Name: John Era Roldan	Date Performed: 9/19/2025
Course/Section: CPE212 CPE31S2	Date Submitted: 9/19/2025
Instructor: Engr. Robin Valenzuela	Semester and SY:

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.

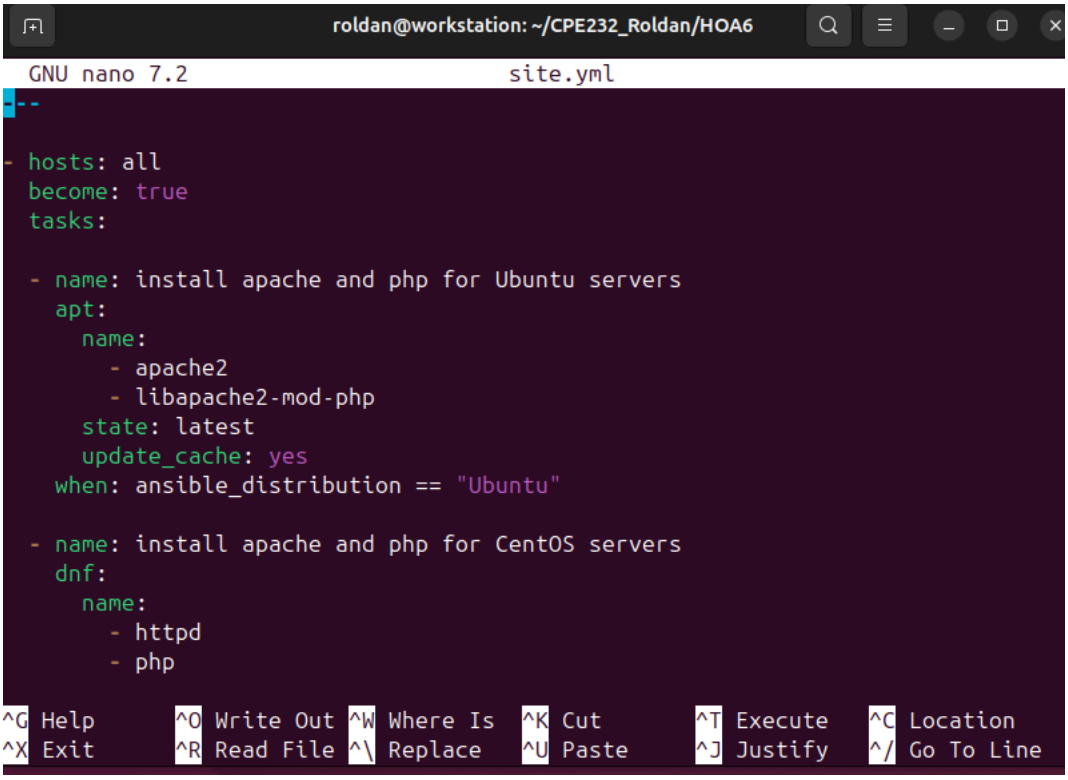
Requirement:

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.

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"
```

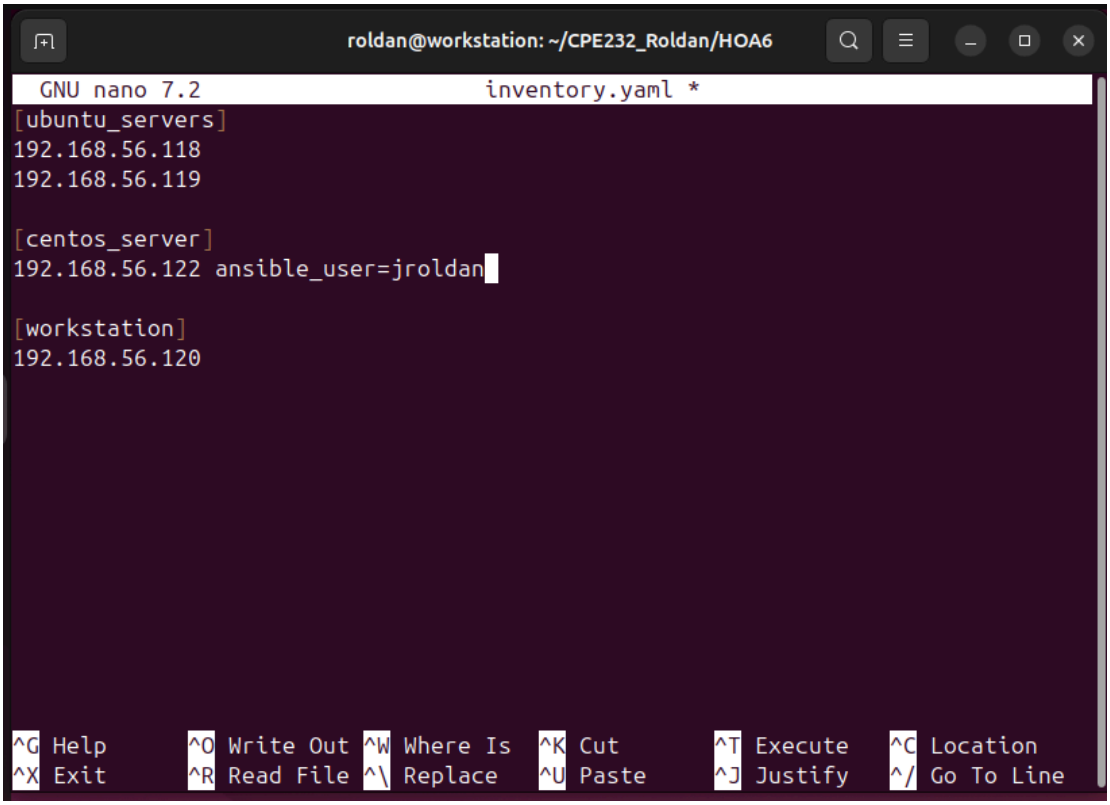


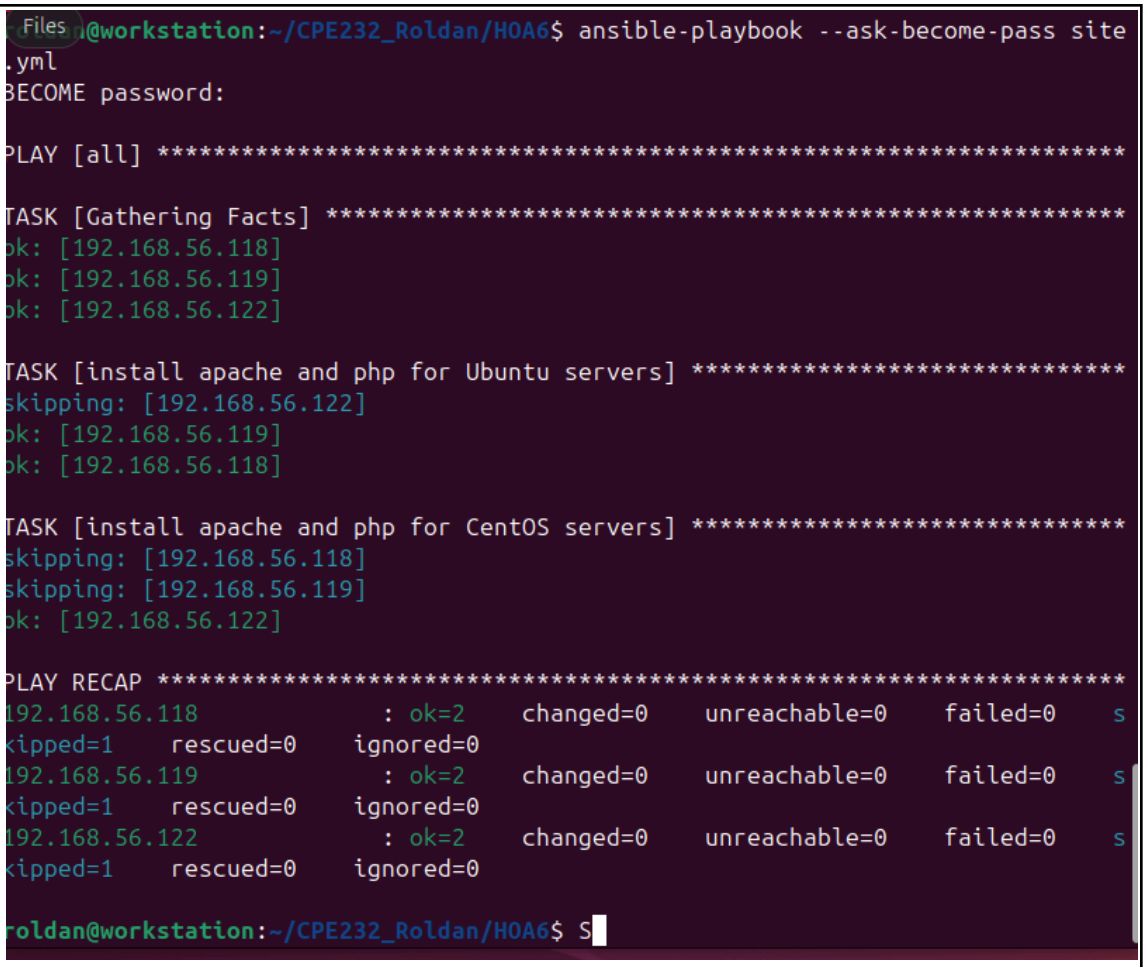

```
      state: latest
    when: ansible_distribution == "CentOS"
```

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
```

Make sure to save the file and exit.



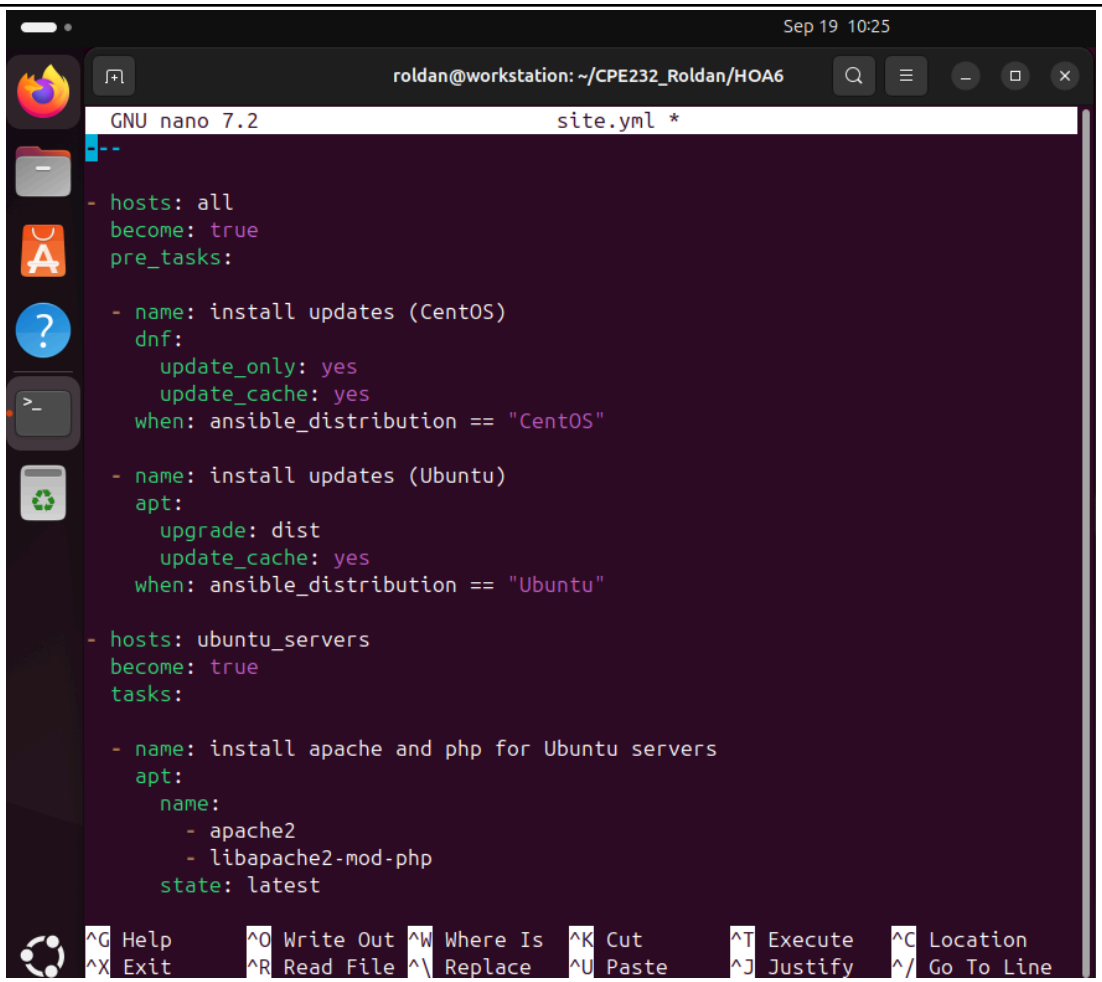


Right now, we have created groups in our inventory file and put each server in its own group. In other cases, you can have a server be a member of multiple groups, for example you have a test server that is also a web server.

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.




```
    when: ansible_distribution == "Ubuntu"

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

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.

```
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --ask-become-pass site
.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.122]
changed: [192.168.56.119]
changed: [192.168.56.118]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
```

```
TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

TASK [install apache and php for Ubuntu servers] *******************************
ok: [192.168.56.119]
ok: [192.168.56.118]

TASK [install apache and php for CentOS servers] *******************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]

PLAY RECAP *********************************************************************
192.168.56.118             : ok=4    changed=1    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=4    changed=1    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.122             : ok=2    changed=0    unreachable=0    failed=0    s
kipped=1    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

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.

```yaml
- hosts: centos_server
  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"
```

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

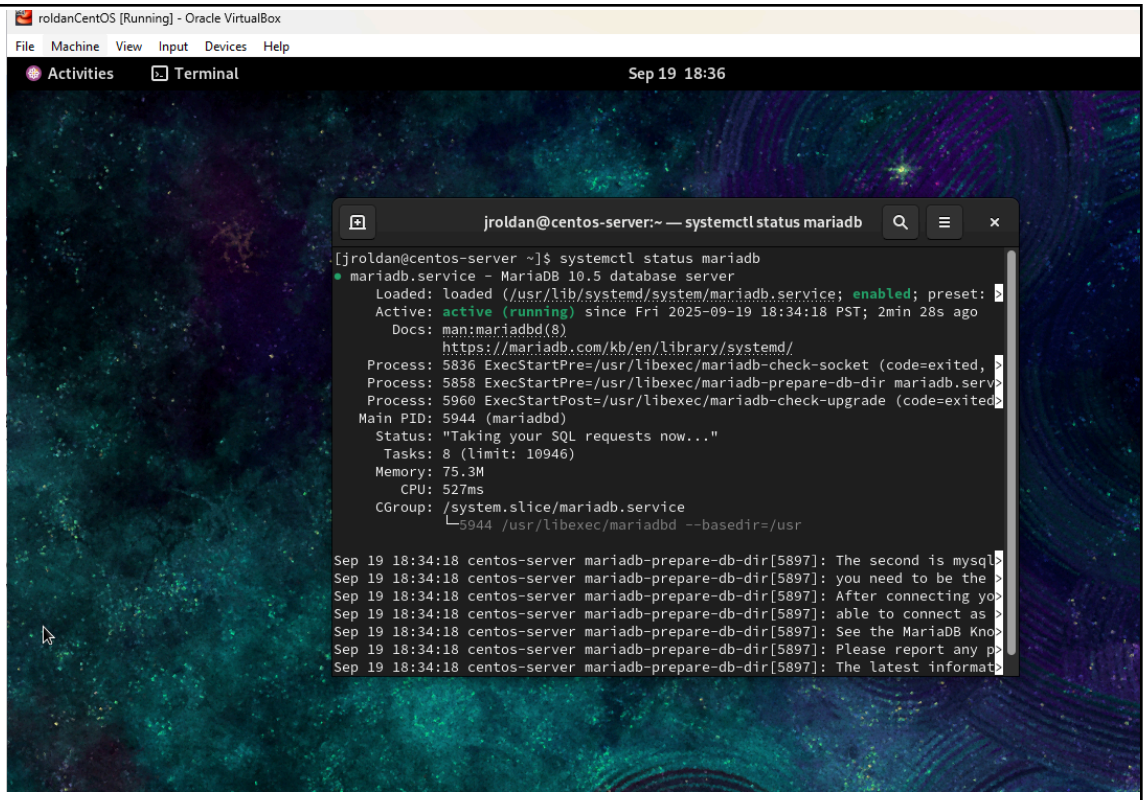
```
Files @workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --ask-become-pass site
.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.122]
ok: [192.168.56.119]
ok: [192.168.56.118]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

TASK [install apache and php for Ubuntu servers] *******************************
ok: [192.168.56.118]
ok: [192.168.56.119]
```

```
TASK [install apache and php for CentOS servers] ******************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]

PLAY [centos_server] **********************************************************

TASK [Gathering Facts] ********************************************************
ok: [192.168.56.122]

TASK [install mariadb package (CentOS)] ***************************************
changed: [192.168.56.122]

TASK [Mariadb - Restarting/Enabling] ******************************************
changed: [192.168.56.122]

TASK [install mariadb package (Ubuntu)] ***************************************
skipping: [192.168.56.122]

PLAY RECAP ********************************************************************
192.168.56.118             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.122             : ok=5    changed=2    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

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

   Describe the output.

```
roldan@workstation:~/CPE232_Roldan/HOA6$ systemctl status mari
Unit mariadb.service could not be found.
roldan@workstation:~/CPE232_Roldan/HOA6$
```



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.

```
- hosts: ubuntu_servers
  become: true
  tasks:

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

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

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.

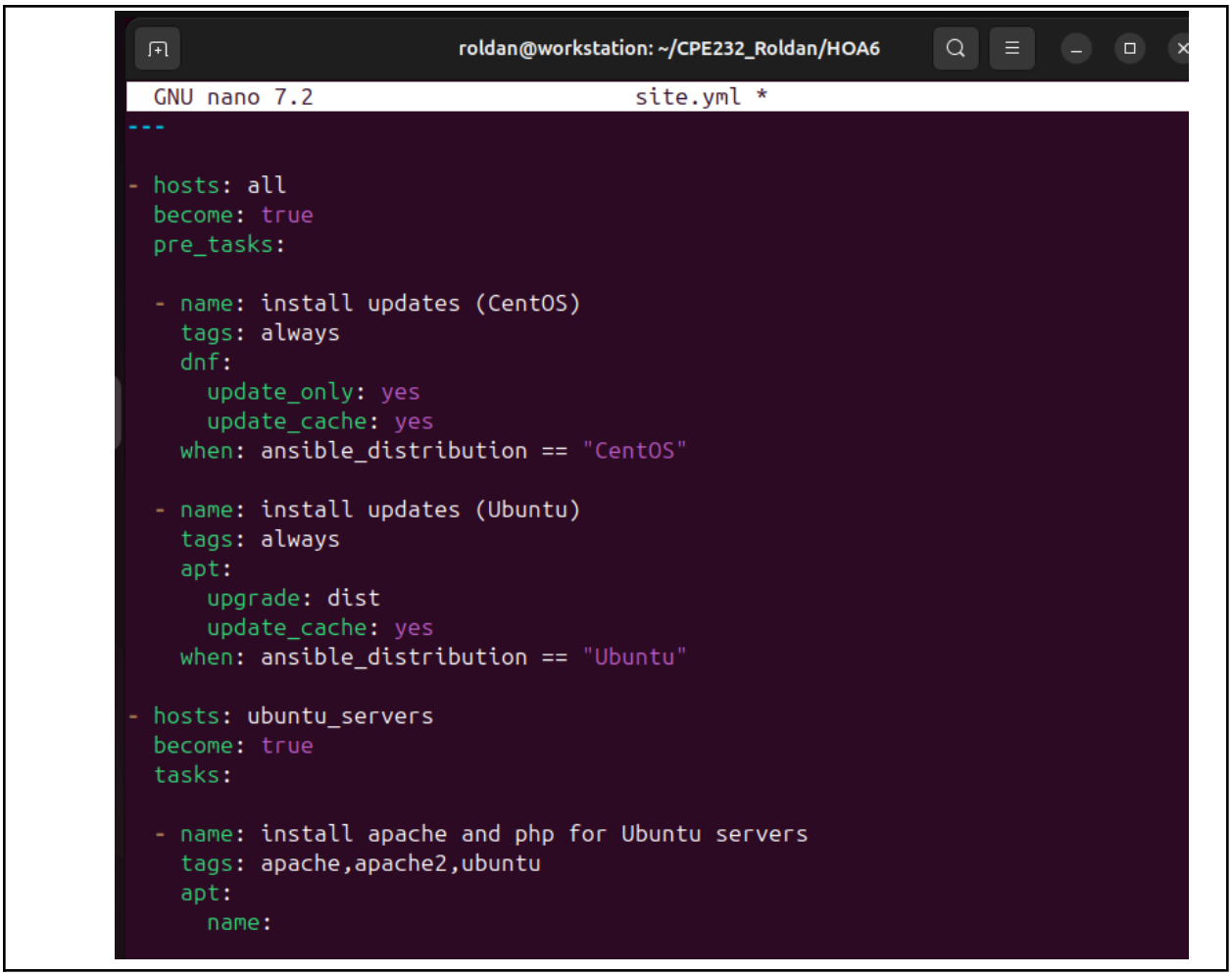
1. Edit the *site.yml* file. Add tags to the playbook. After the name, we can place the tags: *name_of_tag*. This is an arbitrary command, which means you can use any name for a tag.

```yaml
---

- 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"
```

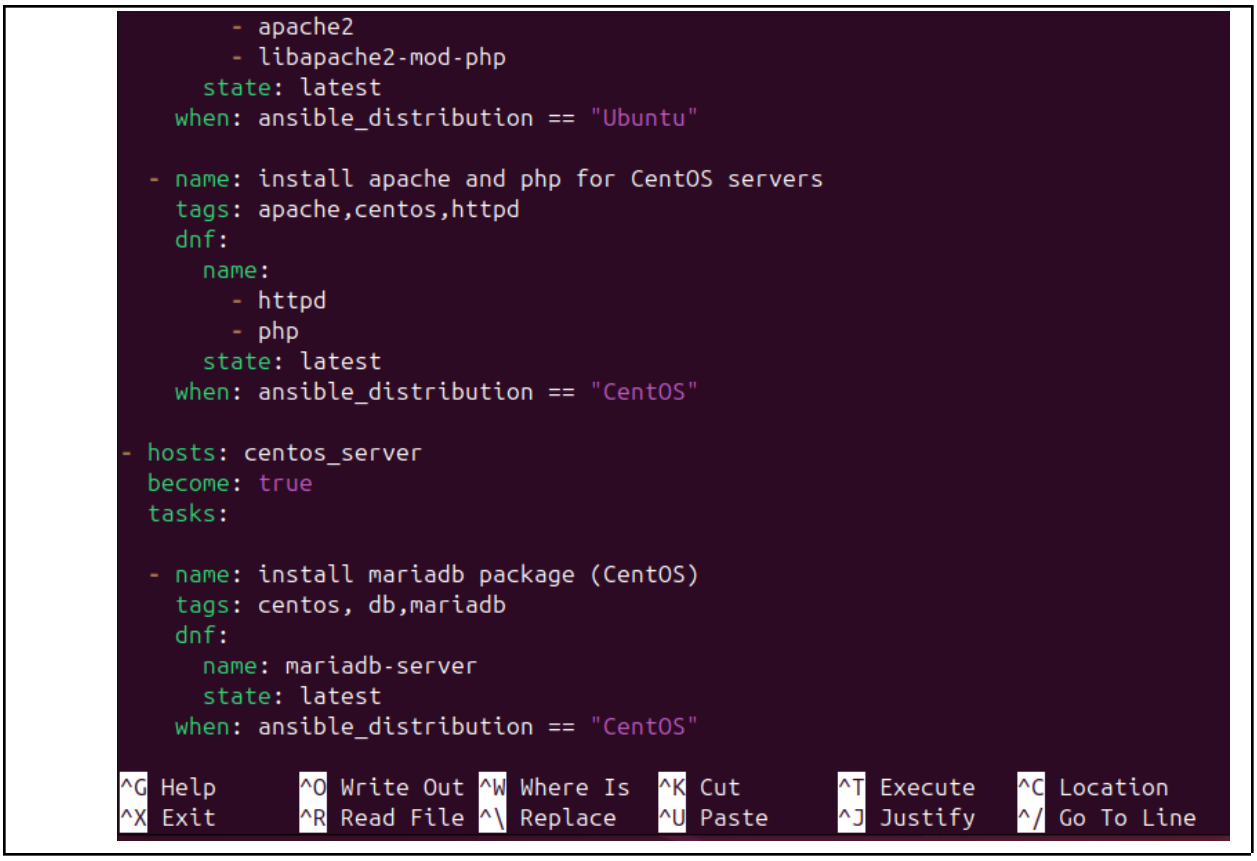
```yaml
- hosts: web_servers
  become: true
  tasks:

  - 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: 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
```

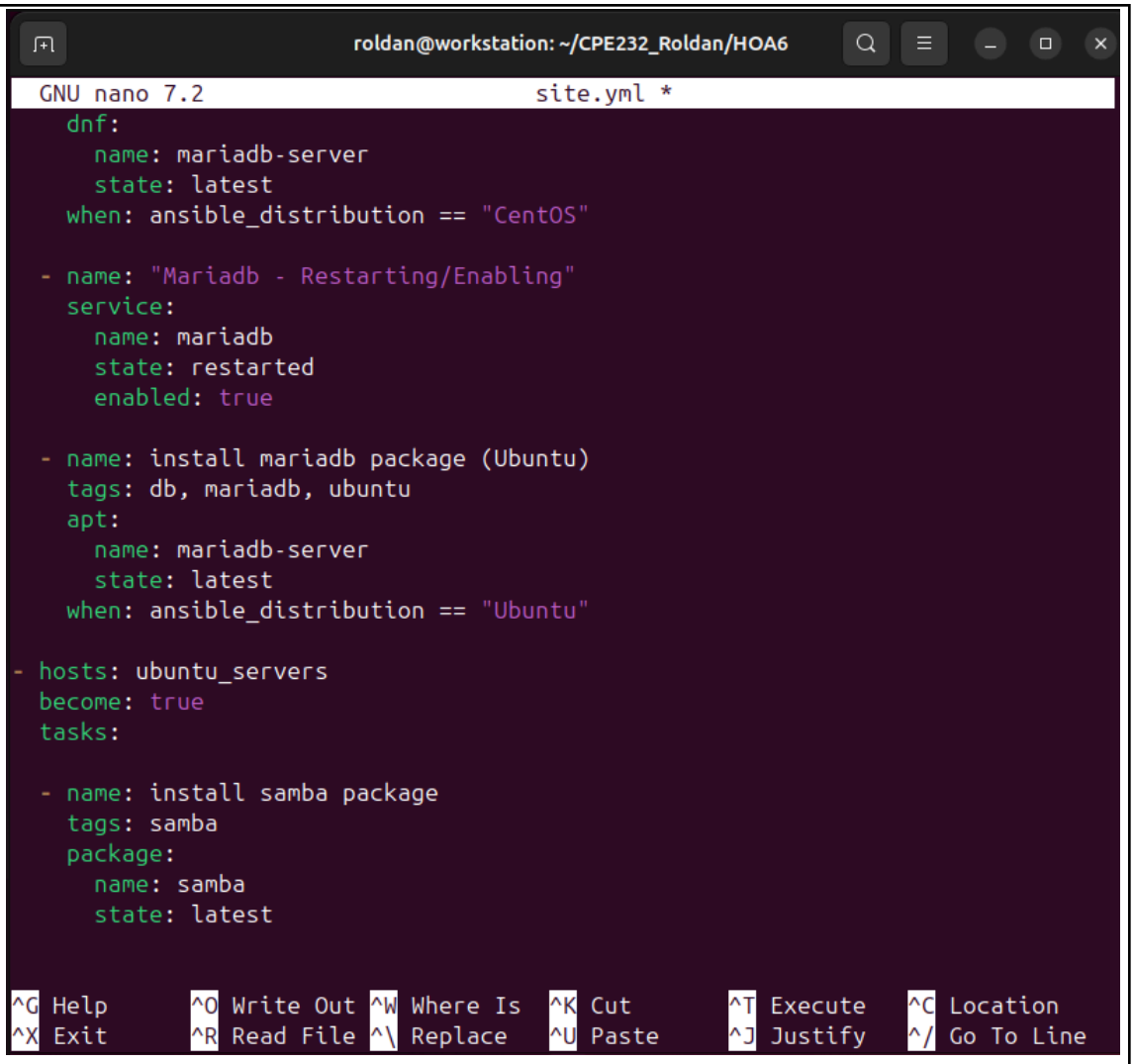
Make sure to save the file and exit.


roldan@workstation: ~/CPE232_Roldan/HOA6
GNU nano 7.2 site.yml *

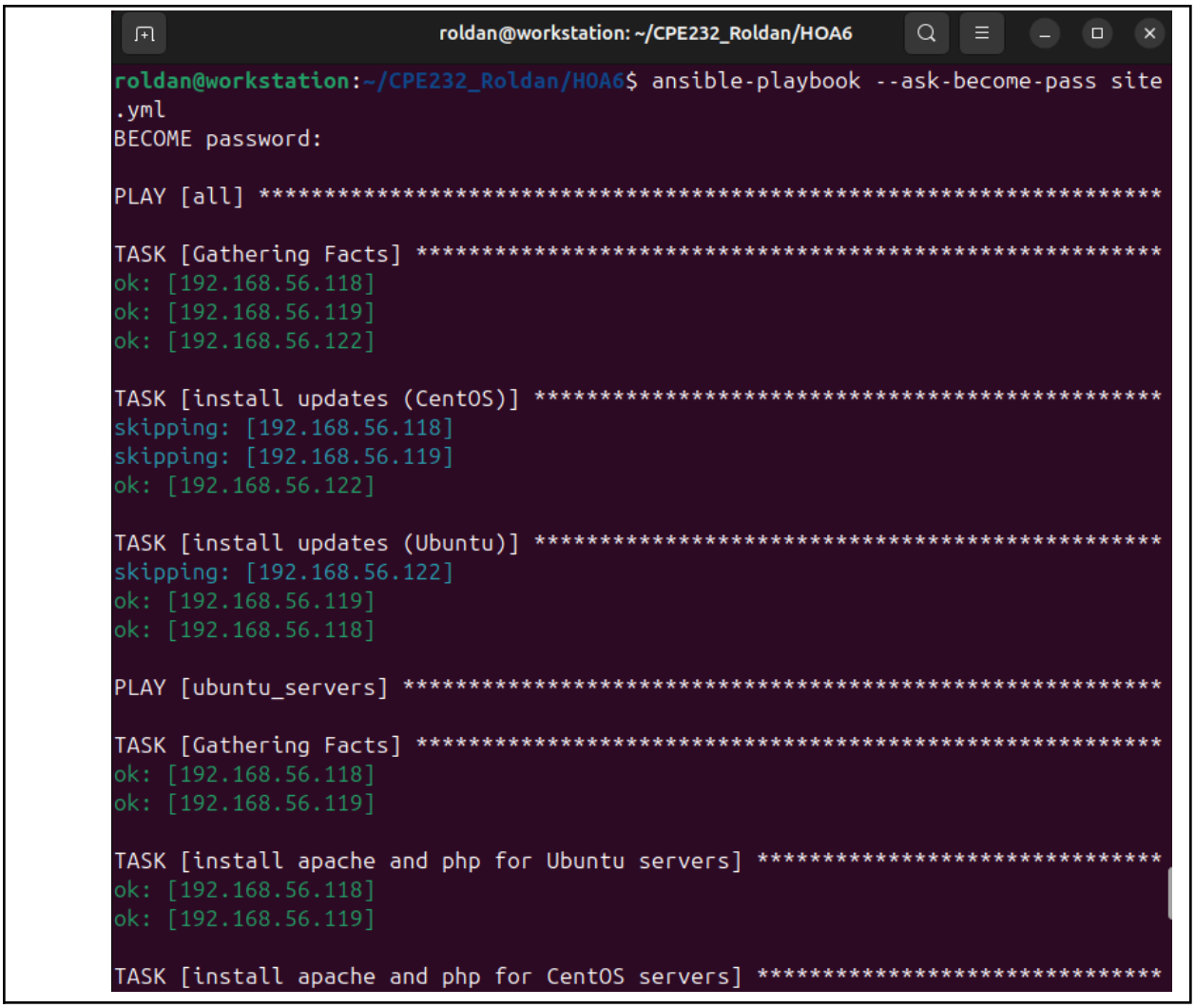
- hosts: all
become: true
pre_tasks:
- name: install updates (CentOS)
tags: always
dnf:
update_only: yes
update_cache: yes
when: ansible_distribution == "CentOS"
- name: install updates (Ubuntu)
tags: always
apt:
upgrade: dist
update_cache: yes
when: ansible_distribution == "Ubuntu"
- hosts: ubuntu_servers
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: centos_server
  become: true
  tasks:

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

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```



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


roldan@workstation: ~/CPE232_Roldan/HOA6
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --ask-become-pass site
.yml
BECOME password:
PLAY [all] **
TASK [Gathering Facts] **
ok: [192.168.56.118]
ok: [192.168.56.119]
ok: [192.168.56.122]
TASK [install updates (CentOS)] ***
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]
TASK [install updates (Ubuntu)] ***
skipping: [192.168.56.122]
ok: [192.168.56.119]
ok: [192.168.56.118]
PLAY [ubuntu_servers] ***
TASK [Gathering Facts] **
ok: [192.168.56.118]
ok: [192.168.56.119]
TASK [install apache and php for Ubuntu servers] ******************************
ok: [192.168.56.118]
ok: [192.168.56.119]
TASK [install apache and php for CentOS servers] ******************************

```
TASK [install apache and php for CentOS servers] ******************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]

PLAY [centos_server] **********************************************************
Help
TASK [Gathering Facts] ********************************************************
ok: [192.168.56.122]

TASK [install mariadb package (CentOS)] ***************************************
ok: [192.168.56.122]

TASK [Mariadb - Restarting/Enabling] ******************************************
changed: [192.168.56.122]

TASK [install mariadb package (Ubuntu)] ***************************************
skipping: [192.168.56.122]

PLAY [ubuntu_servers] *********************************************************

TASK [Gathering Facts] ********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

TASK [install samba package] **************************************************
ok: [192.168.56.119]
ok: [192.168.56.118]

PLAY RECAP ********************************************************************
```

```
PLAY RECAP ********************************************************************
192.168.56.118             : ok=6    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=6    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.122             : ok=5    changed=1    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

2. On the local machine, try to issue the following commands and describe each result:
   2.1 *ansible-playbook --list-tags site.yml*

```
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --list-tags site.yml

playbook: site.yml

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

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

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

  play #4 (ubuntu_servers): ubuntu_servers      TAGS: []
      TASK TAGS: [samba]
roldan@workstation:~/CPE232_Roldan/HOA6$
```

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

```
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --tags centos --ask-be
come-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.122]
ok: [192.168.56.118]
ok: [192.168.56.119]

PLAY [ubuntu_servers] **********************************************************
```

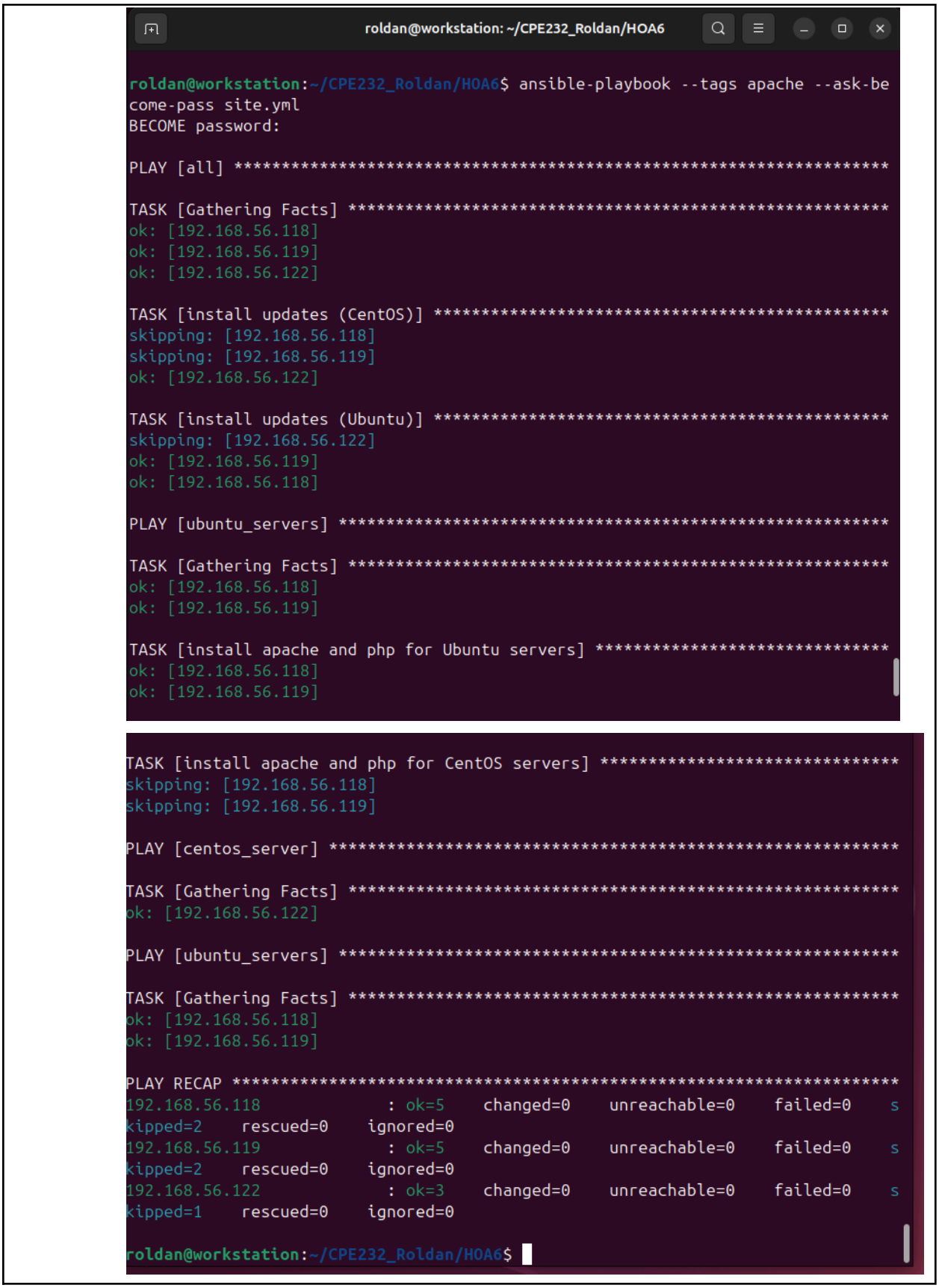
```
TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

TASK [install apache and php for CentOS servers] *******************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]

PLAY [centos_server] ***********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.122]

TASK [install mariadb package (CentOS)] ****************************************
ok: [192.168.56.122]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

PLAY RECAP *********************************************************************
192.168.56.118             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.122             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=1    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

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

```
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --tags db --ask-become
-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.119]
ok: [192.168.56.118]
ok: [192.168.56.122]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.122]
ok: [192.168.56.118]
ok: [192.168.56.119]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
```

```
TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

PLAY [centos_server] ***********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.122]

TASK [install mariadb package (CentOS)] ****************************************
ok: [192.168.56.122]

TASK [install mariadb package (Ubuntu)] ****************************************
skipping: [192.168.56.122]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.119]
ok: [192.168.56.118]

PLAY RECAP *********************************************************************
192.168.56.118             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=1    rescued=0    ignored=0
192.168.56.119             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=1    rescued=0    ignored=0
192.168.56.122             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

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


roldan@workstation: ~/CPE232_Roldan/HOA6
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --tags apache --ask-become-pass site.yml
BECOME password:
PLAY [all] ***
TASK [Gathering Facts] ***
ok: [192.168.56.118]
ok: [192.168.56.119]
ok: [192.168.56.122]
TASK [install updates (CentOS)] **
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]
TASK [install updates (Ubuntu)] **
skipping: [192.168.56.122]
ok: [192.168.56.119]
ok: [192.168.56.118]
PLAY [ubuntu_servers] **
TASK [Gathering Facts] ***
ok: [192.168.56.118]
ok: [192.168.56.119]
TASK [install apache and php for Ubuntu servers] *********************************
ok: [192.168.56.118]
ok: [192.168.56.119]
TASK [install apache and php for CentOS servers] *********************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
PLAY [centos_server] ***
TASK [Gathering Facts] ***
ok: [192.168.56.122]
PLAY [ubuntu_servers] **
TASK [Gathering Facts] ***
ok: [192.168.56.118]
ok: [192.168.56.119]
PLAY RECAP ***
192.168.56.118 : ok=5 changed=0 unreachable=0 failed=0 skipped=2 rescued=0 ignored=0
192.168.56.119 : ok=5 changed=0 unreachable=0 failed=0 skipped=2 rescued=0 ignored=0
192.168.56.122 : ok=3 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
roldan@workstation:~/CPE232_Roldan/HOA6$

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

```
roldan@workstation:~/CPE232_Roldan/HOA6$ ansible-playbook --tags "apache,db" --a
sk-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.119]
ok: [192.168.56.118]
ok: [192.168.56.122]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]
ok: [192.168.56.122]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.122]
ok: [192.168.56.118]
ok: [192.168.56.119]

PLAY [ubuntu_servers] **********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.118]
ok: [192.168.56.119]

TASK [install apache and php for Ubuntu servers] *******************************
ok: [192.168.56.119]
```

```
TASK [install apache and php for Ubuntu servers] ******************************
ok: [192.168.56.119]
ok: [192.168.56.118]

TASK [install apache and php for CentOS servers] ******************************
skipping: [192.168.56.118]
skipping: [192.168.56.119]

PLAY [centos_server] **********************************************************

TASK [Gathering Facts] ********************************************************
ok: [192.168.56.122]

TASK [install mariadb package (CentOS)] ***************************************
ok: [192.168.56.122]

TASK [install mariadb package (Ubuntu)] ***************************************
skipping: [192.168.56.122]

PLAY [ubuntu_servers] *********************************************************

TASK [Gathering Facts] ********************************************************
ok: [192.168.56.119]
ok: [192.168.56.118]

PLAY RECAP ********************************************************************
192.168.56.118             : ok=5    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=5    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
```

```
PLAY RECAP ********************************************************************
192.168.56.118             : ok=5    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.119             : ok=5    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
192.168.56.122             : ok=4    changed=0    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0

roldan@workstation:~/CPE232_Roldan/HOA6$
```

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.

```
- 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"
```

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
```

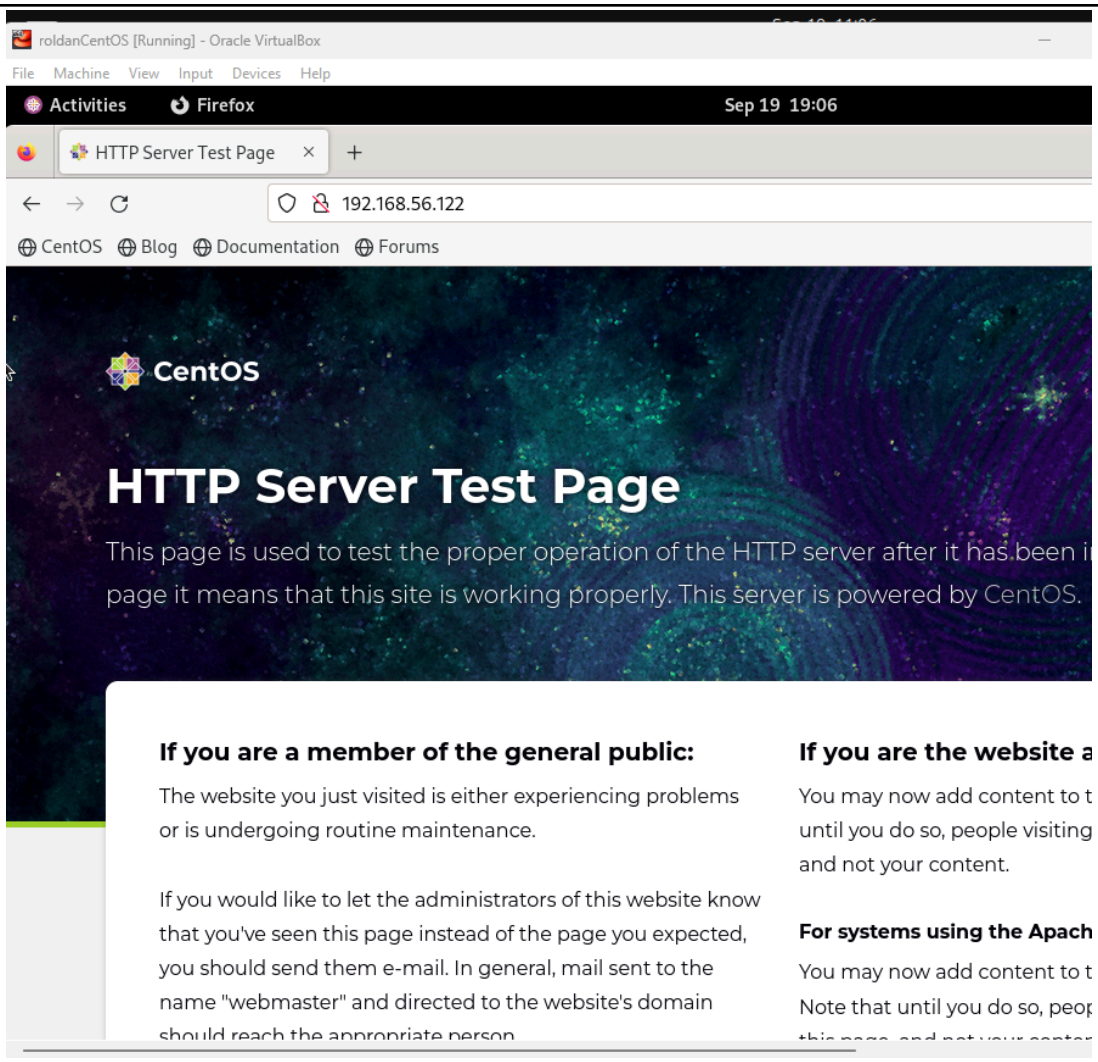
Figure 3.1.2

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.

```
[jroldan@centos-server ~]$ sudo systemctl stop httpd
[sudo] password for jroldan:
[jroldan@centos-server ~]$
```

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.


roldanCentOS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities
Firefox
Sep 19 19:06
HTTP Server Test Page
192.168.56.122
CentOS
Blog
Documentation
Forums
CentOS
HTTP Server Test Page
This page is used to test the proper operation of the HTTP server after it has been i
page it means that this site is working properly. This server is powered by CentOS.
If you are a member of the general public:
The website you just visited is either experiencing problems or is undergoing routine maintenance.
If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.
If you are the website a
You may now add content to t
until you do so, people visiting
and not your content.
For systems using the Apach
You may now add content to t
Note that until you do so, peop

```
roldan@workstation:~/CPE232_Roldan/HOA6$ git add .
roldan@workstation:~/CPE232_Roldan/HOA6$ git commit -m "HOA6.1"
[main 18a59b4] HOA6.1
 3 files changed, 91 insertions(+)
 create mode 100644 HOA6/ansible.cfg
 create mode 100644 HOA6/inventory.yaml
 create mode 100644 HOA6/site.yml
roldan@workstation:~/CPE232_Roldan/HOA6$ git push origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 872 bytes | 872.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object
To github.com:johnera98/CPE232_Roldan.git
   ca4567b..18a59b4  main -> main
roldan@workstation:~/CPE232_Roldan/HOA6$
```

Reflections:

Answer the following:

1. What is the importance of putting our remote servers into groups?
   - Grouping remote servers makes managing them much easier. Instead of configuring or updating each server one by one, you can apply the same tasks to a whole group at once..

2. What is the importance of tags in playbooks?
   - Tags help us run only specific parts of a playbook without going through the whole thing.

3. Why do think some services need to be managed automatically in playbooks?
   - Some services are critical and need to be up and running all the time. Managing them automatically through playbooks ensures they start correctly, restart when needed, and are always set up the right way. This helps avoid downtime and keeps everything working smoothly without relying on someone to do it manually every time.