104]
```

```
TASK [horizon : Configuring dashboard local settings] *****  
changed: [192.168.56.104]
```

```

TASK [horizon : Configuring openstack dashboard] *****
changed: [192.168.56.104]

TASK [cinder : Creating cinder database] *****
ok: [192.168.56.104]

TASK [cinder : Sourcing the admin credentials] *****
ok: [192.168.56.104]

TASK [cinder : Creating cinder user] *****
ok: [192.168.56.104]

TASK [cinder : Creating cinderv2 and cinderv3 service entities] *****
changed: [192.168.56.104]

TASK [cinder : Creating a block storage service API endpoints] *****
changed: [192.168.56.104]

TASK [cinder : Installing cinder] *****
changed: [192.168.56.104]

TASK [cinder : Configuring cinder config file] *****
changed: [192.168.56.104]

TASK [cinder : Configuring nova config file] *****
changed: [192.168.56.104]

TASK [cinder : Populating the Block Storage database] *****
changed: [192.168.56.104]

TASK [cinder : Starting and enabling cinder service] *****
ok: [192.168.56.104]

RUNNING HANDLER [horizon : Restarting httpd and memcached] *****
ok: [192.168.56.104]

RUNNING HANDLER [cinder : Restarting nova] *****
ok: [192.168.56.104]

PLAY RECAP *****
192.168.56.104 : ok=35 changed=19 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

tendencia@workstation:~/HOA15/05/centos$

```

7. Verification of installation in both Ubuntu and CentOS by using the command `systemctl status <>`.

Ubuntu

Neutron

```

tendencia@server2:~$ sudo 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: enable
   Active: active (running) since Sat 2023-12-09 22:54:44 PST; 19s ago
     Docs: man:neutron-metadata-agent(1)
    Main PID: 13561 (neutron-metadata)
      Tasks: 3 (limit: 3432)
     Memory: 123.5M
        CPU: 3.019s
    CGroup: /system.slice/neutron-metadata-agent.service
            └─13561 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --confi
              13591 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --confi
              13592 "neutron-metadata-agent (/usr/bin/python3 /usr/bin/neutron-metadata-agent --confi

Dec 09 22:54:44 server2 systemd[1]: Stopped OpenStack Neutron Metadata Agent.
Dec 09 22:54:44 server2 systemd[1]: neutron-metadata-agent.service: Consumed 15.740s CPU time.
Dec 09 22:54:44 server2 systemd[1]: Started OpenStack Neutron Metadata Agent.
[lines 1-16/16 (END)]

```


Horizon

```
tendencia@server2:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
tendencia@server2:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-09 23:06:21 PST; 23s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 19152 (apache2)
      Tasks: 85 (limit: 3432)
     Memory: 95.9M
        CPU: 1.236s
    CGroup: /system.slice/apache2.service
            └─19152 /usr/sbin/apache2 -k start
              └─19154 "(wsgi:cinder-wsgi" -k start
                └─19155 "(wsgi:cinder-wsgi" -k start
                  └─19156 "(wsgi:cinder-wsgi" -k start
                    └─19157 "(wsgi:cinder-wsgi" -k start
                      └─19158 "(wsgi:cinder-wsgi" -k start
                        └─19159 "(wsgi:horizon) " -k start
                          └─19160 "(wsgi:horizon) " -k start
                            └─19161 "(wsgi:horizon) " -k start
                              └─19162 "(wsgi:keystone-pu" -k start
                                └─19163 "(wsgi:keystone-pu" -k start
                                  └─19164 "(wsgi:keystone-pu" -k start
                                    └─19165 "(wsgi:keystone-pu" -k start
                                      └─19166 "(wsgi:keystone-pu" -k start
                                        └─19168 /usr/sbin/apache2 -k start
                                          └─19169 /usr/sbin/apache2 -k start
                                            └─19170 /usr/sbin/apache2 -k start
                                              └─19171 /usr/sbin/apache2 -k start
                                                └─19178 /usr/sbin/apache2 -k start

Dec 09 23:06:21 server2 systemd[1]: Starting The Apache HTTP Server...
Dec 09 23:06:21 server2 apachectl[19147]: AH00558: apache2: Could not reliably determine the server's
Dec 09 23:06:21 server2 systemd[1]: Started The Apache HTTP Server.
lines 1-32/32 (END)
```

Cinder

```
tendencia@server2:~$ sudo systemctl status cinder-volume
● cinder-volume.service - OpenStack Cinder Volume
   Loaded: loaded (/lib/systemd/system/cinder-volume.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-12-09 23:02:26 PST; 15s ago
     Docs: man:cinder-volume(1)
    Main PID: 17391 (cinder-volume)
      Tasks: 2 (limit: 3432)
     Memory: 148.2M
        CPU: 4.849s
    CGroup: /system.slice/cinder-volume.service
            └─17391 /usr/bin/python3 /usr/bin/cinder-volume --config-file=/etc/cinder/cinder.conf
              └─17427 /usr/bin/python3 /usr/bin/cinder-volume --config-file=/etc/cinder/cinder.conf

Dec 09 23:02:29 server2 cinder-volume[17391]: num_down_hosts = column_property(
Dec 09 23:02:31 server2 sudo[17431]: cinder : PMD=/var/lib/cinder ; USER=root ; COMMAND=/usr/bin/c
Dec 09 23:02:31 server2 sudo[17431]: pam_unix(sudo:session): session opened for user root(uid=0) by
Dec 09 23:02:32 server2 sudo[17431]: pam_unix(sudo:session): session closed for user root
Dec 09 23:02:33 server2 sudo[17464]: cinder : PMD=/var/lib/cinder ; USER=root ; COMMAND=/usr/bin/c
Dec 09 23:02:33 server2 sudo[17464]: pam_unix(sudo:session): session opened for user root(uid=0) by
Dec 09 23:02:35 server2 sudo[17464]: pam_unix(sudo:session): session closed for user root
Dec 09 23:02:37 server2 sudo[17585]: cinder : PMD=/var/lib/cinder ; USER=root ; COMMAND=/usr/bin/c
Dec 09 23:02:37 server2 sudo[17585]: pam_unix(sudo:session): session opened for user root(uid=0) by
Dec 09 23:02:38 server2 sudo[17585]: pam_unix(sudo:session): session closed for user root
lines 1-22/22 (END)
```

CentOS

Horizon

```
[tendencia@controller ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Drop-In: /usr/lib/systemd/system/httpd.service.d
            └─openstack-dashboard.conf
   Active: active (running) since Sat 2023-12-09 21:44:33 PST; 1h 40min ago
     Docs: man:httpd(8)
            man:apachectl(8)
    Main PID: 1545 (httpd)
      Status: "Total requests: 18; Current requests/sec: 0; Current traffic: 0 B/sec"
      Tasks: 7
     CGroup: /system.slice/httpd.service
            └─1545 /usr/sbin/httpd -DFOREGROUND
              └─1791 /usr/sbin/httpd -DFOREGROUND
                └─1793 /usr/sbin/httpd -DFOREGROUND
                  └─1795 /usr/sbin/httpd -DFOREGROUND
                    └─1796 /usr/sbin/httpd -DFOREGROUND
                      └─1797 /usr/sbin/httpd -DFOREGROUND
                        └─5707 /usr/sbin/httpd -DFOREGROUND

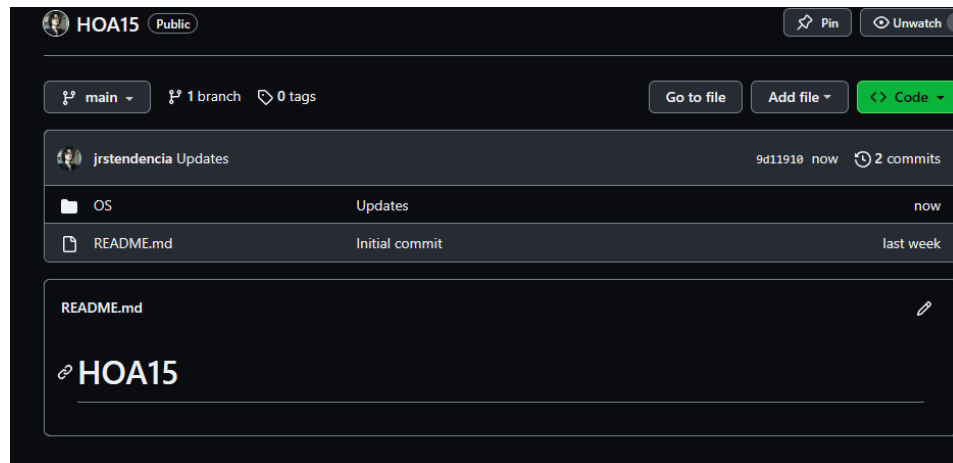
Dec 09 21:44:31 controller systemd[1]: Starting The Apache HTTP S...
Dec 09 21:44:33 controller systemd[1]: Started The Apache HTTP Se...
Hint: Some lines were ellipsized, use -l to show in full.
```

Cinder

```
[tendencia@controller ~]$ sudo systemctl status openstack-cinder-api
● openstack-cinder-api.service - OpenStack Cinder API Server
   Loaded: loaded (/usr/lib/systemd/system/openstack-cinder-api.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2023-12-09 23:47:47 PST; 1min 0s ago
     Main PID: 5461 (cinder-api)
       Tasks: 1
      CGroup: /system.slice/openstack-cinder-api.service
              └─5461 /usr/bin/python2 /usr/bin/cinder-api --config-file /usr/sha...

Dec 09 23:47:47 controller systemd[1]: Started OpenStack Cinder API Server.
Dec 09 23:47:57 controller cinder-api[5461]: Deprecated: Option "logdir" fro...
Hint: Some lines were ellipsized, use -l to show in full.
[tendencia@controller ~]$
```

8. Sync changes in github.



Reflections:

Answer the following:

1. Describe Neutron, Horizon and Cinder services

- Within the OpenStack framework, Neutron functions as a crucial service, providing network connectivity to other OpenStack services. This service enables users to effortlessly create, manage, and configure networks, subnets, routers, and interfaces attached to these networks. Horizon, the web-based dashboard for OpenStack, complements this functionality by offering a graphical user interface for users to interact with various OpenStack services, including Neutron. Additionally, Cinder, an essential OpenStack block storage service, plays a pivotal role in furnishing persistent storage for the broader OpenStack ecosystem. Users can create, attach, and manage block storage volumes and snapshots through Cinder, utilizing these volumes either as primary storage for instances or as supplementary storage attached to instances.

Conclusions:

In summary, utilizing Ansible as the Infrastructure as Code (IaC) tool within an OpenStack environment offers numerous advantages. The straightforward, declarative language and agentless architecture of Ansible simplify the management

and deployment of intricate infrastructure. Its seamless integration with other OpenStack components enables an automated end-to-end workflow. Neutron, Horizon, and Cinder play pivotal roles in the OpenStack cloud computing platform. Neutron delivers network connectivity as a service for other OpenStack services, empowering users to create and oversee networks, subnets, and routers. Horizon serves as the web-based dashboard, offering a graphical interface for users to oversee and interact with diverse OpenStack services. Meanwhile, Cinder delivers block storage services, enabling users to generate, attach, and manage persistent storage volumes and snapshots. Collectively, these components constitute a potent and adaptable platform for consistently and automatically managing cloud infrastructure.