Name: Abigail Laxamana	Date Performed: November 29, 2023
Course/Section: CPE 232 - CPE31S6	Date Submitted: November 30, 2023
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st sem, SY: 2023-2024

Activity 13: OpenStack Prerequisite Installation

1. Objectives

Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).

2. Intended Learning Outcomes

1. Analyze the advantages and disadvantages of cloud services
2. Evaluate different Cloud deployment and service models
3. Create a workflow to install and configure OpenStack base services using Ansible as documentation and execution.

3. Resources

Oracle VirtualBox (Hypervisor)

1x Ubuntu VM or Centos VM

4. Tasks

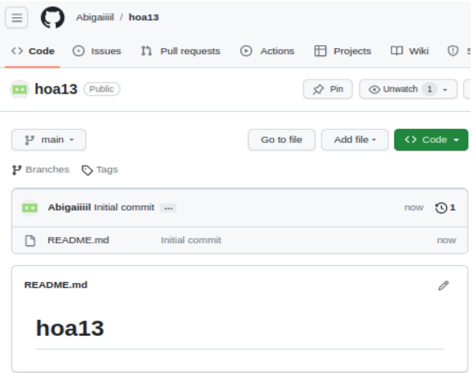
1. Create a new repository for this activity.
2. Create a playbook that converts the steps in the following items in https://docs.openstack.org/install-guide/
   a. NTP
   b. OpenStack packages
   c. SQL Database
   d. Message Queue
   e. Memcached
   f. Etcd

g. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in Inventory file.

h. Add, commit and push it to your GitHub repo.

5. Output (screenshots and explanations)

Step 1. Create a new repository for this activity in Github.



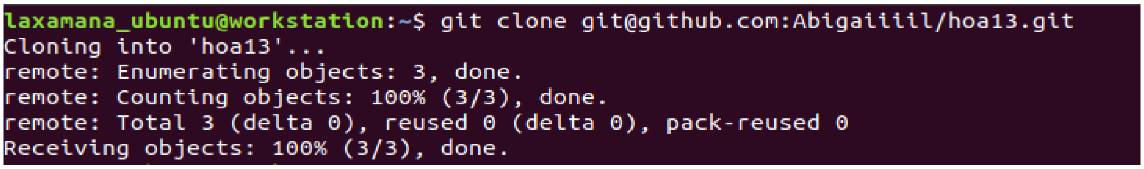
Abigaiiiil Initial commit ... now 1

README.md Initial commit now

README.md

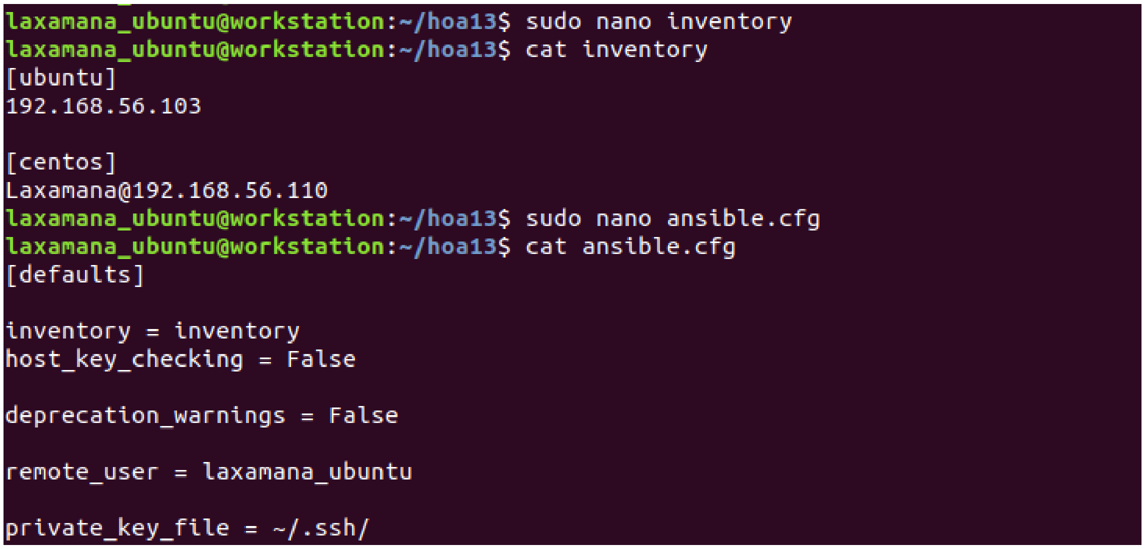
hoa13

Step 2. clone your repository to your workstation using **git clone** command.



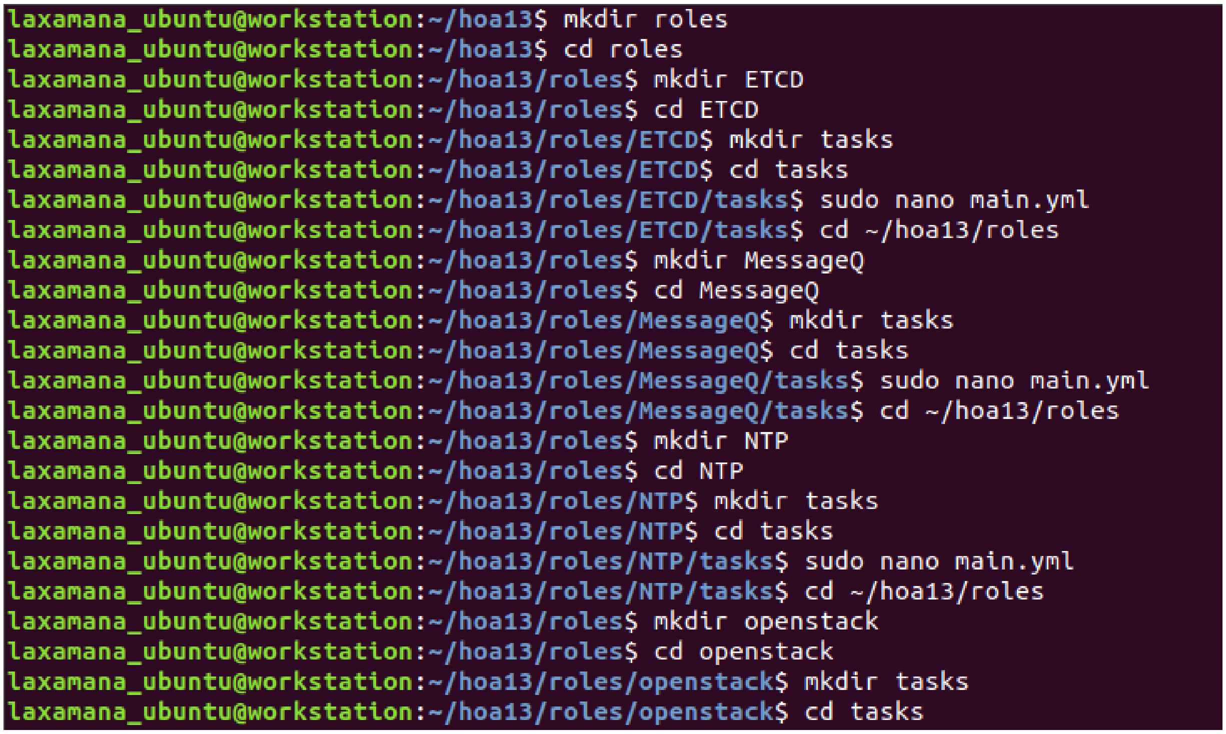
```
laxamana_ubuntu@workstation:~$ git clone git@github.com:Abigaiiiil/hoa13.git
Cloning into 'hoa13'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

Step3. Create an inventory file that contains the addresses of the servers to be used, and create an ansible file which contains the settings.

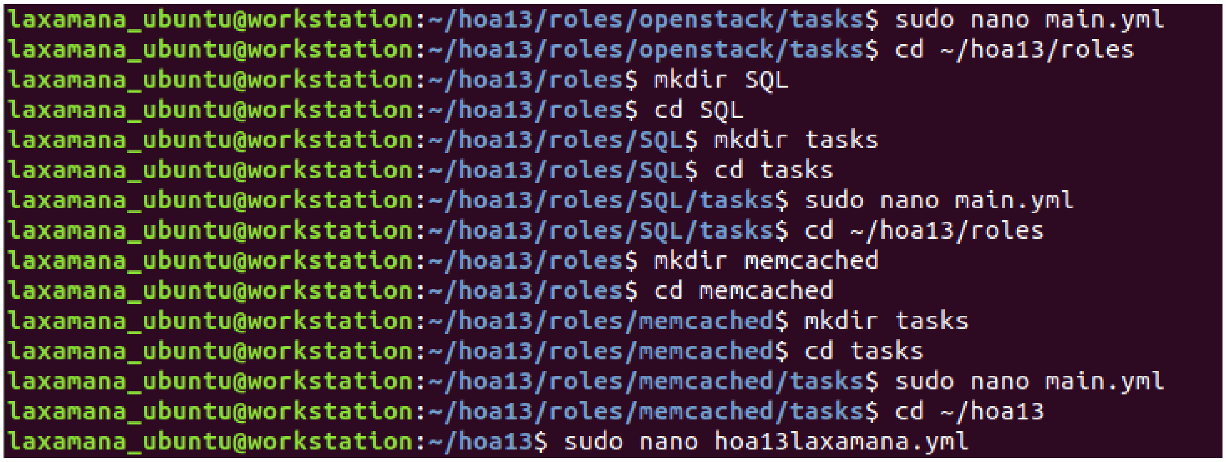


```
laxamana_ubuntu@workstation:~/hoa13$ sudo nano inventory
laxamana_ubuntu@workstation:~/hoa13$ cat inventory
[ubuntu]
192.168.56.103

[centos]
Laxamana@192.168.56.110
laxamana_ubuntu@workstation:~/hoa13$ sudo nano ansible.cfg
laxamana_ubuntu@workstation:~/hoa13$ cat ansible.cfg
[defaults]

inventory = inventory
host_key_checking = False

deprecation_warnings = False

remote_user = laxamana_ubuntu

private_key_file = ~/.ssh/
```

Step 4. Create roles and files/folders for each prerequisite. Inside those, create a directory named **tasks** that contain **main.yml** file.

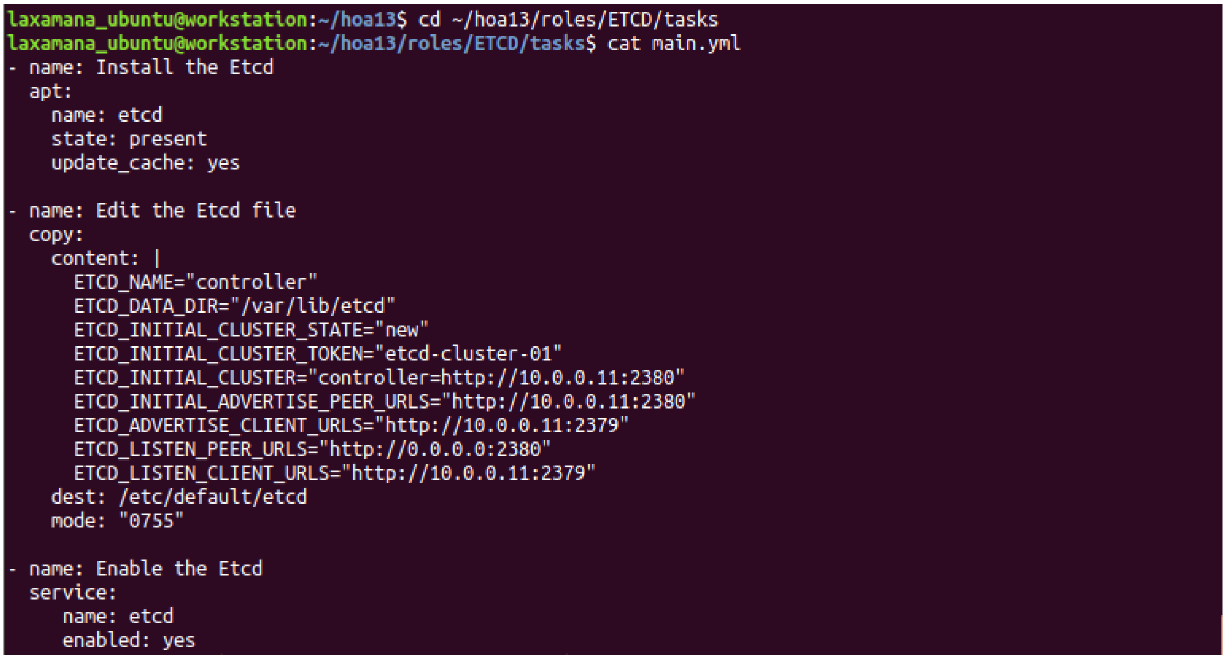


```
laxamana_ubuntu@workstation:~/hoa13$ mkdir roles
laxamana_ubuntu@workstation:~/hoa13$ cd roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir ETCD
laxamana_ubuntu@workstation:~/hoa13/roles$ cd ETCD
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD$ cd tasks
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD/tasks$ cd ~/hoa13/roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir MessageQ
laxamana_ubuntu@workstation:~/hoa13/roles$ cd MessageQ
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ$ cd tasks
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ/tasks$ cd ~/hoa13/roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir NTP
laxamana_ubuntu@workstation:~/hoa13/roles$ cd NTP
laxamana_ubuntu@workstation:~/hoa13/roles/NTP$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/NTP$ cd tasks
laxamana_ubuntu@workstation:~/hoa13/roles/NTP/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/NTP/tasks$ cd ~/hoa13/roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir openstack
laxamana_ubuntu@workstation:~/hoa13/roles$ cd openstack
laxamana_ubuntu@workstation:~/hoa13/roles/openstack$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/openstack$ cd tasks
```



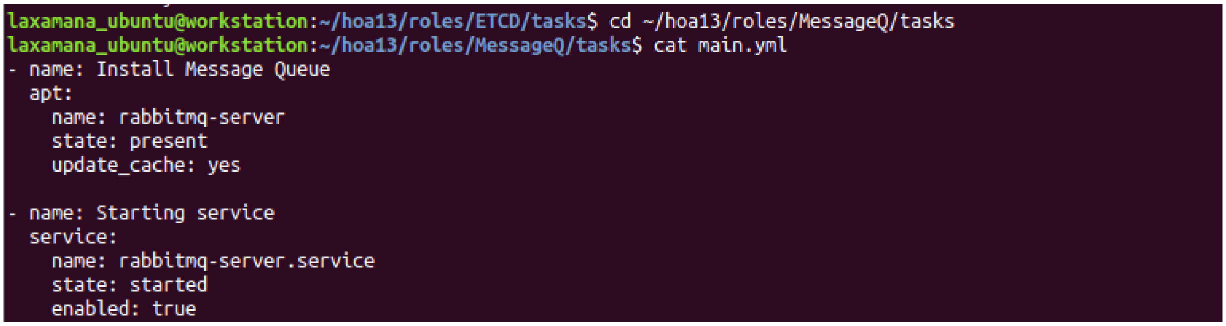
```
laxamana_ubuntu@workstation:~/hoa13/roles/openstack/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/openstack/tasks$ cd ~/hoa13/roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir SQL
laxamana_ubuntu@workstation:~/hoa13/roles$ cd SQL
laxamana_ubuntu@workstation:~/hoa13/roles/SQL$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/SQL$ cd tasks
laxamana_ubuntu@workstation:~/hoa13/roles/SQL/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/SQL/tasks$ cd ~/hoa13/roles
laxamana_ubuntu@workstation:~/hoa13/roles$ mkdir memcached
laxamana_ubuntu@workstation:~/hoa13/roles$ cd memcached
laxamana_ubuntu@workstation:~/hoa13/roles/memcached$ mkdir tasks
laxamana_ubuntu@workstation:~/hoa13/roles/memcached$ cd tasks
laxamana_ubuntu@workstation:~/hoa13/roles/memcached/tasks$ sudo nano main.yml
laxamana_ubuntu@workstation:~/hoa13/roles/memcached/tasks$ cd ~/hoa13
laxamana_ubuntu@workstation:~/hoa13$ sudo nano hoa13laxamana.yml
```

ETCD main.yml



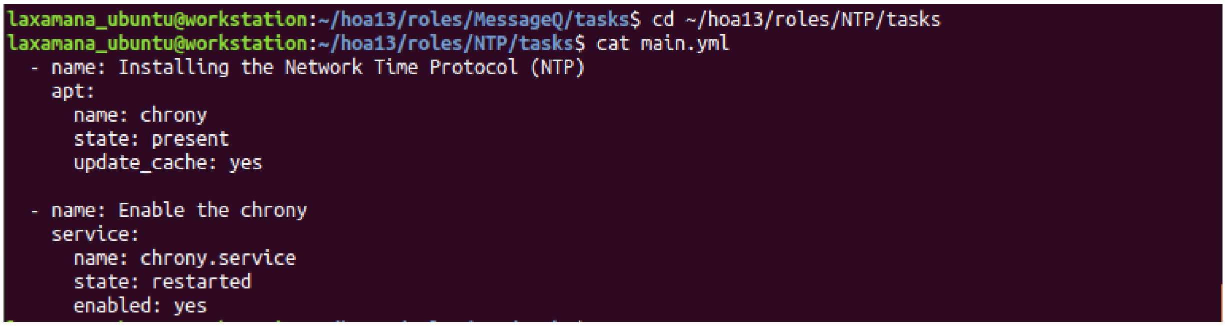
```
laxamana_ubuntu@workstation:~/hoa13$ cd ~/hoa13/roles/ETCD/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD/tasks$ cat main.yml
- name: Install the Etcd
  apt:
    name: etcd
    state: present
    update_cache: yes

- name: Edit the Etcd file
  copy:
    content: |
      ETCD_NAME="controller"
      ETCD_DATA_DIR="/var/lib/etcd"
      ETCD_INITIAL_CLUSTER_STATE="new"
      ETCD_INITIAL_CLUSTER_TOKEN="etcd-cluster-01"
      ETCD_INITIAL_CLUSTER="controller=http://10.0.0.11:2380"
      ETCD_INITIAL_ADVERTISE_PEER_URLS="http://10.0.0.11:2380"
      ETCD_ADVERTISE_CLIENT_URLS="http://10.0.0.11:2379"
      ETCD_LISTEN_PEER_URLS="http://0.0.0.0:2380"
      ETCD_LISTEN_CLIENT_URLS="http://10.0.0.11:2379"
    dest: /etc/default/etcd
    mode: "0755"

- name: Enable the Etcd
  service:
     name: etcd
     enabled: yes
```

MessageQ main.yml



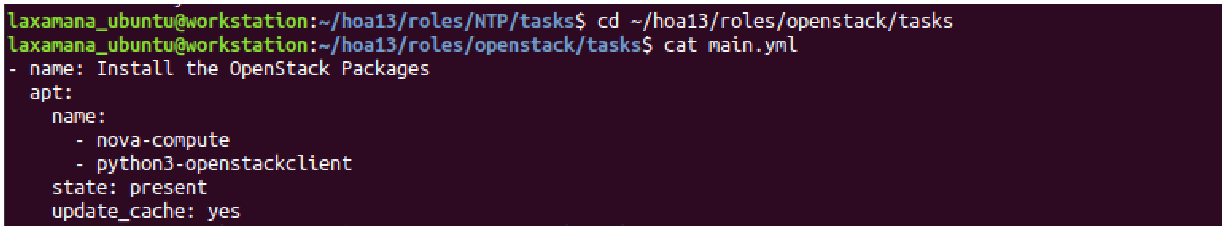
```
laxamana_ubuntu@workstation:~/hoa13/roles/ETCD/tasks$ cd ~/hoa13/roles/MessageQ/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ/tasks$ cat main.yml
- name: Install Message Queue
  apt:
    name: rabbitmq-server
    state: present
    update_cache: yes

- name: Starting service
  service:
    name: rabbitmq-server.service
    state: started
    enabled: true
```

NTP main.yml



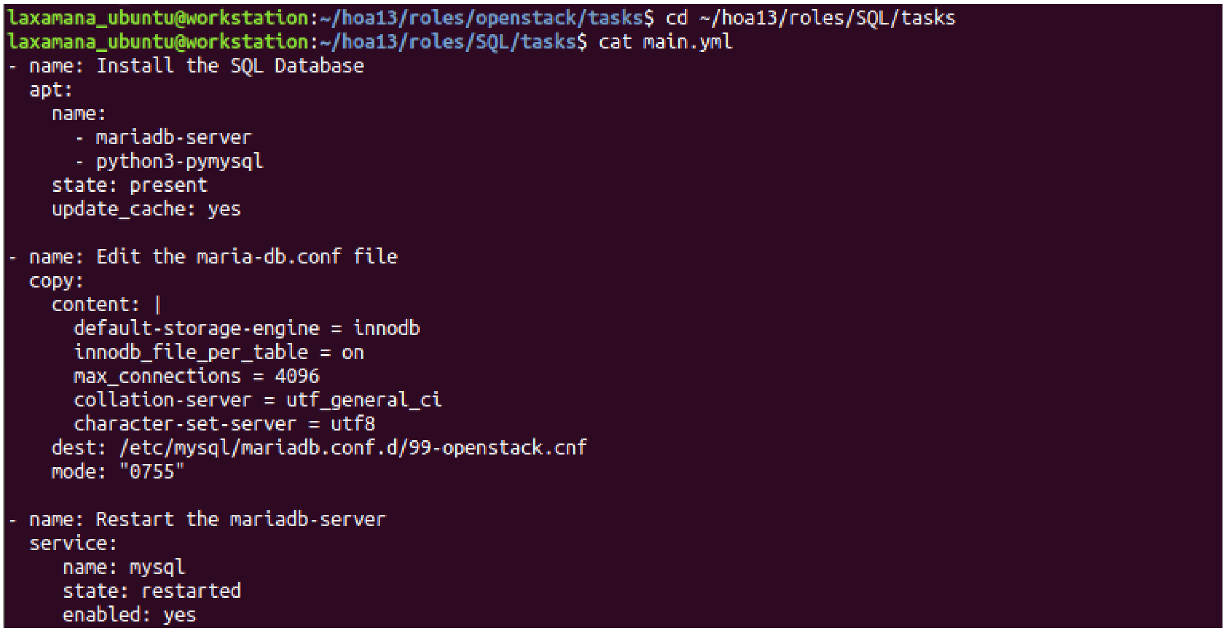
```
laxamana_ubuntu@workstation:~/hoa13/roles/MessageQ/tasks$ cd ~/hoa13/roles/NTP/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/NTP/tasks$ cat main.yml
  - name: Installing the Network Time Protocol (NTP)
    apt:
      name: chrony
      state: present
      update_cache: yes

  - name: Enable the chrony
    service:
      name: chrony.service
      state: restarted
      enabled: yes
```

openstack main.yml



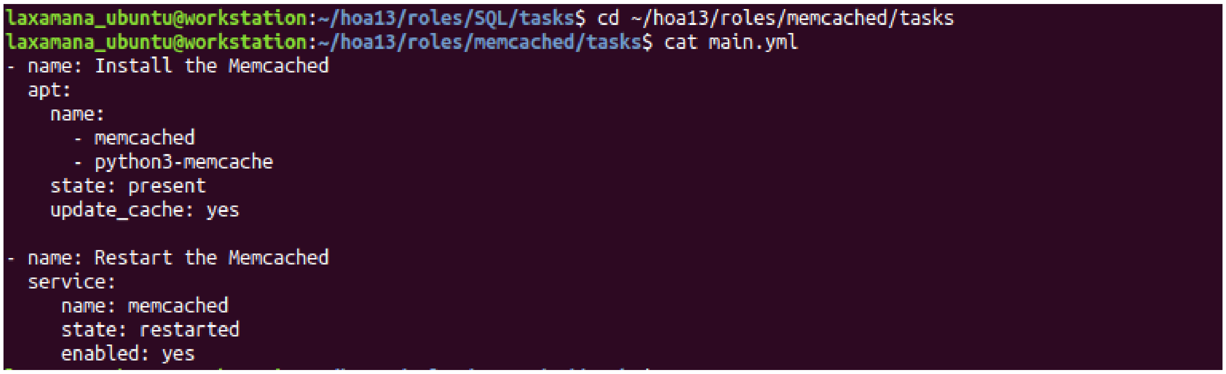
```
laxamana_ubuntu@workstation:~/hoa13/roles/NTP/tasks$ cd ~/hoa13/roles/openstack/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/openstack/tasks$ cat main.yml
- name: Install the OpenStack Packages
  apt:
    name:
      - nova-compute
      - python3-openstackclient
    state: present
    update_cache: yes
```

SQL main.yml



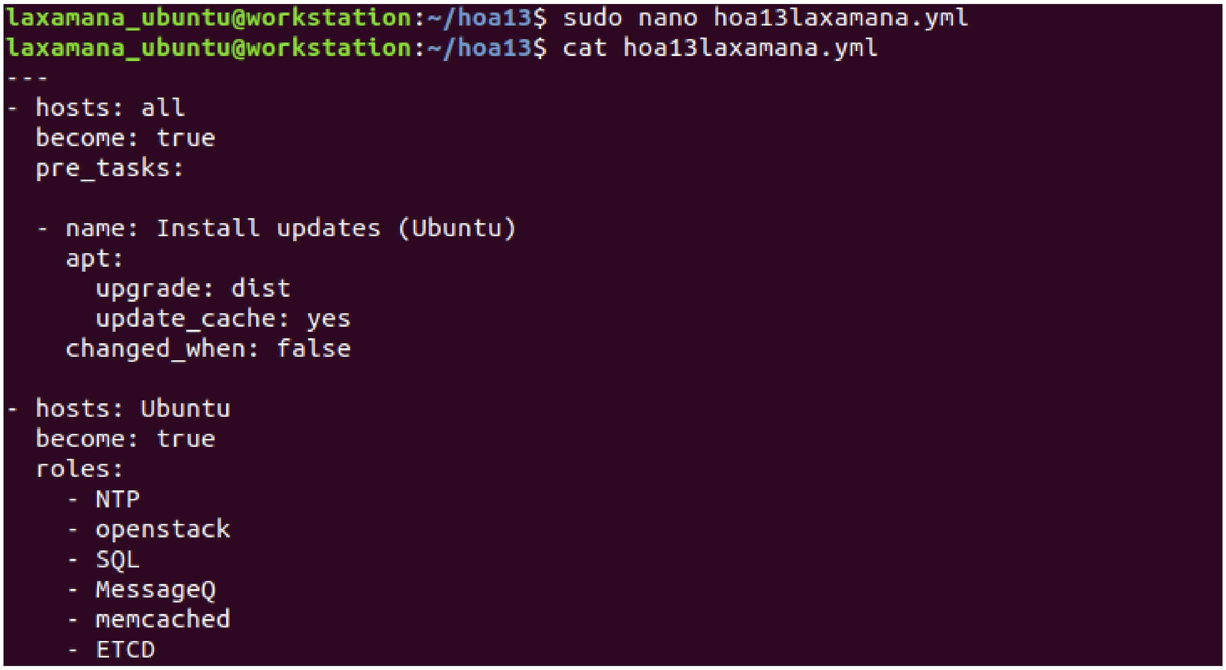
```
laxamana_ubuntu@workstation:~/hoa13/roles/openstack/tasks$ cd ~/hoa13/roles/SQL/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/SQL/tasks$ cat main.yml
- name: Install the SQL Database
  apt:
    name:
      - mariadb-server
      - python3-pymysql
    state: present
    update_cache: yes

- name: Edit the maria-db.conf file
  copy:
    content: |
      default-storage-engine = innodb
      innodb_file_per_table = on
      max_connections = 4096
      collation-server = utf_general_ci
      character-set-server = utf8
    dest: /etc/mysql/mariadb.conf.d/99-openstack.cnf
    mode: "0755"

- name: Restart the mariadb-server
  service:
     name: mysql
     state: restarted
     enabled: yes
```

memcached main.yml



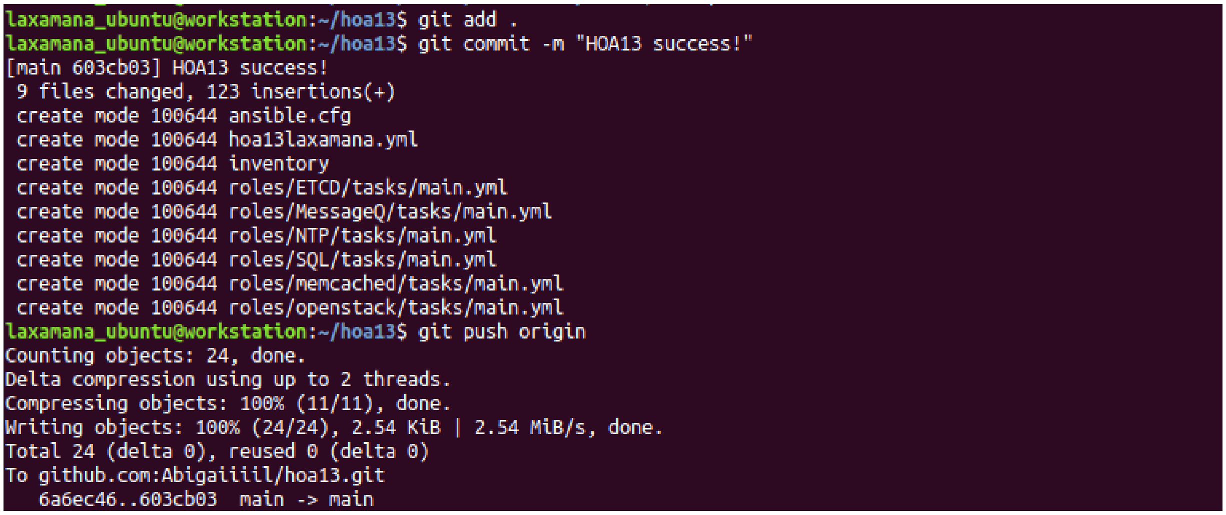
```
laxamana_ubuntu@workstation:~/hoa13/roles/SQL/tasks$ cd ~/hoa13/roles/memcached/tasks
laxamana_ubuntu@workstation:~/hoa13/roles/memcached/tasks$ cat main.yml
- name: Install the Memcached
  apt:
    name:
      - memcached
      - python3-memcache
    state: present
    update_cache: yes

- name: Restart the Memcached
  service:
     name: memcached
     state: restarted
     enabled: yes
```

Step 5. outside roles, create the main yml file.



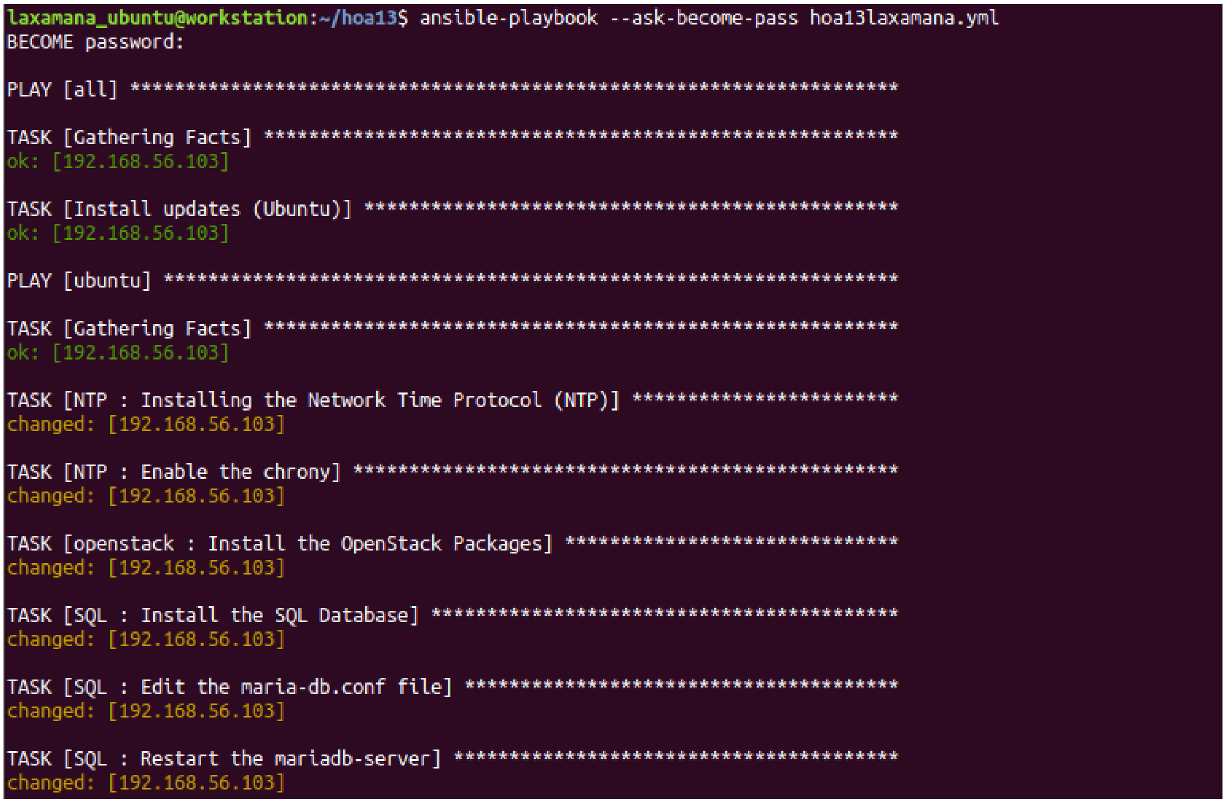
```
laxamana_ubuntu@workstation:~/hoa13$ sudo nano hoa13laxamana.yml
laxamana_ubuntu@workstation:~/hoa13$ cat hoa13laxamana.yml
---
- hosts: all
  become: true
  pre_tasks:

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

- hosts: Ubuntu
  become: true
  roles:
    - NTP
    - openstack
    - SQL
    - MessageQ
    - memcached
    - ETCD
```

Step 6. Commit the changes to your github repository usign the commands **git add .**, **git commit -m** “<message>”, and **git push origin.**

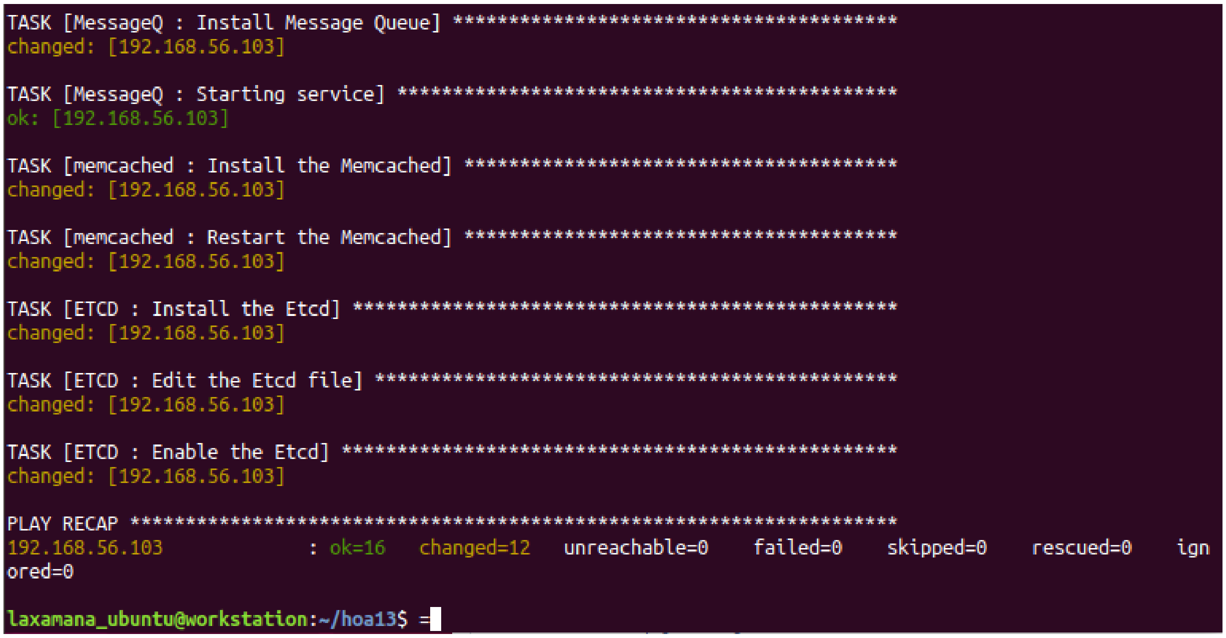


```
laxamana_ubuntu@workstation:~/hoa13$ git add .
laxamana_ubuntu@workstation:~/hoa13$ git commit -m "HOA13 success!"
[main 603cb03] HOA13 success!
 9 files changed, 123 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 hoa13laxamana.yml
 create mode 100644 inventory
 create mode 100644 roles/ETCD/tasks/main.yml
 create mode 100644 roles/MessageQ/tasks/main.yml
 create mode 100644 roles/NTP/tasks/main.yml
 create mode 100644 roles/SQL/tasks/main.yml
 create mode 100644 roles/memcached/tasks/main.yml
 create mode 100644 roles/openstack/tasks/main.yml
laxamana_ubuntu@workstation:~/hoa13$ git push origin
Counting objects: 24, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (11/11), done.
Writing objects: 100% (24/24), 2.54 KiB | 2.54 MiB/s, done.
Total 24 (delta 0), reused 0 (delta 0)
To github.com:Abigaiiiil/hoa13.git
   6a6ec46..603cb03  main -> main
```

Playbook process

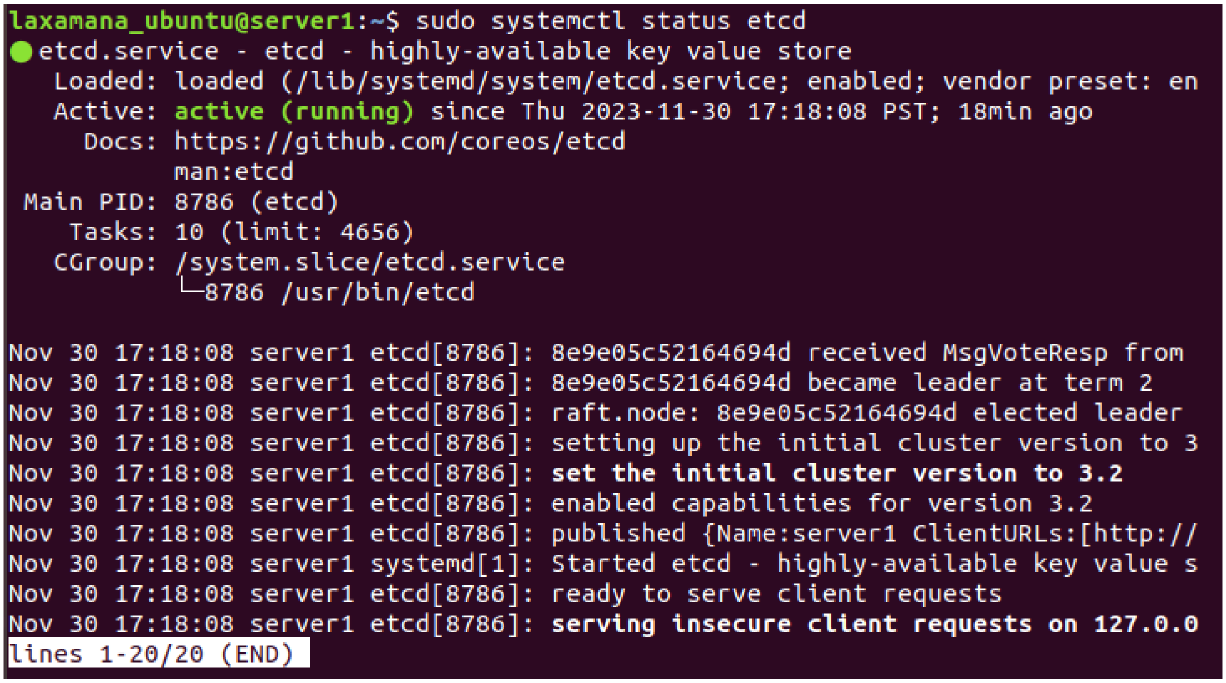


```
laxamana_ubuntu@workstation:~/hoa13$ ansible-playbook --ask-become-pass hoa13laxamana.yml
BECOME password:

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

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

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

PLAY [ubuntu] ******************************************************************

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

TASK [NTP : Installing the Network Time Protocol (NTP)] ************************
changed: [192.168.56.103]

TASK [NTP : Enable the chrony] *************************************************
changed: [192.168.56.103]

TASK [openstack : Install the OpenStack Packages] ******************************
changed: [192.168.56.103]

TASK [SQL : Install the SQL Database] ******************************************
changed: [192.168.56.103]

TASK [SQL : Edit the maria-db.conf file] ***************************************
changed: [192.168.56.103]

TASK [SQL : Restart the mariadb-server] ****************************************
changed: [192.168.56.103]
```

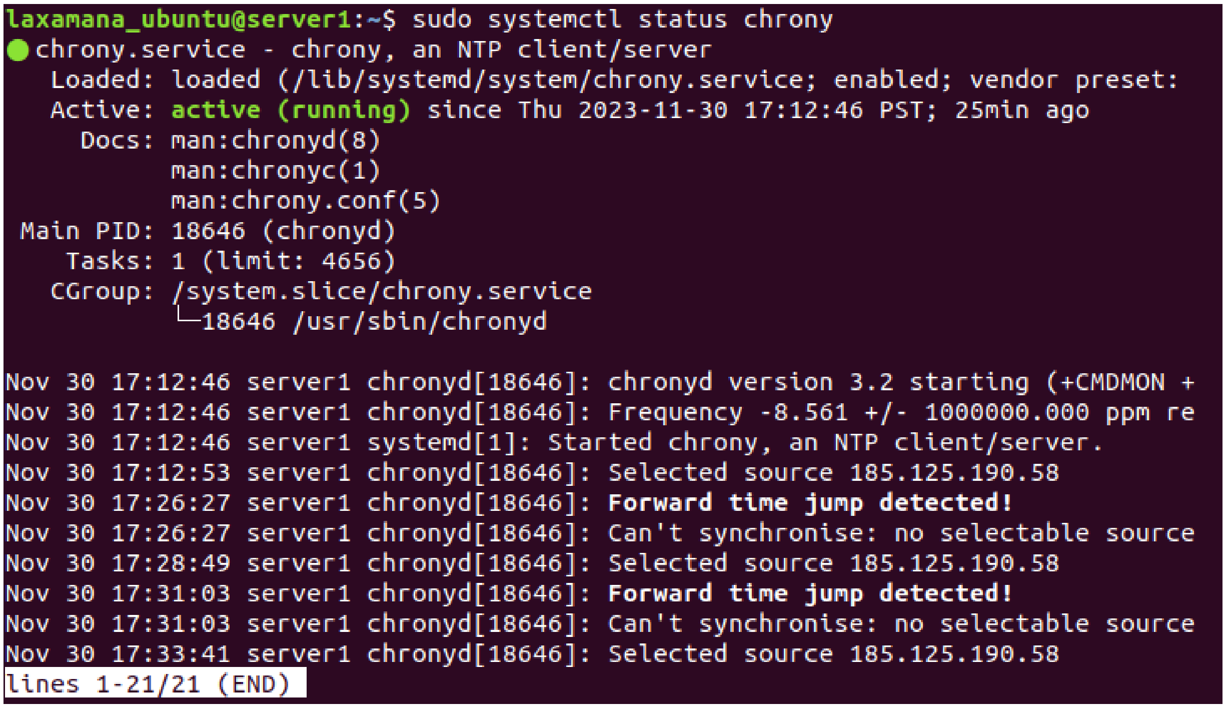


```
TASK [MessageQ : Install Message Queue] ****************************************
changed: [192.168.56.103]

TASK [MessageQ : Starting service] *********************************************
ok: [192.168.56.103]

TASK [memcached : Install the Memcached] ***************************************
changed: [192.168.56.103]

TASK [memcached : Restart the Memcached] ***************************************
changed: [192.168.56.103]

TASK [ETCD : Install the Etcd] *************************************************
changed: [192.168.56.103]

TASK [ETCD : Edit the Etcd file] ***********************************************
changed: [192.168.56.103]

TASK [ETCD : Enable the Etcd] **************************************************
changed: [192.168.56.103]

PLAY RECAP *********************************************************************
192.168.56.103             : ok=16   changed=12   unreachable=0    failed=0    skipped=0    rescued=0    ign
ored=0

laxamana_ubuntu@workstation:~/hoa13$ =
```

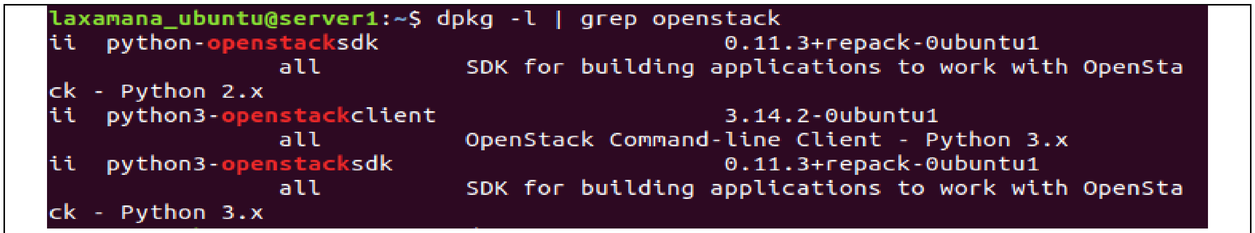
Proof



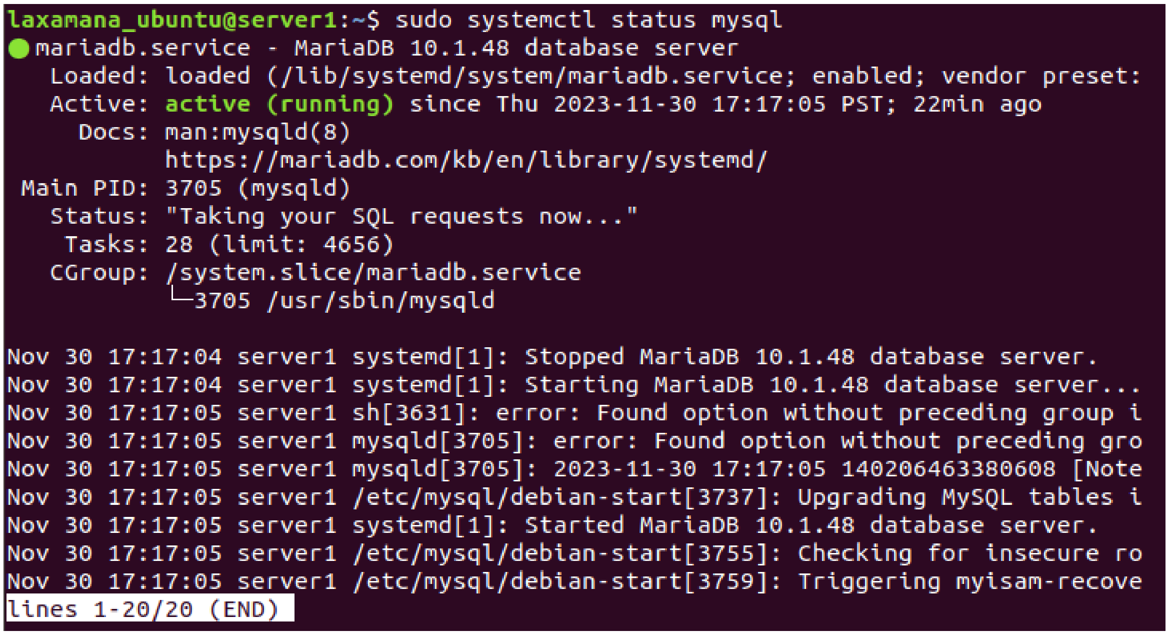
```
laxamana_ubuntu@server1:~$ sudo systemctl status etcd
● etcd.service - etcd - highly-available key value store
   Loaded: loaded (/lib/systemd/system/etcd.service; enabled; vendor preset: en
   Active: active (running) since Thu 2023-11-30 17:18:08 PST; 18min ago
     Docs: https://github.com/coreos/etcd
           man:etcd
 Main PID: 8786 (etcd)
    Tasks: 10 (limit: 4656)
   CGroup: /system.slice/etcd.service
           └─8786 /usr/bin/etcd

Nov 30 17:18:08 server1 etcd[8786]: 8e9e05c52164694d received MsgVoteResp from
Nov 30 17:18:08 server1 etcd[8786]: 8e9e05c52164694d became leader at term 2
Nov 30 17:18:08 server1 etcd[8786]: raft.node: 8e9e05c52164694d elected leader
Nov 30 17:18:08 server1 etcd[8786]: setting up the initial cluster version to 3
Nov 30 17:18:08 server1 etcd[8786]: set the initial cluster version to 3.2
Nov 30 17:18:08 server1 etcd[8786]: enabled capabilities for version 3.2
Nov 30 17:18:08 server1 etcd[8786]: published {Name:server1 ClientURLs:[http://
Nov 30 17:18:08 server1 systemd[1]: Started etcd - highly-available key value s
Nov 30 17:18:08 server1 etcd[8786]: ready to serve client requests
Nov 30 17:18:08 server1 etcd[8786]: serving insecure client requests on 127.0.0
lines 1-20/20 (END)
```



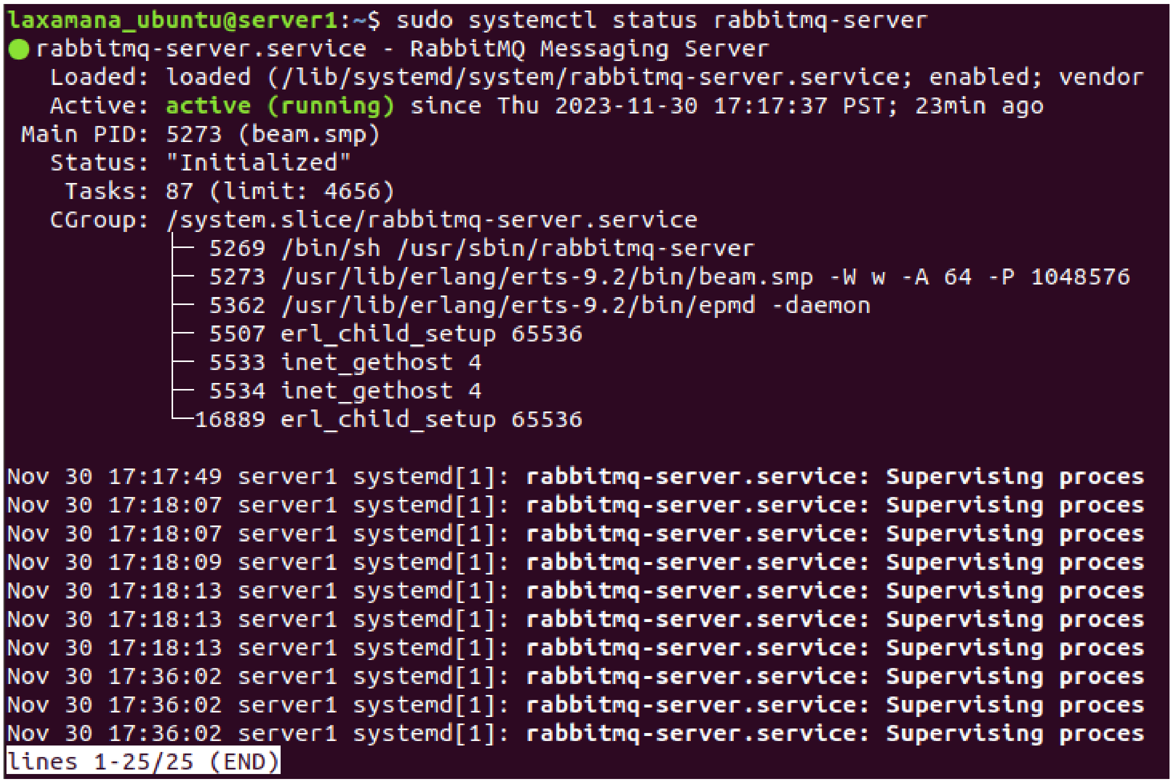
```
laxamana_ubuntu@server1:~$ sudo systemctl status chrony
● chrony.service - chrony, an NTP client/server
   Loaded: loaded (/lib/systemd/system/chrony.service; enabled; vendor preset:
   Active: active (running) since Thu 2023-11-30 17:12:46 PST; 25min ago
     Docs: man:chronyd(8)
           man:chronyc(1)
           man:chrony.conf(5)
 Main PID: 18646 (chronyd)
    Tasks: 1 (limit: 4656)
   CGroup: /system.slice/chrony.service
           └─18646 /usr/sbin/chronyd

Nov 30 17:12:46 server1 chronyd[18646]: chronyd version 3.2 starting (+CMDMON +
Nov 30 17:12:46 server1 chronyd[18646]: Frequency -8.561 +/- 1000000.000 ppm re
Nov 30 17:12:46 server1 systemd[1]: Started chrony, an NTP client/server.
Nov 30 17:12:53 server1 chronyd[18646]: Selected source 185.125.190.58
Nov 30 17:26:27 server1 chronyd[18646]: Forward time jump detected!
Nov 30 17:26:27 server1 chronyd[18646]: Can't synchronise: no selectable source
Nov 30 17:28:49 server1 chronyd[18646]: Selected source 185.125.190.58
Nov 30 17:31:03 server1 chronyd[18646]: Forward time jump detected!
Nov 30 17:31:03 server1 chronyd[18646]: Can't synchronise: no selectable source
Nov 30 17:33:41 server1 chronyd[18646]: Selected source 185.125.190.58
lines 1-21/21 (END)
```



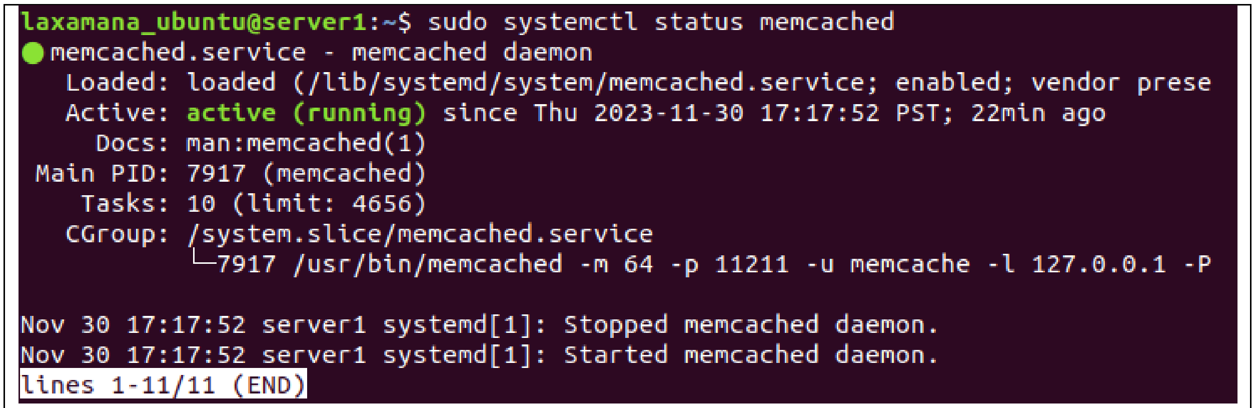
```
laxamana_ubuntu@server1:~$ dpkg -l | grep openstack
ii  python-openstacksdk                    0.11.3+repack-0ubuntu1
            all          SDK for building applications to work with OpenSta
ck - Python 2.x
ii  python3-openstackclient                3.14.2-0ubuntu1
            all          OpenStack Command-line Client - Python 3.x
ii  python3-openstacksdk                   0.11.3+repack-0ubuntu1
            all          SDK for building applications to work with OpenSta
ck - Python 3.x
```



```
laxamana_ubuntu@server1:~$ sudo systemctl status mysql
● mariadb.service - MariaDB 10.1.48 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset:
   Active: active (running) since Thu 2023-11-30 17:17:05 PST; 22min ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
 Main PID: 3705 (mysqld)
   Status: "Taking your SQL requests now..."
    Tasks: 28 (limit: 4656)
   CGroup: /system.slice/mariadb.service
           └─3705 /usr/sbin/mysqld

Nov 30 17:17:04 server1 systemd[1]: Stopped MariaDB 10.1.48 database server.
Nov 30 17:17:04 server1 systemd[1]: Starting MariaDB 10.1.48 database server...
Nov 30 17:17:05 server1 sh[3631]: error: Found option without preceding group i
Nov 30 17:17:05 server1 mysqld[3705]: error: Found option without preceding gro
Nov 30 17:17:05 server1 mysqld[3705]: 2023-11-30 17:17:05 140206463380608 [Note
Nov 30 17:17:05 server1 /etc/mysql/debian-start[3737]: Upgrading MySQL tables i
Nov 30 17:17:05 server1 systemd[1]: Started MariaDB 10.1.48 database server.
Nov 30 17:17:05 server1 /etc/mysql/debian-start[3755]: Checking for insecure ro
Nov 30 17:17:05 server1 /etc/mysql/debian-start[3759]: Triggering myisam-recove
lines 1-20/20 (END)
```



```
laxamana_ubuntu@server1:~$ sudo systemctl status rabbitmq-server
● rabbitmq-server.service - RabbitMQ Messaging Server
   Loaded: loaded (/lib/systemd/system/rabbitmq-server.service; enabled; vendor
   Active: active (running) since Thu 2023-11-30 17:17:37 PST; 23min ago
 Main PID: 5273 (beam.smp)
   Status: "Initialized"
    Tasks: 87 (limit: 4656)
   CGroup: /system.slice/rabbitmq-server.service
           ├─ 5269 /bin/sh /usr/sbin/rabbitmq-server
           ├─ 5273 /usr/lib/erlang/erts-9.2/bin/beam.smp -W w -A 64 -P 1048576
           ├─ 5362 /usr/lib/erlang/erts-9.2/bin/epmd -daemon
           ├─ 5507 erl_child_setup 65536
           ├─ 5533 inet_gethost 4
           ├─ 5534 inet_gethost 4
           └─16889 erl_child_setup 65536

Nov 30 17:17:49 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:07 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:07 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:09 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:13 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:13 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:18:13 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:36:02 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:36:02 server1 systemd[1]: rabbitmq-server.service: Supervising proces
Nov 30 17:36:02 server1 systemd[1]: rabbitmq-server.service: Supervising proces
lines 1-25/25 (END)
```



```
laxamana_ubuntu@server1:~$ sudo systemctl status memcached
● memcached.service - memcached daemon
   Loaded: loaded (/lib/systemd/system/memcached.service; enabled; vendor prese
   Active: active (running) since Thu 2023-11-30 17:17:52 PST; 22min ago
     Docs: man:memcached(1)
 Main PID: 7917 (memcached)
    Tasks: 10 (limit: 4656)
   CGroup: /system.slice/memcached.service
           └─7917 /usr/bin/memcached -m 64 -p 11211 -u memcache -l 127.0.0.1 -P

Nov 30 17:17:52 server1 systemd[1]: Stopped memcached daemon.
Nov 30 17:17:52 server1 systemd[1]: Started memcached daemon.
lines 1-11/11 (END)
```

Github link

https://github.com/Abigaiiiil/hoa13.git

Reflections:

Answer the following:

1. What are the benefits of implementing OpenStack?

When establishing private or public cloud infrastructure, enterprises may profit from a number of advantages when utilizing OpenStack, an open-source cloud computing platform. Organizations may avoid the high expenses that come with exclusive cloud solutions by using the open-source nature of OpenStack. The platform makes it possible to use standard hardware, which reduces infrastructure costs and makes setting up and maintaining cloud environments more economical. OpenStack makes it possible to automate administration and installation procedures, which simplifies operations and reduces the need for human interaction. It has orchestration features that enable the creation of complex multi-tier applications and workflows through the use of tools like Heat. OpenStack's scalability allows companies to modify their resource allocations based on demand. Because of its adaptability while managing storage areas and virtual machines. With networking capabilities, it can handle a variety of workloads and adapt to the needs of the company without being vendor-locked in. Additionally, it offers security features and access controls to protect data and ensure compliance with regulatory requirements. OpenStack benefits from a vibrant developer and contributor community that continuously enhances the platform's capabilities since it is an open-source platform. This promotes innovation and ensures a steady stream of improvements, additions, and upgrades.

Conclusions:

Upon the completion of this activity, I have observed the important turning point accomplished in creating a simplified and effective basis for a reliable cloud architecture. The installation process may be made more consistent, repeatable, and less vulnerable to human mistake by utilizing Ansible's automation features. This completion shows how well-designed playbooks work for coordinating the deployment of necessary prerequisites, such as dependencies, system updates, package installs, and configuration settings for the OpenStack environment. It also emphasizes how automation helps to simplify difficult operations, save deployment times, and guarantee consistent settings across many environments or nodes. The remaining stages of establishing and operating an OpenStack cloud are made possible by accomplishing all the requirements for OpenStack with Ansible in Ubuntu. This provides a solid foundation for scalability, adaptability, and smooth operations inside the cloud architecture.