Name: Daniela Marie D. Rabang	**Date Performed:** 10/02/2023
Course/Section: CPE232/CPE31S4	**Date Submitted:** 10/02/2023
Instructor: Dr. Jonathan V. Taylar	**Semester and SY:** 1st Sem 2023-2024

Activity 6: Targeting Specific Nodes and Managing Services

1. Objectives:

1.1 Individualize hosts
1.2 Apply tags in selecting plays to run
1.3 Managing Services from remote servers using playbooks

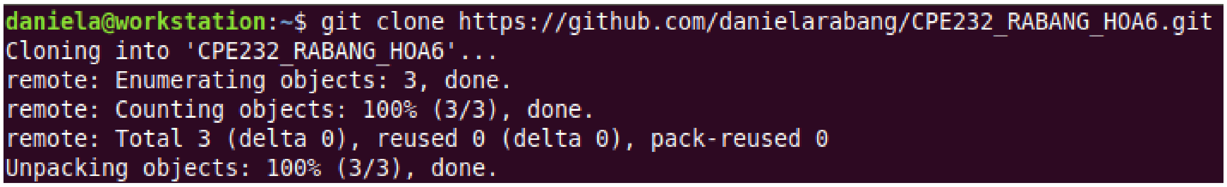
2. Discussion:

In this activity, we try to individualize hosts. For example, we don't want apache on all our servers, or maybe only one of our servers is a web server, or maybe we have different servers like database or file servers running different things on different categories of servers and that is what we are going to take a look at in this activity.

We also try to manage services that do not automatically run using the automations in playbook. For example, when we install web servers or httpd for CentOS, we notice that the service did not start automatically.

Requirement:

In this activity, you will need to create another Ubuntu VM and name it Server 3. Likewise, you need to activate the second adapter to a host-only adapter after the installations. Take note of the IP address of the Server 3. Make sure to use the command *ssh-copy-id* to copy the public key to Server 3. Verify if you can successfully SSH to Server 3.

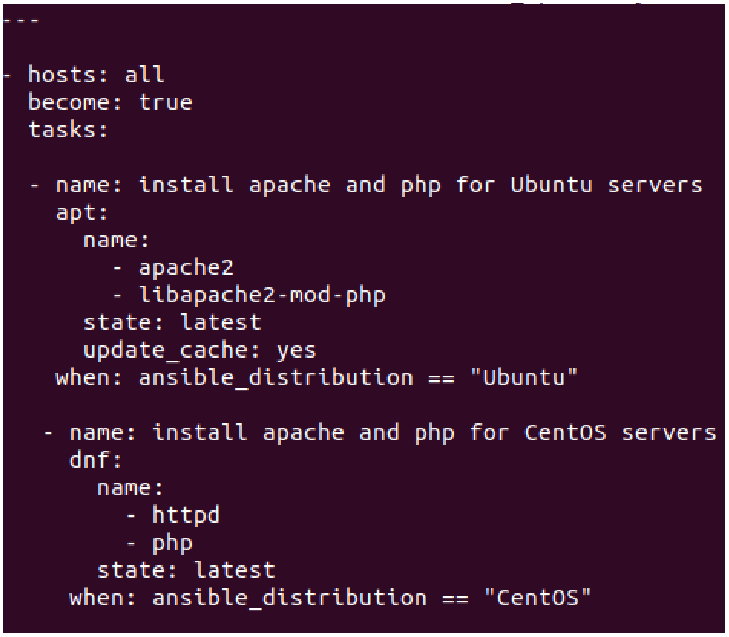


```
daniela@workstation:~$ git clone https://github.com/danielarabang/CPE232_RABANG_HOA6.git
Cloning into 'CPE232_RABANG_HOA6'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
```

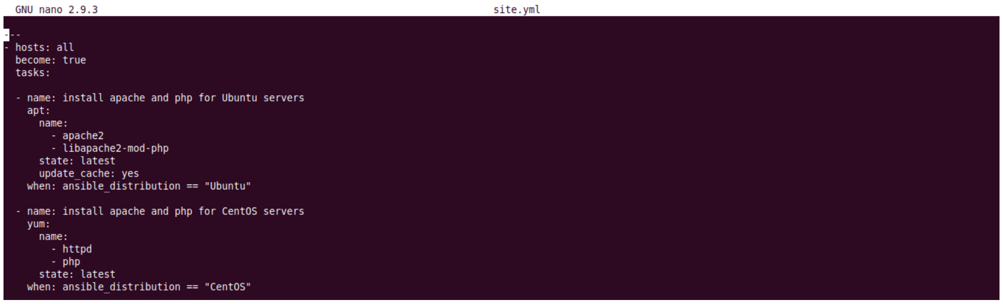
- I created a repository named CPE232_RABANG_HOA6. I cloned it to my workstation.

Task 1: Targeting Specific Nodes

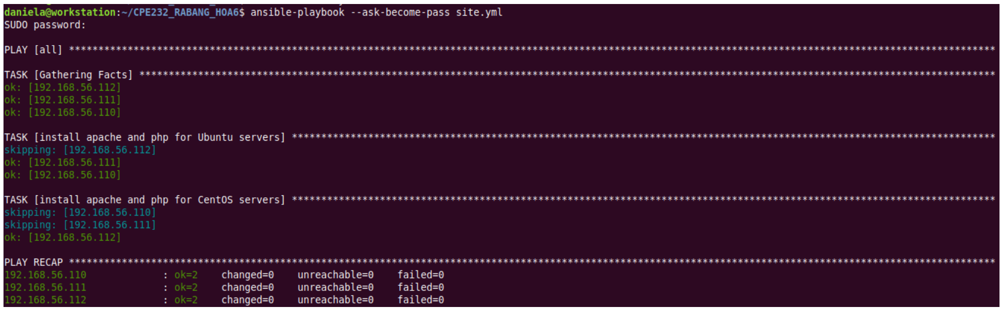
1. Create a new playbook and named it site.yml. Follow the commands as shown in the image below. Make sure to save the file and exit.



```
---

- hosts: all
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
      update_cache: yes
    when: ansible_distribution == "Ubuntu"

   - name: install apache and php for CentOS servers
     dnf:
       name:
         - httpd
         - php
       state: latest
     when: ansible_distribution == "CentOS"
```



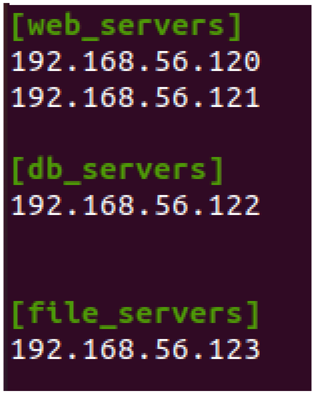
```
  GNU nano 2.9.3                                              site.yml

---
- hosts: all
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
      update_cache: yes
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    yum:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"
```



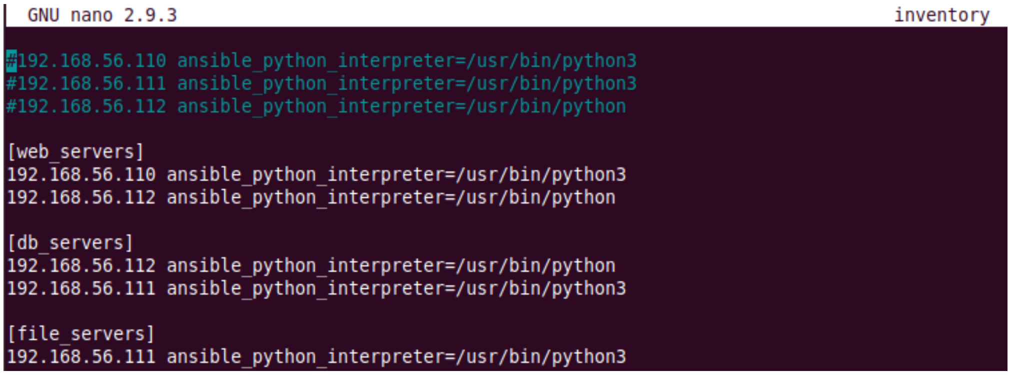
```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ****************************************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

TASK [install apache and php for Ubuntu servers] **************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] **************************************************************************************
skipping: [192.168.56.110]
skipping: [192.168.56.111]
ok: [192.168.56.112]

PLAY RECAP ****************************************************************************************************************************
192.168.56.110             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=2    changed=0    unreachable=0    failed=0
```

In this section I had created a play book which is named site.yml, and I ordered it to install apache and php for both Ubuntu and CentOS servers. Then I had issued a command that ran all the tasks that are in my playbook.

2. Edit the inventory file. Remove the variables we put in our last activity and group according to the image shown below:



```
[web_servers]
192.168.56.120
192.168.56.121

[db_servers]
192.168.56.122


[file_servers]
192.168.56.123
```



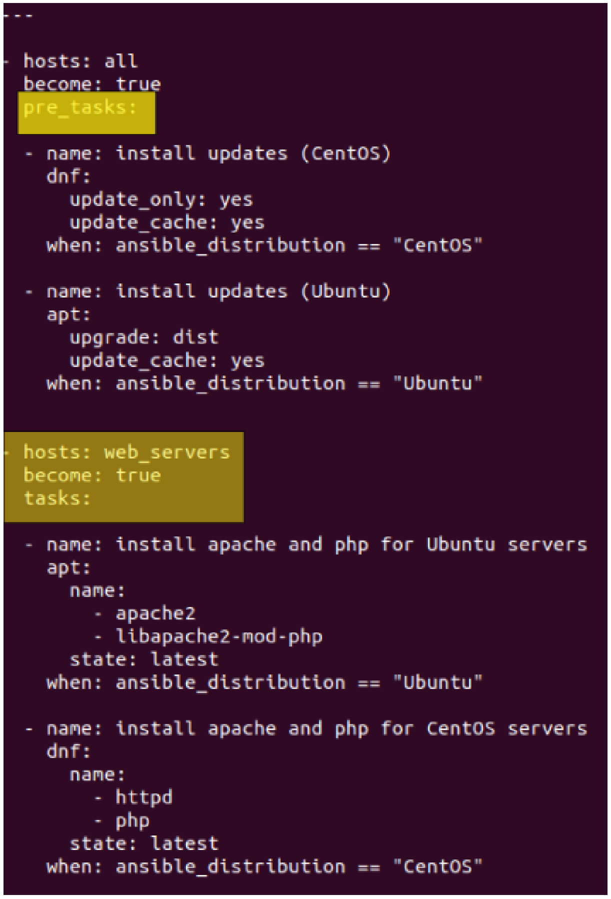
```
  GNU nano 2.9.3                                                    inventory

#192.168.56.110 ansible_python_interpreter=/usr/bin/python3
#192.168.56.111 ansible_python_interpreter=/usr/bin/python3
#192.168.56.112 ansible_python_interpreter=/usr/bin/python

[web_servers]
192.168.56.110 ansible_python_interpreter=/usr/bin/python3
192.168.56.112 ansible_python_interpreter=/usr/bin/python

[db_servers]
192.168.56.112 ansible_python_interpreter=/usr/bin/python
192.168.56.111 ansible_python_interpreter=/usr/bin/python3

[file_servers]
192.168.56.111 ansible_python_interpreter=/usr/bin/python3
```

Make sure to save the file and exit.

In my inventory file where all my virtual machines ip addresses are put in. So I had removed the variables that I had from the past activities and I had commented on them. Then I inserted the variables that determine the three which are the web_servers, db_servers, and the file_servers.

Right now, we have created groups in our inventory file and put each server in its own group. In other cases, you can have a server be a member of multiple groups, for example you have a test server that is also a web server.

3. Edit the *site.yml* by following the image below:

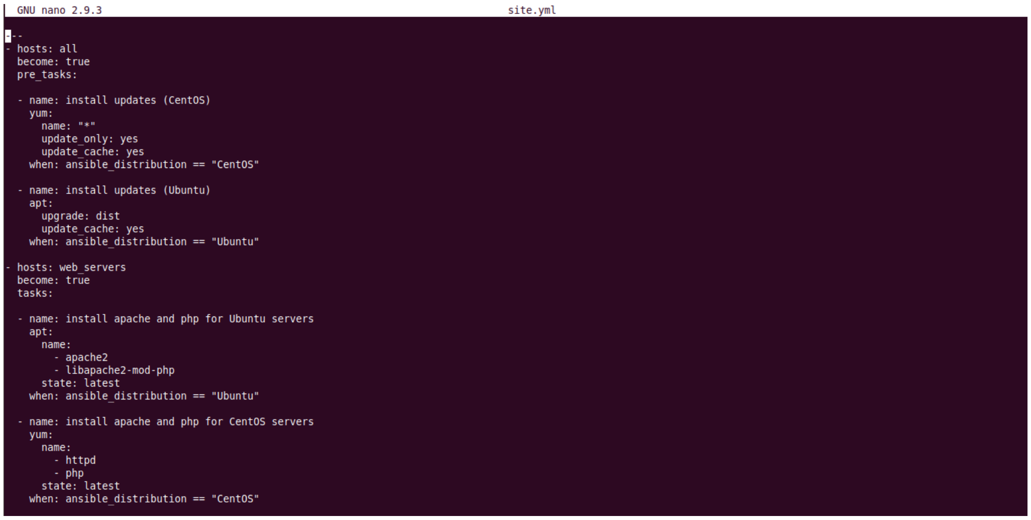


```
---

- hosts: all
  become: true
  pre_tasks:

  - name: install updates (CentOS)
    dnf:
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"


- hosts: web_servers
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    dnf:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"
```

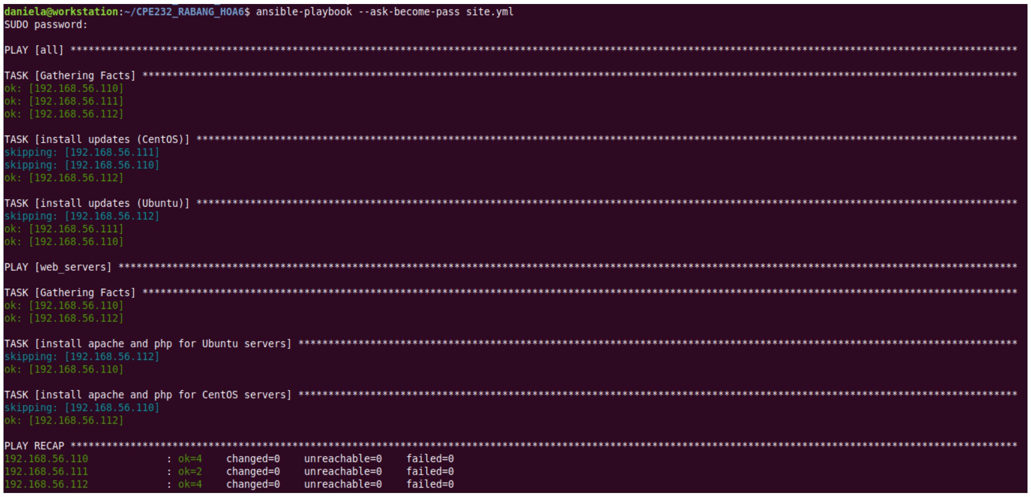
Make sure to save the file and exit.

The *pre-tasks* command tells the ansible to run it before any other thing. In the *pre-tasks*, CentOS will install updates while Ubuntu will upgrade its distribution package. This will run before running the second play, which is targeted at *web_servers*. In the second play, apache and php will be installed on both Ubuntu servers and CentOS servers.

Run the *site.yml* file and describe the result.



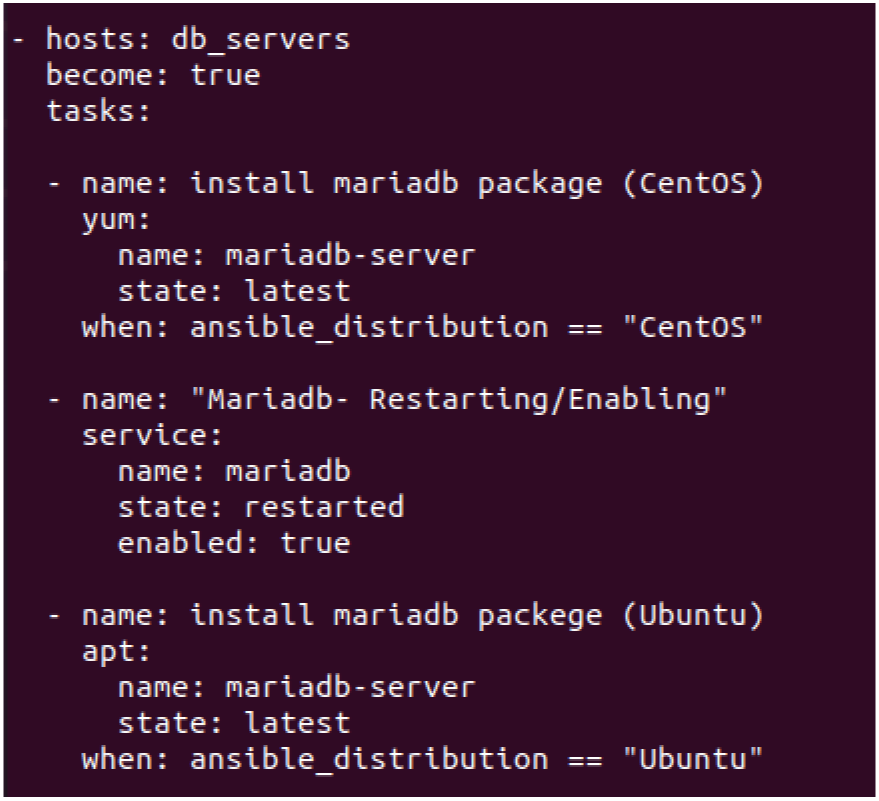
```
  GNU nano 2.9.3                                                    site.yml

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

  - name: install updates (CentOS)
    yum:
      name: "*"
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"

- hosts: web_servers
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    yum:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"
```



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ****************************************************************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install updates (CentOS)] *******************************************************************************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] *******************************************************************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] ********************************************************************************************************************************************************

TASK [Gathering Facts] ****************************************************************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for Ubuntu servers] **************************************************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] **************************************************************************************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY RECAP ****************************************************************************************************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=4    changed=0    unreachable=0    failed=0
```

In this section I had put tasks in the playbook where it installed updates for both servers. Then I had modified the playbook where the hosts must be web_servers to run the tasks that install apache and php.

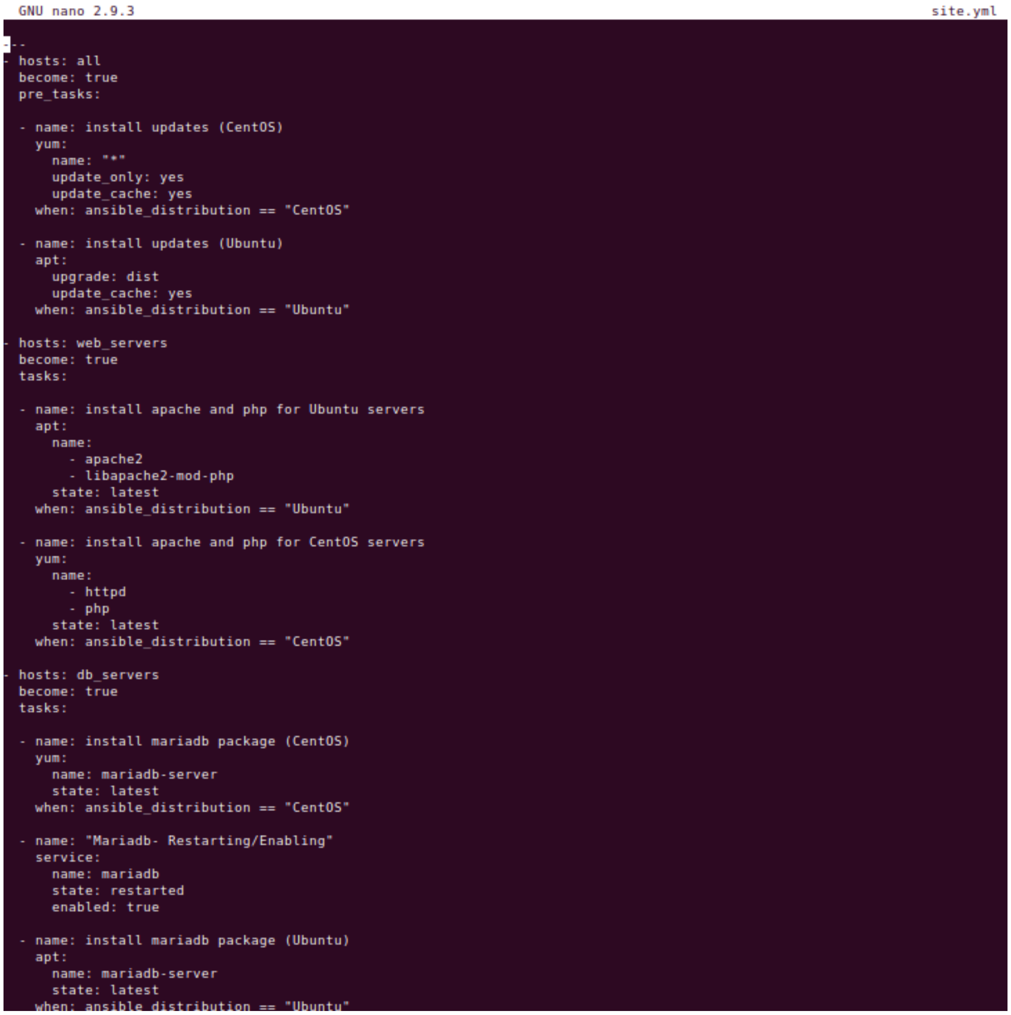
4. Let's try to edit again the *site.yml* file. This time, we are going to add plays targeting the other servers. This time we target the *db_servers* by adding it on the current *site.yml*. Below is an example: (Note add this at the end of the playbooks from task 1.3.



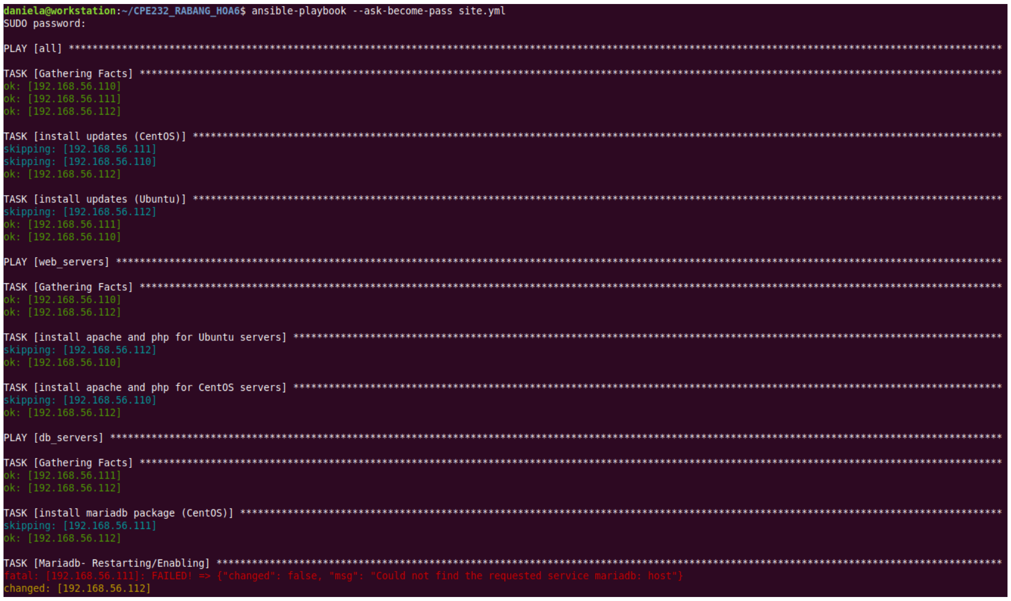
```
- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    yum:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true

  - name: install mariadb packege (Ubuntu)
    apt:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "Ubuntu"
```

Make sure to save the file and exit.

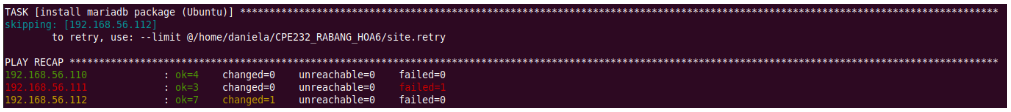
Run the *site.yml* file and describe the result.



```
  GNU nano 2.9.3                                                                              site.yml

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

  - name: install updates (CentOS)
    yum:
      name: "*"
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"

- hosts: web_servers
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    yum:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"

- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    yum:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true

  - name: install mariadb package (Ubuntu)
    apt:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "Ubuntu"
```



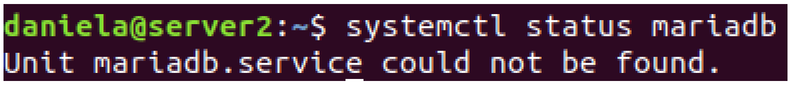
```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ***********************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install updates (CentOS)] **************************************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] **************************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] ***************************************************************************************************************

TASK [Gathering Facts] ***********************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for Ubuntu servers] *********************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] *********************************************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] ****************************************************************************************************************

TASK [Gathering Facts] ***********************************************************************************************************
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (CentOS)] ******************************************************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [Mariadb- Restarting/Enabling] **********************************************************************************************
fatal: [192.168.56.111]: FAILED! => {"changed": false, "msg": "Could not find the requested service mariadb: host"}
changed: [192.168.56.112]
```



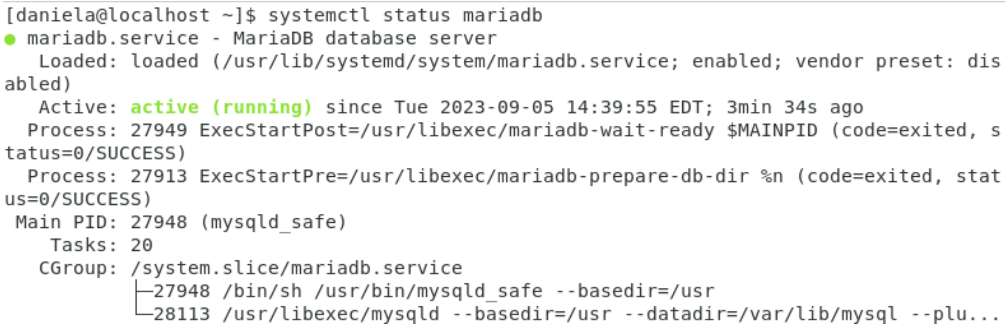
```
TASK [install mariadb package (Ubuntu)] *************************************************************************************************
skipping: [192.168.56.112]
        to retry, use: --limit @/home/daniela/CPE232_RABANG_HOA6/site.retry

PLAY RECAP *******************************************************************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=3    changed=0    unreachable=0    failed=1
192.168.56.112             : ok=7    changed=1    unreachable=0    failed=0
```

The playbook is modified again to run special tasks for all the servers that are included in the db_servers. The tasks install mariadb package for both Ubuntu and CentOS servers.

5. Go to the remote server (Ubuntu) terminal that belongs to the db_servers group and check the status for mariadb installation using the command: *systemctl status mariadb.* Do this on the CentOS server also.



```
daniela@server2:~$ systemctl status mariadb
Unit mariadb.service could not be found.
```

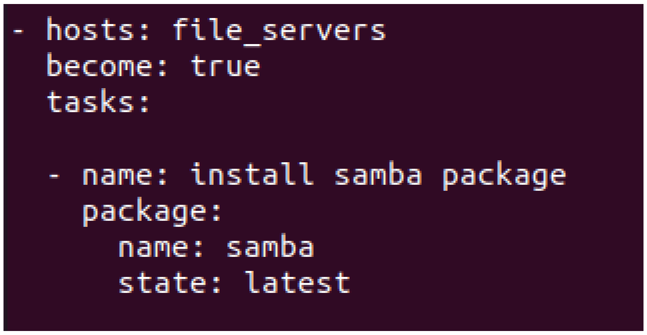


```
[daniela@localhost ~]$ systemctl status mariadb
● mariadb.service - MariaDB database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: dis
abled)
   Active: active (running) since Tue 2023-09-05 14:39:55 EDT; 3min 34s ago
  Process: 27949 ExecStartPost=/usr/libexec/mariadb-wait-ready $MAINPID (code=exited, s
tatus=0/SUCCESS)
  Process: 27913 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir %n (code=exited, stat
us=0/SUCCESS)
 Main PID: 27948 (mysqld_safe)
    Tasks: 20
   CGroup: /system.slice/mariadb.service
           ├─27948 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
           └─28113 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --plu...
```

Describe the output.

For the output I had observed that the mariadb package is not installed in the server2 and in the CentOS it is installed. This is because in server 2 the mariadb request or package is not found.

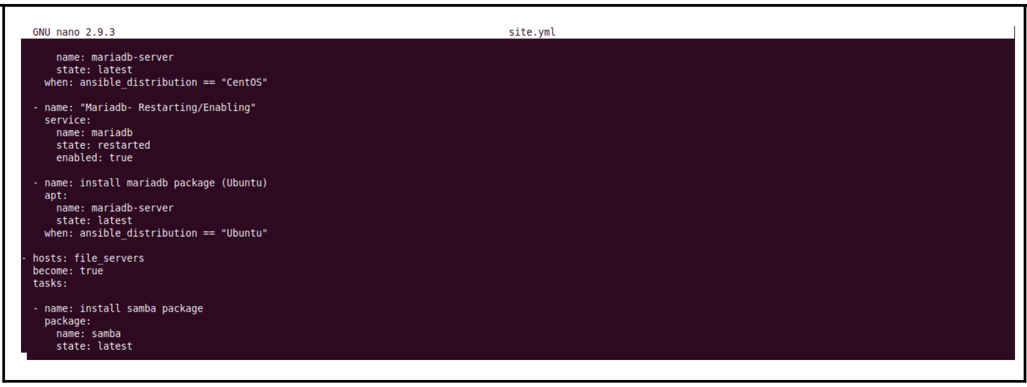
6. Edit the *site.yml* again. This time we will append the code to configure installation on the *file_servers* group. We can add the following on our file.



```
- hosts: file_servers
  become: true
  tasks:

  - name: install samba package
    package:
      name: samba
      state: latest
```

Make sure to save the file and exit.

Run the *site.yml* file and describe the result.



```
GNU nano 2.9.3                                                    site.yml

      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true

  - name: install mariadb package (Ubuntu)
    apt:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "Ubuntu"

- hosts: file_servers
  become: true
  tasks:

  - name: install samba package
    package:
      name: samba
      state: latest
```

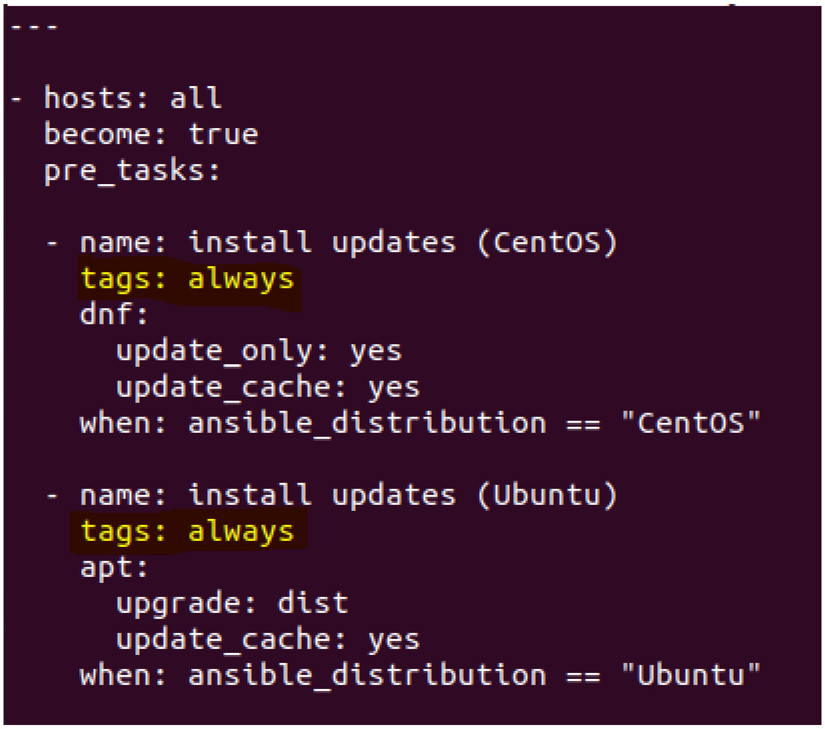
The playbook is modified again and I added the tasks for the file_servers which is installing samba packages onto the servers that are in the category file_server.

The testing of the *file_servers* is beyond the scope of this activity, and as well as our topics and objectives. However, in this activity we were able to show that we can target hosts or servers using grouping in ansible playbooks.

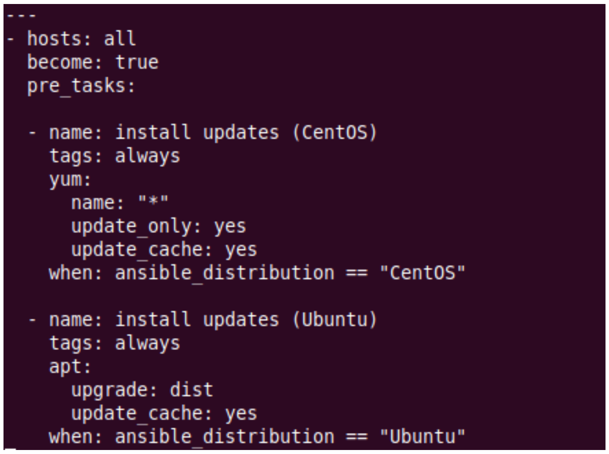
Task 2: Using Tags in running playbooks

In this task, our goal is to add metadata to our plays so that we can only run the plays that we want to run, and not all the plays in our playbook.

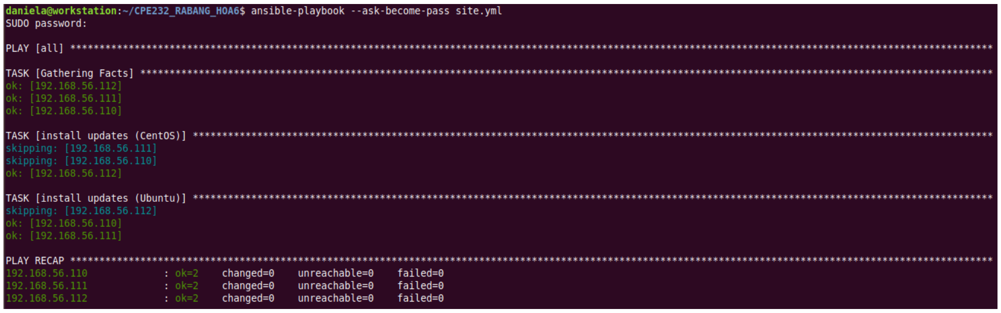
1. Edit the *site.yml* file. Add tags to the playbook. After the name, we can place the tags: *name_of_tag*. This is an arbitrary command, which means you can use any name for a tag.



```
---

- hosts: all
  become: true
  pre_tasks:

  - name: install updates (CentOS)
    tags: always
    dnf:
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    tags: always
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"
```

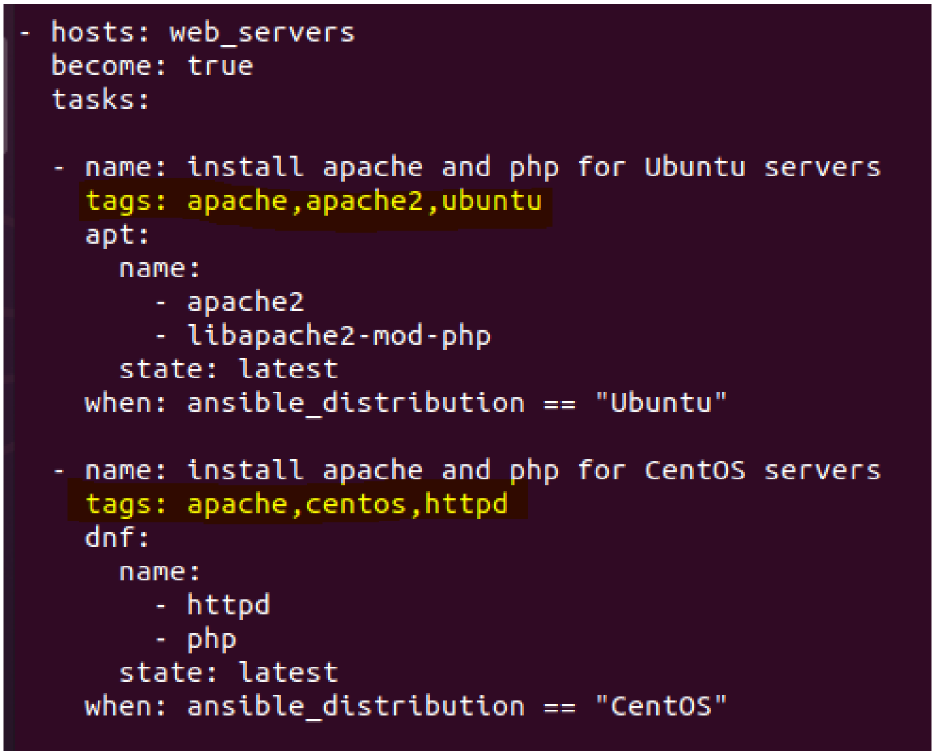


```
---
- hosts: all
  become: true
  pre_tasks:

  - name: install updates (CentOS)
    tags: always
    yum:
      name: "*"
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    tags: always
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"
```

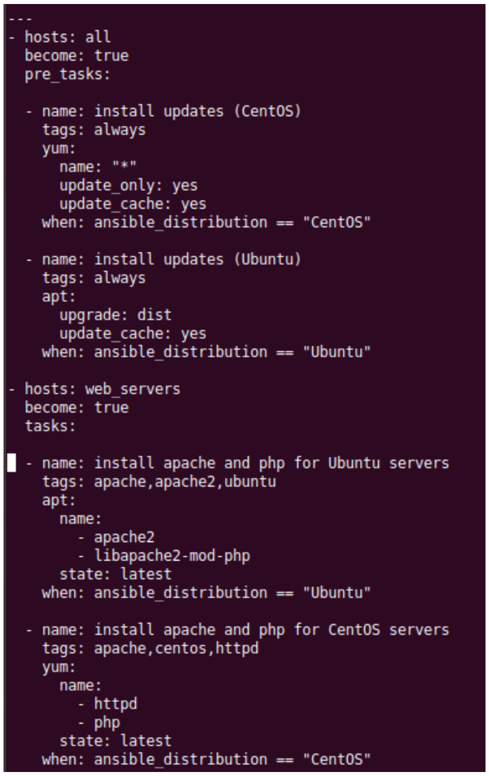


```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ****************************************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

TASK [install updates (CentOS)] *******************************************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] *******************************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

PLAY RECAP ****************************************************************************************************************************
192.168.56.110             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=2    changed=0    unreachable=0    failed=0
```

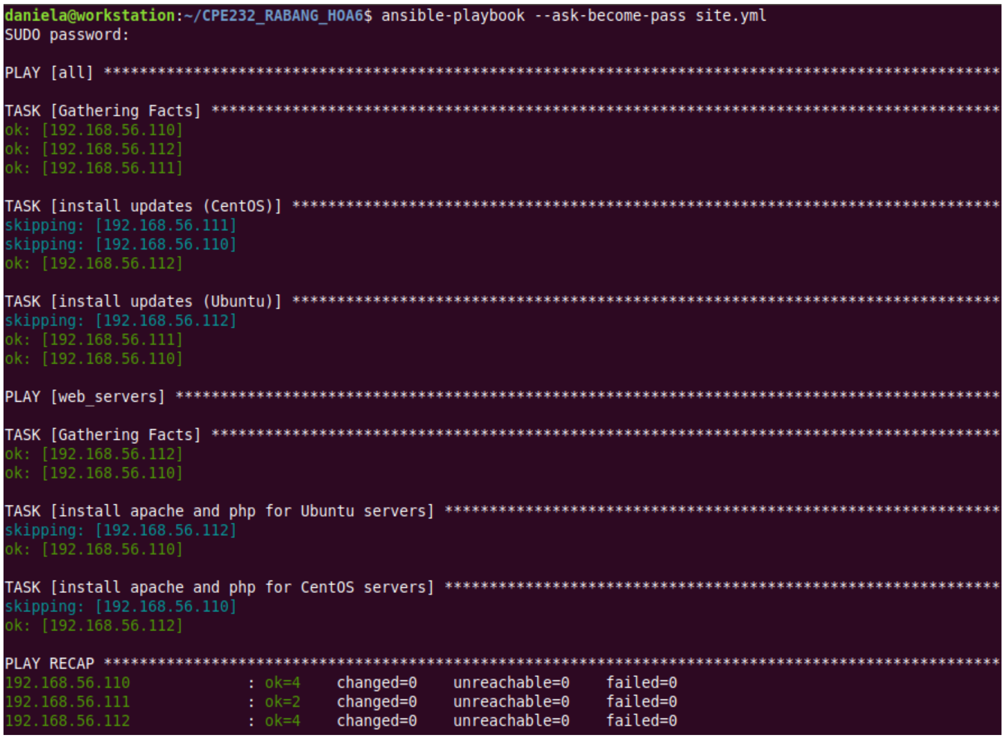
For this part, I had put codes on to the playbook where it installed updates but with the detection of tags.



```
- hosts: web_servers
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    tags: apache,apache2,ubuntu
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    tags: apache,centos,httpd
    dnf:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"
```

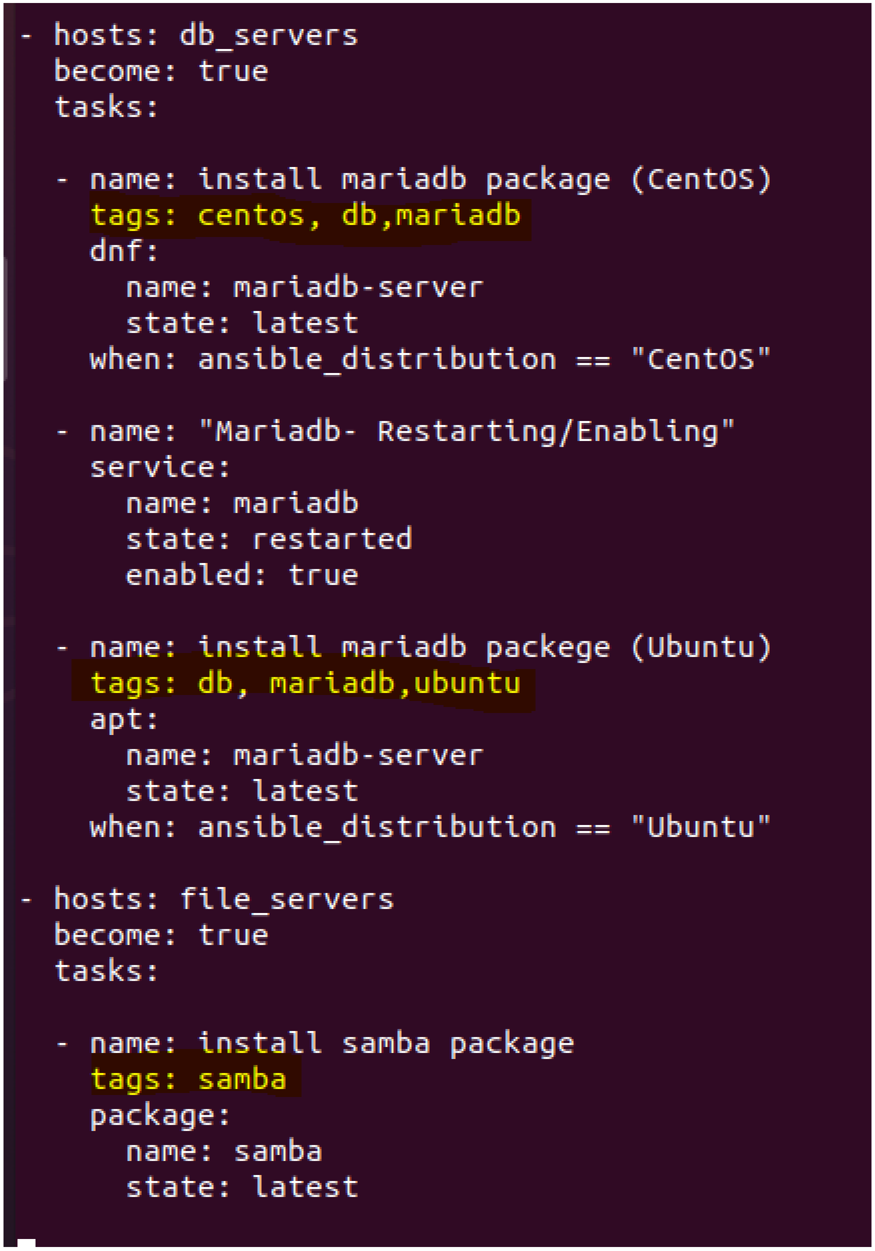


```
---
- hosts: all
  become: true
  pre_tasks:

  - name: install updates (CentOS)
    tags: always
    yum:
      name: "*"
      update_only: yes
      update_cache: yes
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    tags: always
    apt:
      upgrade: dist
      update_cache: yes
    when: ansible_distribution == "Ubuntu"

- hosts: web_servers
  become: true
  tasks:

  - name: install apache and php for Ubuntu servers
    tags: apache,apache2,ubuntu
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: install apache and php for CentOS servers
    tags: apache,centos,httpd
    yum:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"
```

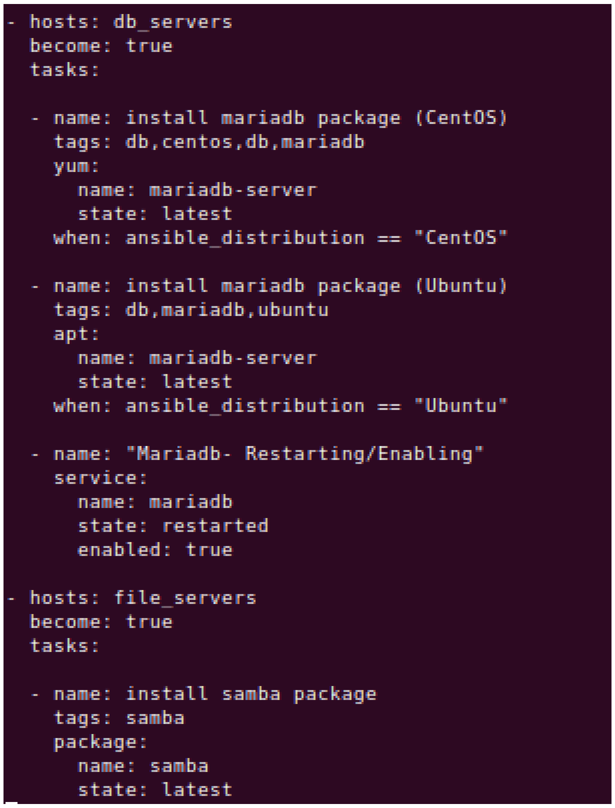


```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]
ok: [192.168.56.111]

TASK [install updates (CentOS)] ***************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] ***************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] ****************************************************************************************

TASK [Gathering Facts] ************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for Ubuntu servers] **********************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] **********************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY RECAP ************************************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=2    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=4    changed=0    unreachable=0    failed=0
```

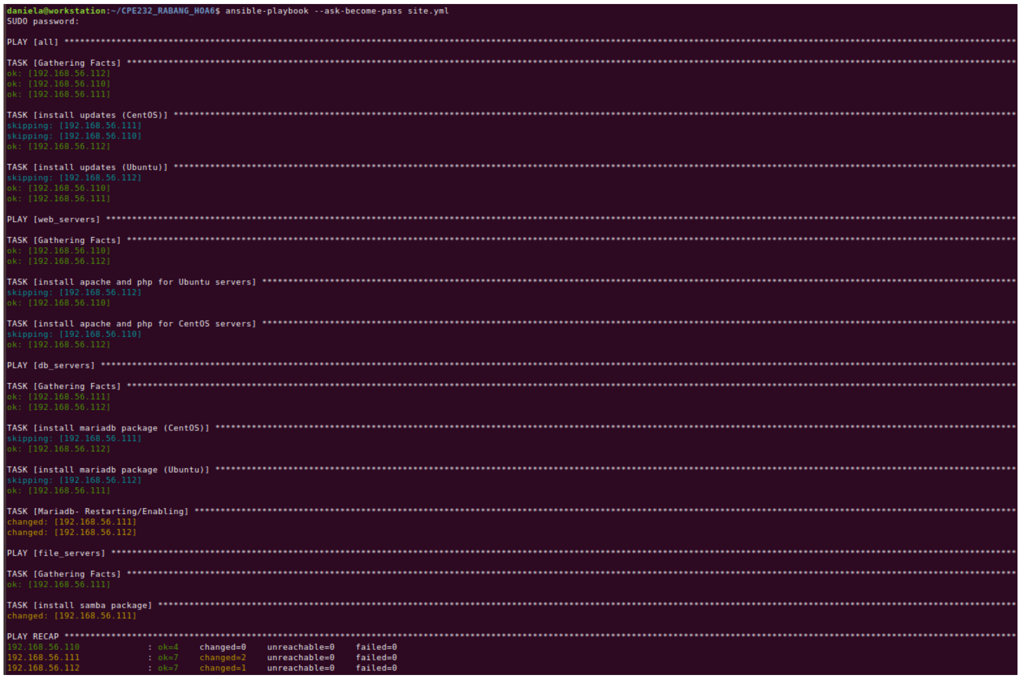
For this part, I had put codes on to the playbook where it installed updates but with the detection of tags. For the tags I put the appropriate tags for the right server.



```
- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    tags: centos, db,mariadb
    dnf:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true

  - name: install mariadb packege (Ubuntu)
    tags: db, mariadb,ubuntu
    apt:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "Ubuntu"

- hosts: file_servers
  become: true
  tasks:

  - name: install samba package
    tags: samba
    package:
      name: samba
      state: latest
```



```
- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    tags: db,centos,db,mariadb
    yum:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: install mariadb package (Ubuntu)
    tags: db,mariadb,ubuntu
    apt:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true

- hosts: file_servers
  become: true
  tasks:

  - name: install samba package
    tags: samba
    package:
      name: samba
      state: latest
```



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ******************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

TASK [install updates (CentOS)] *********************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] *********************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

PLAY [web_servers] **********************************************************************************************

TASK [Gathering Facts] ******************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for Ubuntu servers] ****************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] ****************************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] ***********************************************************************************************

TASK [Gathering Facts] ******************************************************************************************
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (CentOS)] *************************************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (Ubuntu)] *************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]

TASK [Mariadb- Restarting/Enabling] *****************************************************************************
changed: [192.168.56.111]
changed: [192.168.56.112]

PLAY [file_servers] *********************************************************************************************

TASK [Gathering Facts] ******************************************************************************************
ok: [192.168.56.111]

TASK [install samba package] ************************************************************************************
changed: [192.168.56.111]

PLAY RECAP ******************************************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=7    changed=2    unreachable=0    failed=0
192.168.56.112             : ok=7    changed=1    unreachable=0    failed=0
```

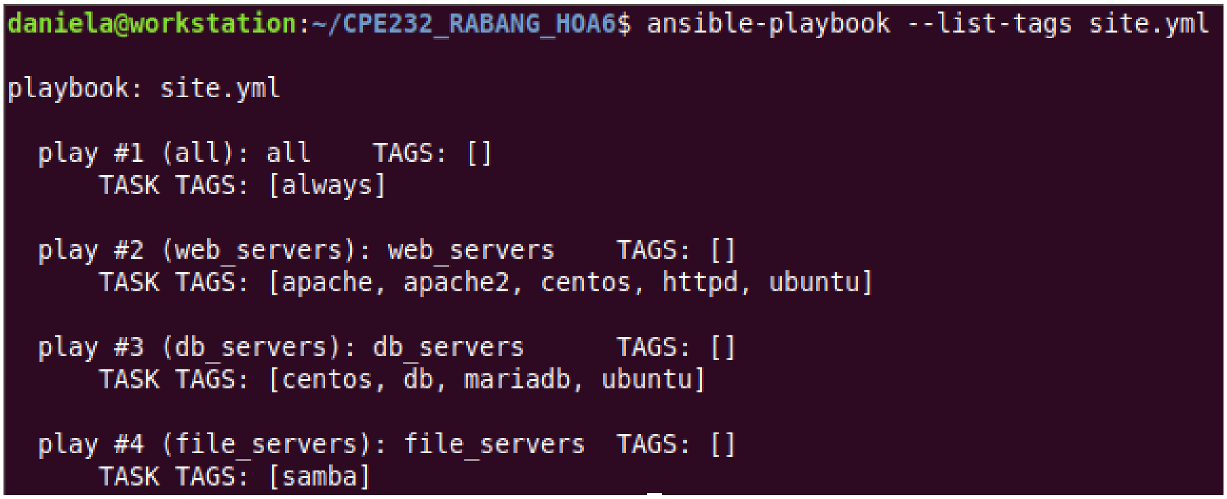
Make sure to save the file and exit.

Run the *site.yml* file and describe the result.

I modified the playbook and also a task where it installed the samba package on file_servers and added a tag where it says samba.

2. On the local machine, try to issue the following commands and describe each result:

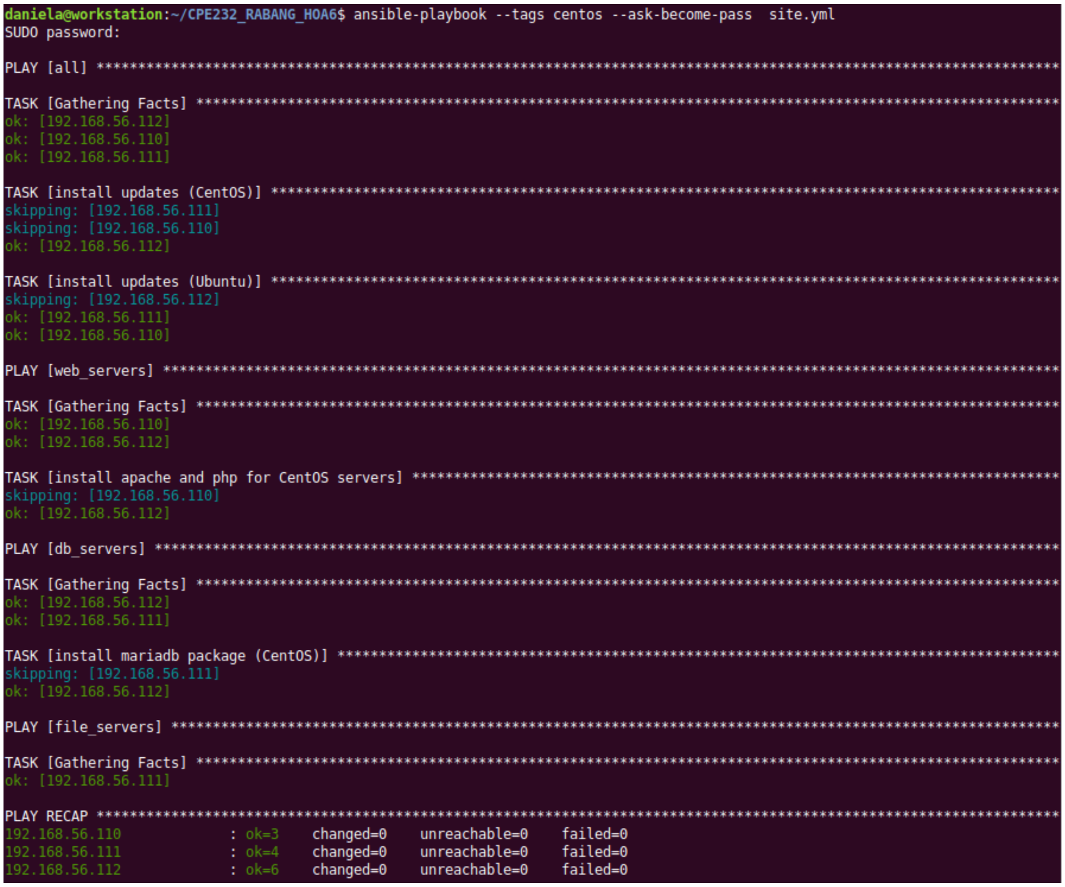
2.1 *ansible-playbook --list-tags site.yml*



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --list-tags site.yml

playbook: site.yml

  play #1 (all): all    TAGS: []
      TASK TAGS: [always]

  play #2 (web_servers): web_servers    TAGS: []
      TASK TAGS: [apache, apache2, centos, httpd, ubuntu]

  play #3 (db_servers): db_servers      TAGS: []
      TASK TAGS: [centos, db, mariadb, ubuntu]

  play #4 (file_servers): file_servers  TAGS: []
      TASK TAGS: [samba]
```

It runs all the tasks that are listed as tags that are in the playbook.

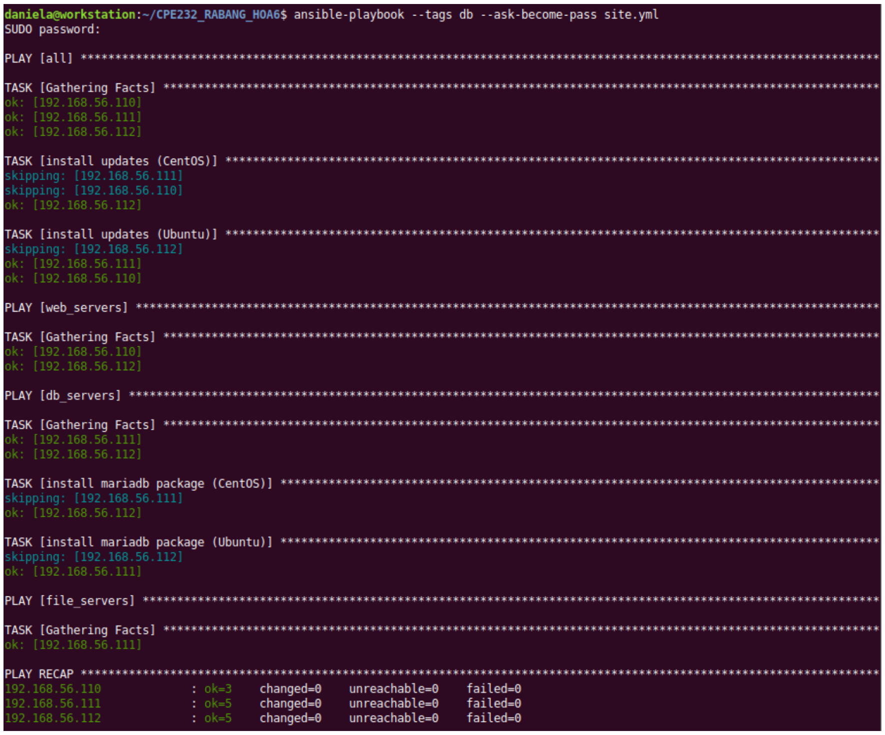
2.2 *ansible-playbook --tags centos --ask-become-pass site.yml*



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --tags centos --ask-become-pass  site.yml
SUDO password:

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

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

TASK [install updates (CentOS)] ************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] ************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] *************************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for CentOS servers] *******************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] **************************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]

TASK [install mariadb package (CentOS)] ****************************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

PLAY [file_servers] ************************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.111]

PLAY RECAP *********************************************************************************************
192.168.56.110             : ok=3    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=6    changed=0    unreachable=0    failed=0
```

It runs all the tasks that are tagged as centos that are in the playbook.

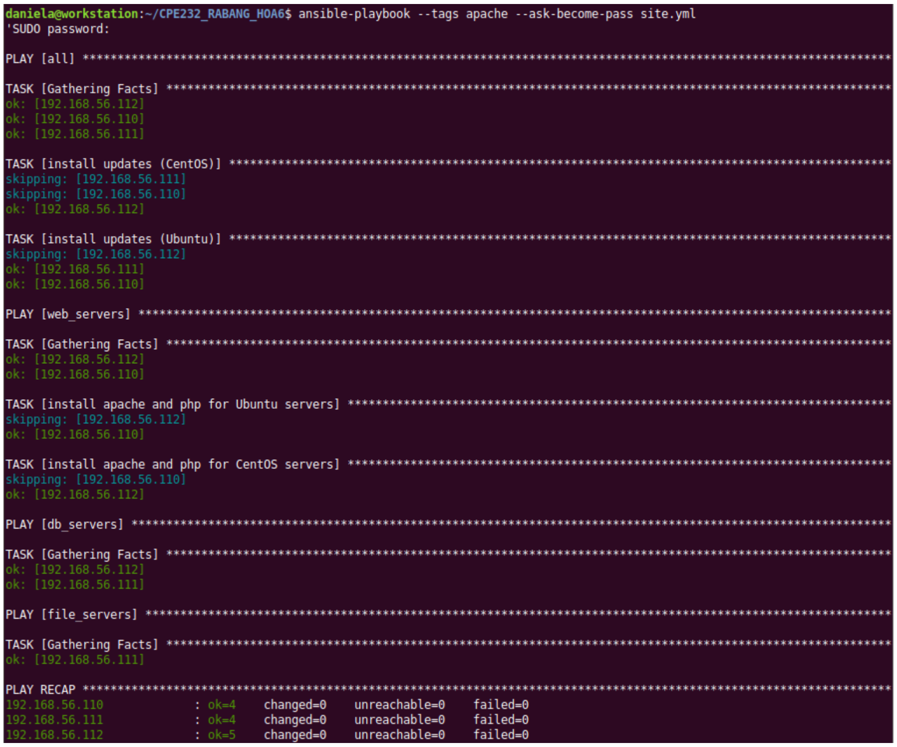
2.3 *ansible-playbook --tags db --ask-become-pass site.yml*



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --tags db --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install updates (CentOS)] ***********************************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] ***********************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] ************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] *************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (CentOS)] ***************************************************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (Ubuntu)] ***************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]

PLAY [file_servers] ***********************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.111]

PLAY RECAP ********************************************************************************************************************
192.168.56.110             : ok=3    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=5    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=5    changed=0    unreachable=0    failed=0
```

It runs all the tasks that are tagged as db that are in the playbook.

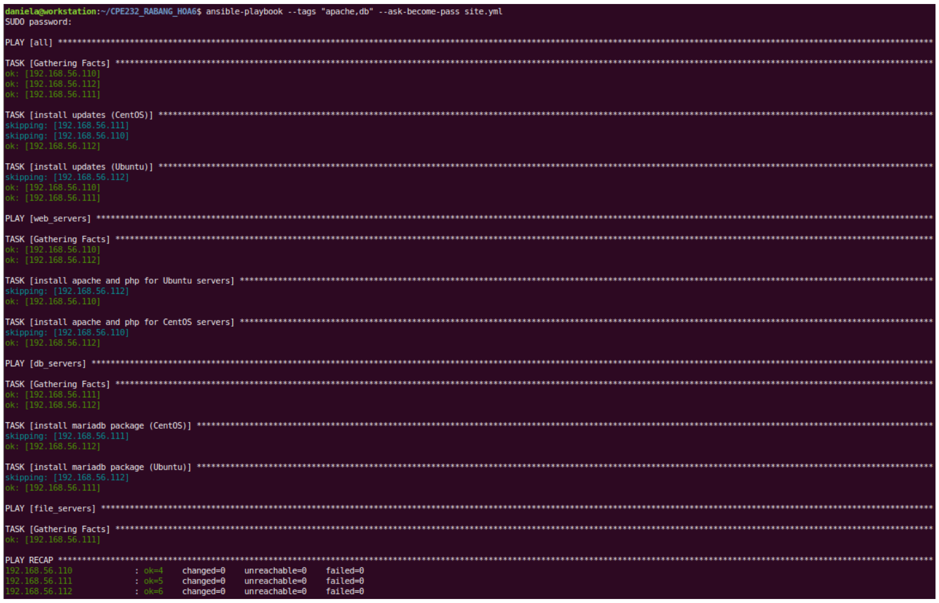
2.4 *ansible-playbook --tags apache --ask-become-pass site.yml*



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --tags apache --ask-become-pass site.yml
'SUDO password:

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

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

TASK [install updates (CentOS)] ***********************************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] ***********************************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]
ok: [192.168.56.110]

PLAY [web_servers] ************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for Ubuntu servers] ******************************************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] ******************************************************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] *************************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]

PLAY [file_servers] ***********************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************************
ok: [192.168.56.111]

PLAY RECAP ********************************************************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=5    changed=0    unreachable=0    failed=0
```

It runs all the tasks that are tagged as apache that are in the playbook.

2.5 *ansible-playbook --tags “apache,db” --ask-become-pass site.yml*

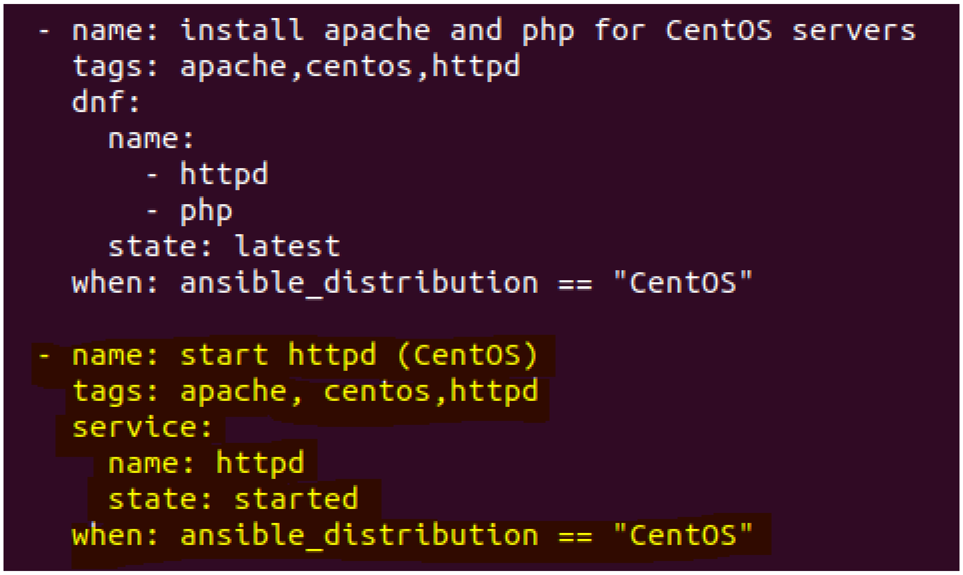


```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --tags "apache,db" --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] **************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]
ok: [192.168.56.111]

TASK [install updates (CentOS)] *****************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [install updates (Ubuntu)] *****************************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]
ok: [192.168.56.111]

PLAY [web_servers] ******************************************************************

TASK [Gathering Facts] **************************************************************
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for Ubuntu servers] ************************************
skipping: [192.168.56.112]
ok: [192.168.56.110]

TASK [install apache and php for CentOS servers] ************************************
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY [db_servers] *******************************************************************

TASK [Gathering Facts] **************************************************************
ok: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (CentOS)] *********************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [install mariadb package (Ubuntu)] *********************************************
skipping: [192.168.56.112]
ok: [192.168.56.111]

PLAY [file_servers] *****************************************************************

TASK [Gathering Facts] **************************************************************
ok: [192.168.56.111]

PLAY RECAP **************************************************************************
192.168.56.110             : ok=4    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=5    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=6    changed=0    unreachable=0    failed=0
```

It runs all the tasks that are tagged as apachedb that are in the playbook.

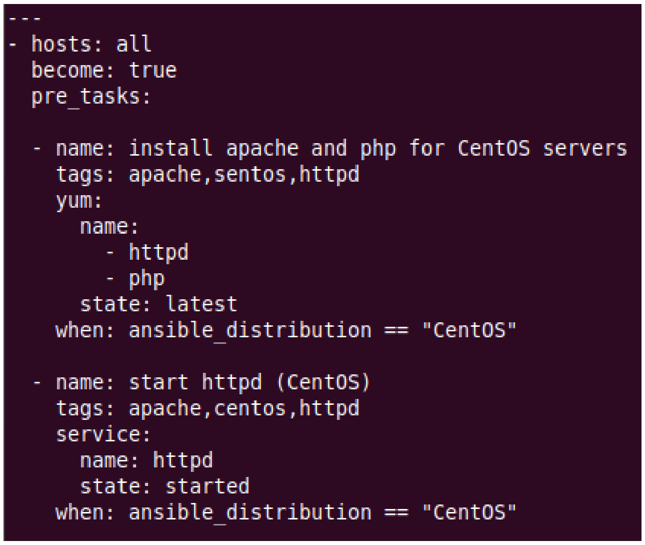
Task 3: Managing Services

1. Edit the file site.yml and add a play that will automatically start the httpd on CentOS server.

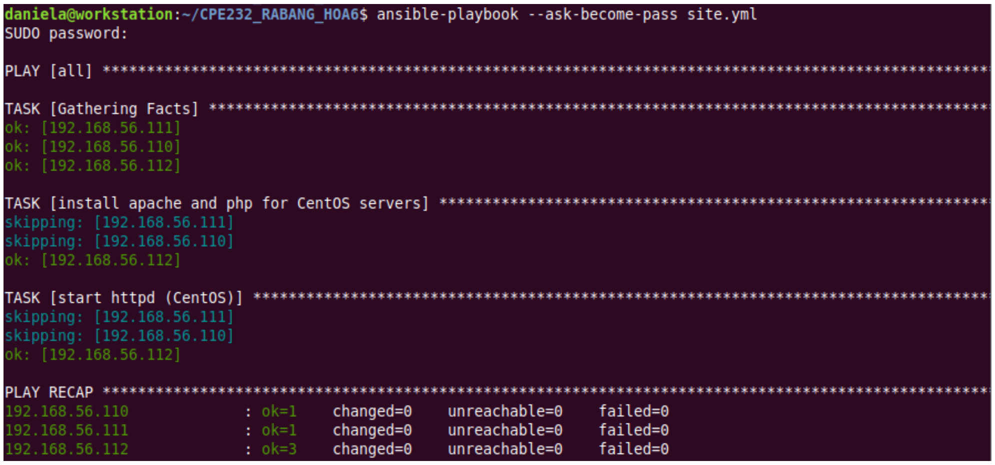


```
- name: install apache and php for CentOS servers
  tags: apache,centos,httpd
  dnf:
    name:
      - httpd
      - php
    state: latest
  when: ansible_distribution == "CentOS"

- name: start httpd (CentOS)
  tags: apache, centos,httpd
  service:
    name: httpd
    state: started
  when: ansible_distribution == "CentOS"
```

Figure 3.1.1
Make sure to save the file and exit.



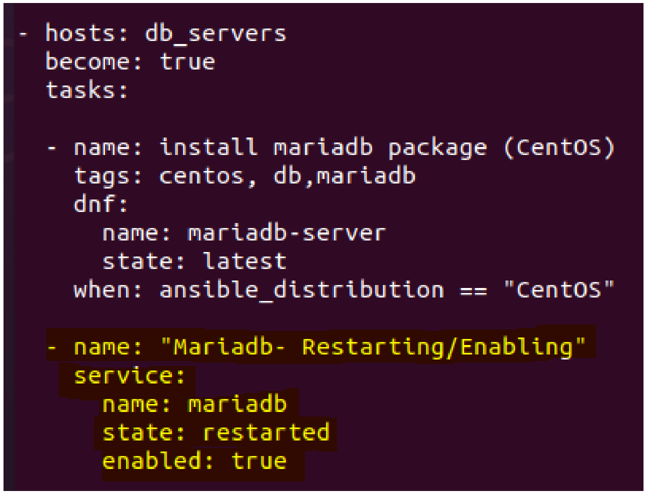
```
---
- hosts: all
  become: true
  pre_tasks:

  - name: install apache and php for CentOS servers
    tags: apache,sentos,httpd
    yum:
      name:
        - httpd
        - php
      state: latest
    when: ansible_distribution == "CentOS"

  - name: start httpd (CentOS)
    tags: apache,centos,httpd
    service:
      name: httpd
      state: started
    when: ansible_distribution == "CentOS"
```



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

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

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.111]
ok: [192.168.56.110]
ok: [192.168.56.112]

TASK [install apache and php for CentOS servers] *******************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

TASK [start httpd (CentOS)] ****************************************************************************
skipping: [192.168.56.111]
skipping: [192.168.56.110]
ok: [192.168.56.112]

PLAY RECAP *********************************************************************************************
192.168.56.110             : ok=1    changed=0    unreachable=0    failed=0
192.168.56.111             : ok=1    changed=0    unreachable=0    failed=0
192.168.56.112             : ok=3    changed=0    unreachable=0    failed=0
```

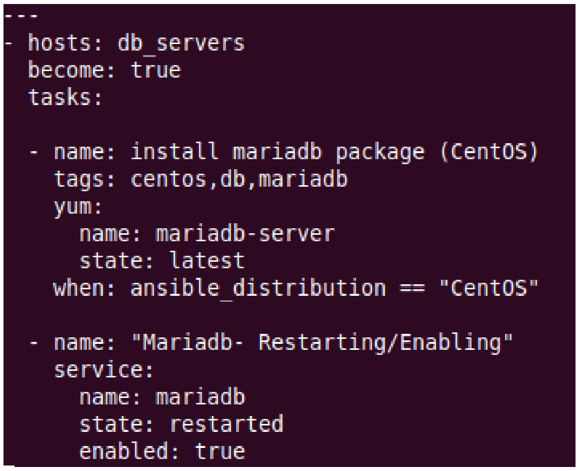
In this part, I had modified the playbook and inserted the following tasks that start httpd on CentOS. It also includes tags and the state is also stated.

You would also notice from our previous activity that we already created a module that runs a service.

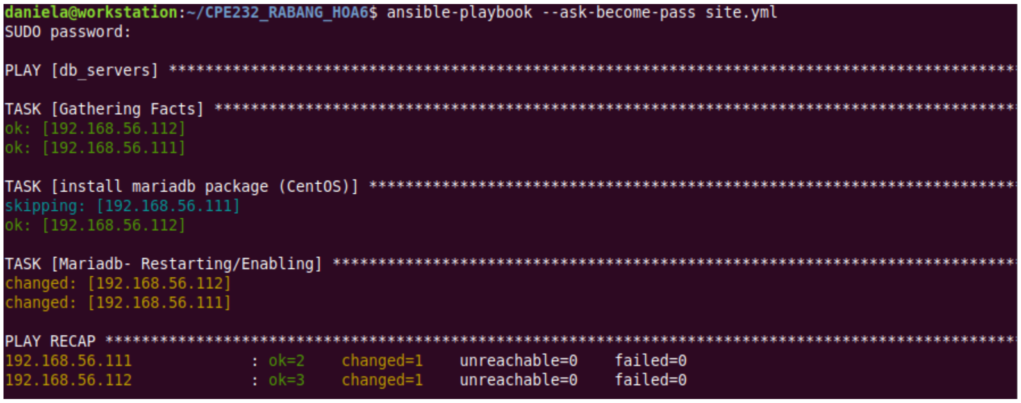


```
- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    tags: centos, db,mariadb
    dnf:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true
```

Figure 3.1.2



```
---
- hosts: db_servers
  become: true
  tasks:

  - name: install mariadb package (CentOS)
    tags: centos,db,mariadb
    yum:
      name: mariadb-server
      state: latest
    when: ansible_distribution == "CentOS"

  - name: "Mariadb- Restarting/Enabling"
    service:
      name: mariadb
      state: restarted
      enabled: true
```

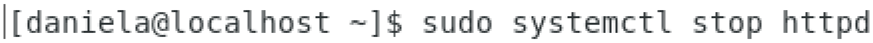


```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

PLAY [db_servers] ******************************************************************************

TASK [Gathering Facts] *************************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]

TASK [install mariadb package (CentOS)] ********************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [Mariadb- Restarting/Enabling] ************************************************************
changed: [192.168.56.112]
changed: [192.168.56.111]

PLAY RECAP *************************************************************************************
192.168.56.111             : ok=2    changed=1    unreachable=0    failed=0
192.168.56.112             : ok=3    changed=1    unreachable=0    failed=0
```

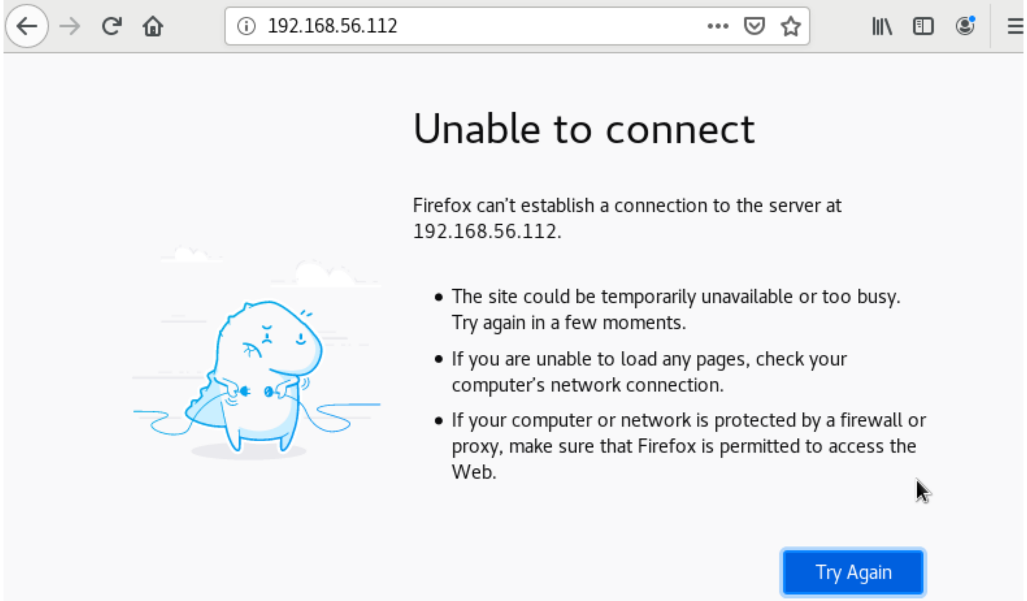
In this part, I created a task where it restarts and enables mariadb.

This is because in CentOS, installed packages' services are not run automatically. Thus, we need to create the module to run it automatically.

2. To test it, before you run the saved playbook, go to the CentOS server and stop the currently running httpd using the command *sudo systemctl stop httpd.* When prompted, enter the sudo password. After that, open the browser and enter the CentOS server's IP address. You should not be getting a display because we stopped the httpd service already.



```
[daniela@localhost ~]$ sudo systemctl stop httpd
```



Unable to connect

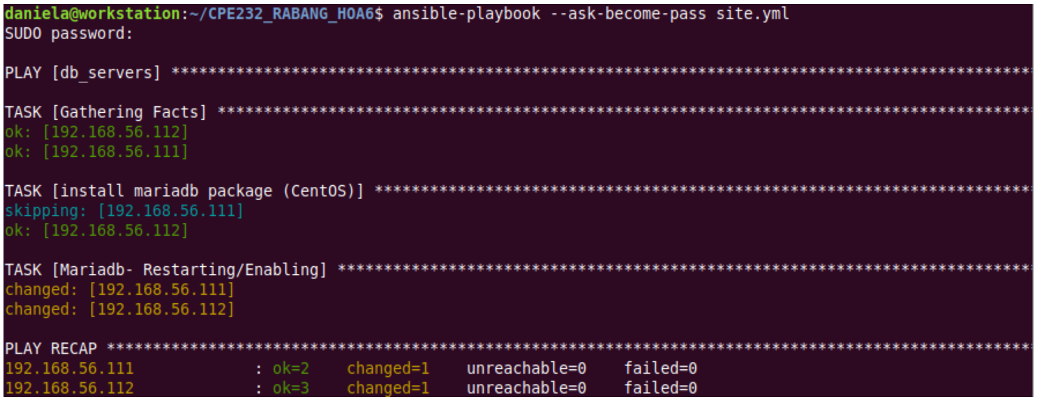
Firefox can't establish a connection to the server at 192.168.56.112.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

Since the httpd is stopped, the output shows that I cannot be able to connect since it had been stopped using the command sudo systemctl stop httpd.

3. Go to the local machine and this time, run the *site.yml* file. Then after running the file, go again to the CentOS server and enter its IP address on the browser. Describe the result.

   To automatically enable the service every time we run the playbook, use the command *enabled: true* similar to Figure 7.1.2 and save the playbook.



```
daniela@workstation:~/CPE232_RABANG_HOA6$ ansible-playbook --ask-become-pass site.yml
SUDO password:

PLAY [db_servers] **************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.112]
ok: [192.168.56.111]

TASK [install mariadb package (CentOS)] ****************************************************
skipping: [192.168.56.111]
ok: [192.168.56.112]

TASK [Mariadb- Restarting/Enabling] ********************************************************
changed: [192.168.56.111]
changed: [192.168.56.112]

PLAY RECAP *********************************************************************************
192.168.56.111             : ok=2    changed=1    unreachable=0    failed=0
192.168.56.112             : ok=3    changed=1    unreachable=0    failed=0
```

This is the results when the playbook is being played after stopping the httpd.

Reflections:

Answer the following:

1. What is the importance of putting our remote servers into groups?
   - The importance of putting remote servers into groups is to manage it more easily when it is more organized. Also it is much easier for giving tasks that need to be runned from different servers at the same time.

2. What is the importance of tags in playbooks?
   - The tags on playbook are important for a certain reason, tags are used to allow the user to pick a certain task to be runned or not to run.

3. Why do you think some services need to be managed automatically in playbooks?
   - For the user's convenience, other services need to be maintained automatically via playbooks. This boosts consistency and efficiency. Additionally, it improves security and compliance while lowering operational costs.

Conclusion:

In this hands-on activity which is all about targeting specific nodes and managing services, I have created a new repository that is named CPE232_RABANG_HOA6. Then I had cloned it in my workstation so that I can manage the repository that I had created through the use of my workstation. So first I had created a playbook where it runs tasks that can install apache and php on both Ubuntu and CentOS. Then I edited the inventory file where I created three group variables that is named after web_servers, db_servers, and file_servers. I also included thet ansible_python_interperter to make the playbook run without any error. Then I modified the playbook and added new features like adding pre_tasks and adding the host: web_servers where the following tasks that are below it can be runned on all the servers that are in the web_servers category. Then I modified it again and assigns tasks that can be runned for the db_servers which it installs mariadb package on both CentOS and Ubuntu. To verify that the mariadb package is installed for the db servers which is the server 2 and the CentOS server, I checked it using the command systemctl status mariadb for both server 2 and CentOS. Then I had modified the playbook again and added a tasks where it assigned on the file_servers. This is to install samba package on the file_servers. Then on the next task, I am tasked to run playbooks using tasks. Then i run the playbook using tags that are centos, db, apache, and apache, db. Then for the next task, which is the task number 3 I had manage the server to start tasks immediately. Therefore, I can say that after I had finished this hands-on activity, I learned a lot of thing from editing playbooks that satisfy certain tasks for a certain group of servers to be runned to the use of tags on playbooks.