| | |
|---|---|
| **Name:** Repani, Justin Jello J. | **Date Performed:** September 28, 2023 |
| **Course/Section:** CPE31S6 | **Date Submitted:** September 28, 2023 |
| **Instructor:** Dr. Jonathan V. Taylar | **Semester and SY:** 1st Sem: SY 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.

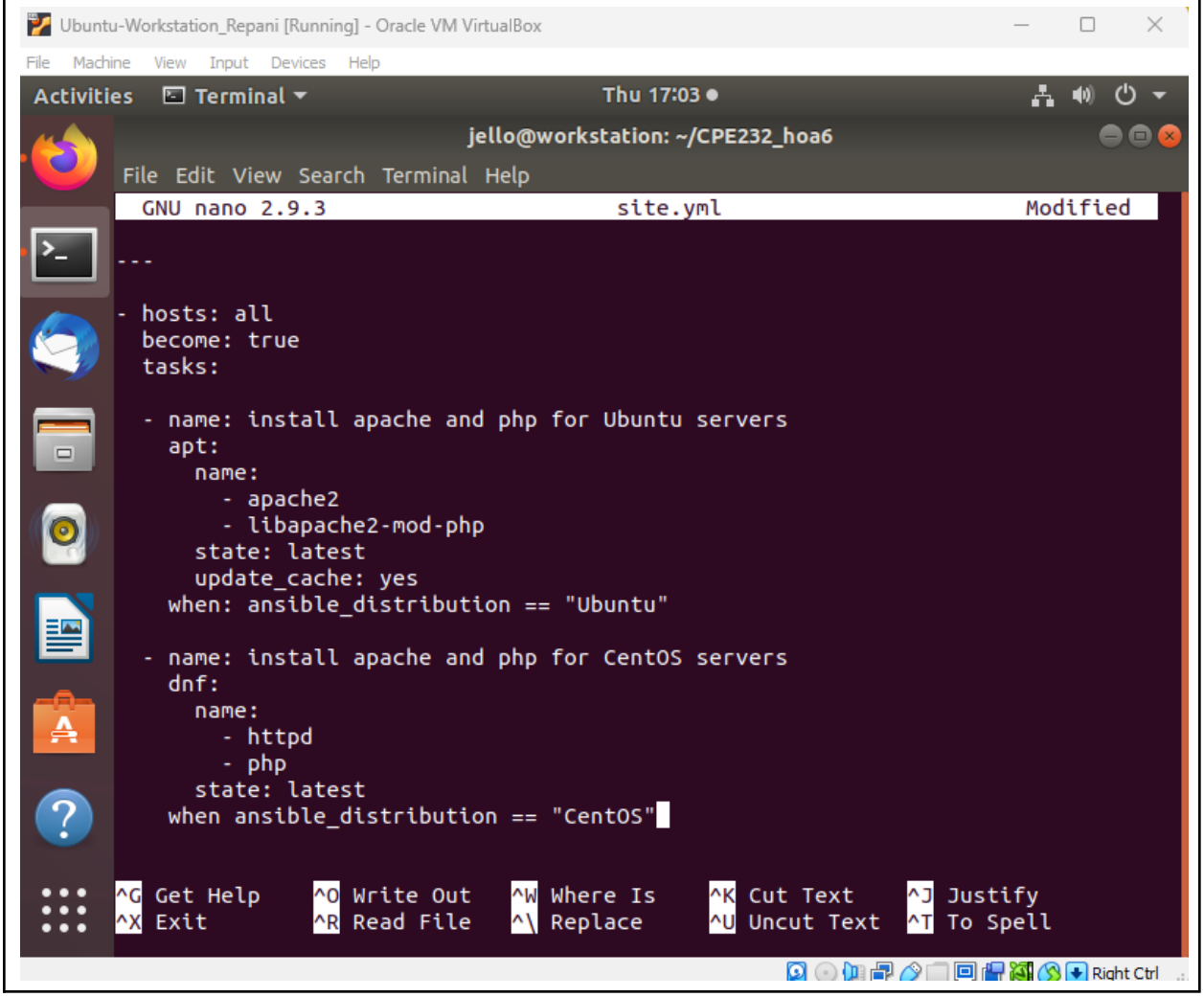
We also try to manage services that do not automatically run using the automations in playbook. For example, when we install web servers or httpd for CentOS, we notice that the service did not start automatically.

**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.

**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.

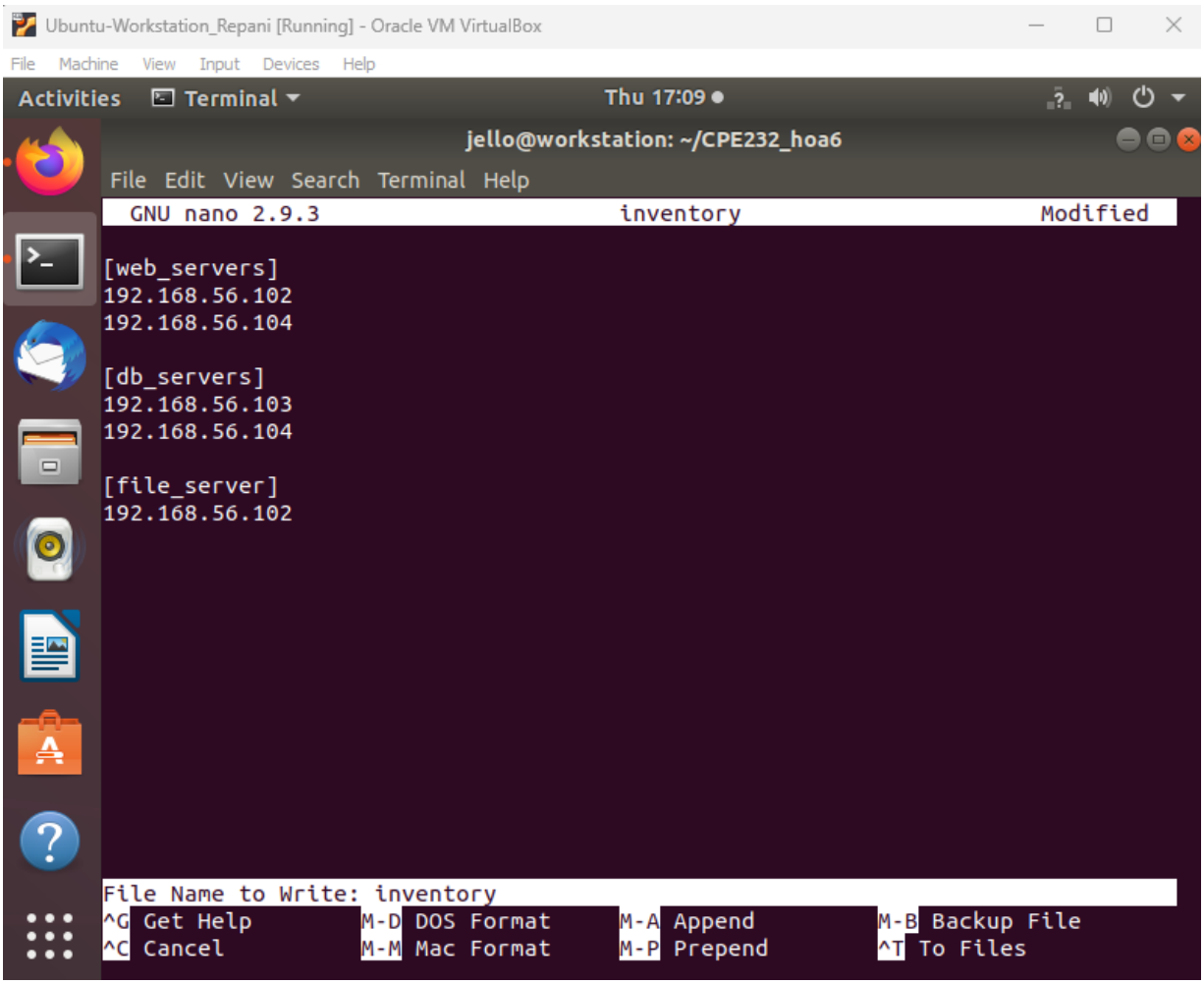
```
---

- 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"
```



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.



```
jello@workstation:~/CPE232_hoa6$ ansible web_servers -m ping
192.168.56.102 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
192.168.56.104 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
jello@workstation:~/CPE232_hoa6$ ansible db_servers -m ping
192.168.56.104 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.56.103 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
```

```
jello@workstation:~/CPE232_hoa6$ ansible file_server -m ping
192.168.56.102 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

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.

```
jello@workstation: ~/CPE232_hoa6
File  Edit  View  Search  Terminal  Help
  GNU nano 2.9.3                    site.yml                    Modified

---

- 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"
```

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.

- The playbook updated both the ubuntu and centOS and also successfully installed apache and php for all

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
changed: [192.168.56.102]
changed: [192.168.56.103]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *******************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]
```

```
TASK [install apache and php for CentOS servers] *******************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]

PLAY RECAP *********************************************************************************
192.168.56.102             : ok=4    changed=1    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0
   ignored=0
192.168.56.104             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
```

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.

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

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

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

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

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo
```

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

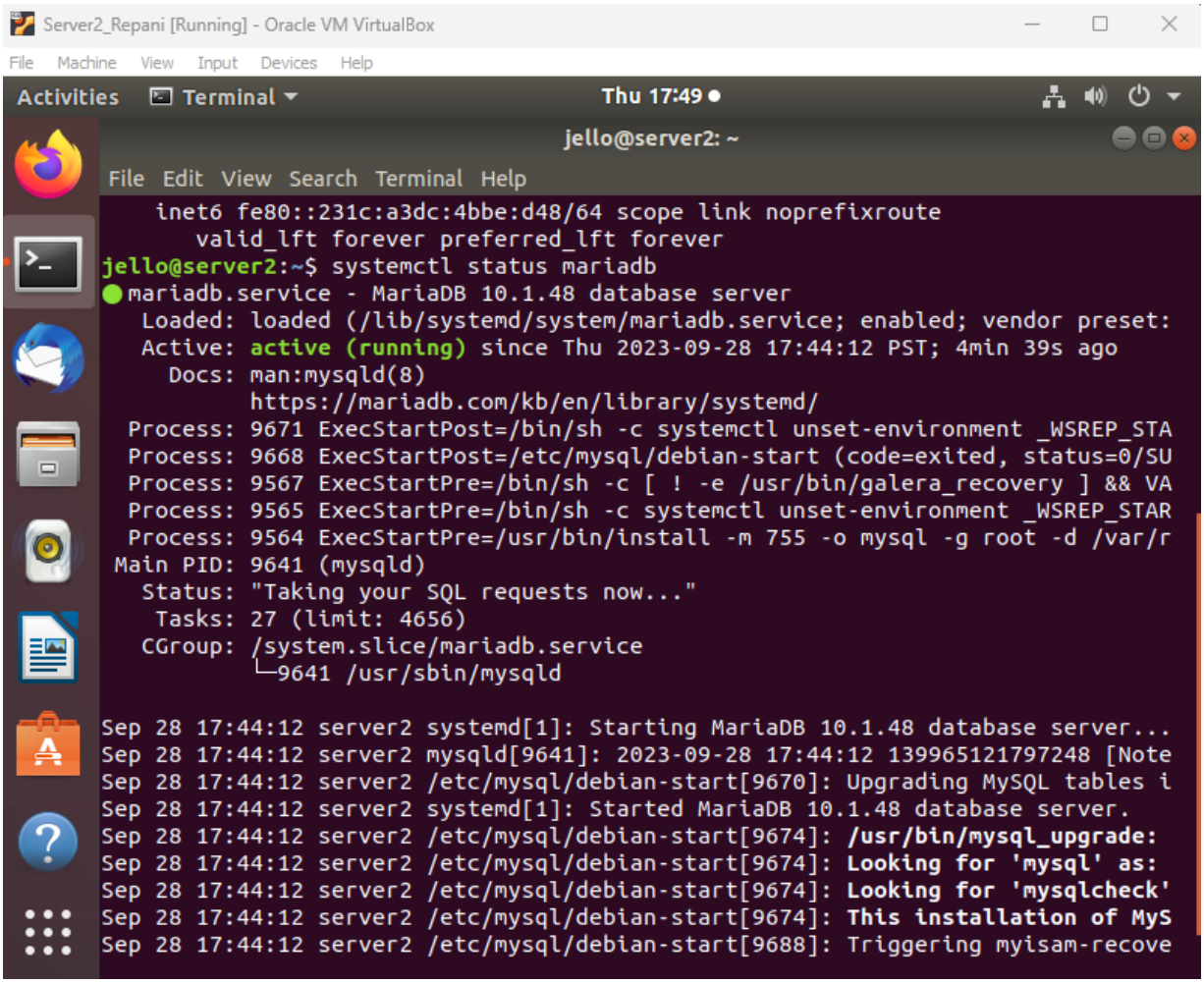
- The playbook successfully ran all the commands, as for the yml file the part in the ubuntu installation of mariadb and the running of services is

interchanged so that the playbook would work since the service was trying to run without having the mariadb installed first.

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *******************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *******************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (CentOS)] ****************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (Ubuntu)] ****************************************************
skipping: [192.168.56.104]
changed: [192.168.56.103]

TASK [Mariadb- Restarting/Enabling] ********************************************************
changed: [192.168.56.103]
changed: [192.168.56.104]

PLAY RECAP *********************************************************************************
192.168.56.102             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=5    changed=2    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=7    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
```

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.



```
CentOS_Repani [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Applications  Places  Terminal                                   Thu 17:48
jello@localhost:~
File  Edit  View  Search  Terminal  Help
 group default qlen 1000
    link/ether 52:54:00:2b:fa:a9 brd ff:ff:ff:ff:ff:ff
[jello@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 Thu 2023-09-28 17:44:18 PST; 4min 2s ago
  Process: 7579 ExecStartPost=/usr/libexec/mariadb-wait-ready $MAINPID (code=exited, st
atus=0/SUCCESS)
  Process: 7544 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir %n (code=exited, statu
s=0/SUCCESS)
 Main PID: 7578 (mysqld_safe)
    Tasks: 20
   CGroup: /system.slice/mariadb.service
           ├─7578 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
           └─7743 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --plug...

Sep 28 17:44:16 localhost.localdomain systemd[1]: Starting MariaDB database server...
Sep 28 17:44:16 localhost.localdomain mariadb-prepare-db-dir[7544]: Database MariaDB...
Sep 28 17:44:16 localhost.localdomain mariadb-prepare-db-dir[7544]: If this is not t...
Sep 28 17:44:16 localhost.localdomain mysqld_safe[7578]: 230928 17:44:16 mysqld_safe...
Sep 28 17:44:16 localhost.localdomain mysqld_safe[7578]: 230928 17:44:16 mysqld_safe...
Sep 28 17:44:18 localhost.localdomain systemd[1]: Started MariaDB database server.
Hint: Some lines were ellipsized, use -l to show in full.
[jello@localhost ~]$
```

Describe the output.

- Both Ubuntu server 2 and the CentOS server are now installed with mariadb and is running after the service command.

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: file_server
  become: true
  tasks:

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

File Name to Write: site.yml
^G Get Help        M-D DOS Format     M-A Append         M-B Backup File
^C Cancel          M-M Mac Format     M-P Prepend        ^T To Files
```

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

- The playbook was successfully run and samba was installed in the file server

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]
ok: [192.168.56.102]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]
ok: [192.168.56.103]

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *******************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *******************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

```
PLAY [file_server] ***********************************************************************************

TASK [Gathering Facts] *******************************************************************************
ok: [192.168.56.102]

TASK [install samba package] *************************************************************************
changed: [192.168.56.102]

PLAY RECAP *******************************************************************************************
192.168.56.102             : ok=6    changed=1    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=7    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
```

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"
```

```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.

jello@workstation: ~/CPE232_hoa6

File Edit View Search Terminal Help

GNU nano 2.9.3 site.yml Modified

```
---

- 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: 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"
```

```
  - name: install mariadb package (CentOS)
    tags: 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_server
  become: true
  tasks:

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

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

- The playbook was successfully

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *******************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *******************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (CentOS)] ****************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (Ubuntu)] ****************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]

TASK [Mariadb- Restarting/Enabling] ********************************************************
changed: [192.168.56.103]
changed: [192.168.56.104]

PLAY [file_server] *************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]

TASK [install samba package] ***************************************************************
ok: [192.168.56.102]
```

```
PLAY RECAP *********************************************************************************
192.168.56.102             : ok=6    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=7    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
```

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

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

The command shows all the tags added for each part of the commands

```
jello@workstation:~/CPE232_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_server): file_server    TAGS: []
      TASK TAGS: [samba]
```

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

Only the parts with the tags centos are run, otherwise it is skipped

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --tags centos --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for CentOS servers] *******************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]

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

```
TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (CentOS)] ************************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

PLAY [file_server] *******************************************************************************

TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.102]

PLAY RECAP ***************************************************************************************
192.168.56.102             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=3    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=6    changed=0    unreachable=0    failed=0    skipped=1    rescued=0
   ignored=0
```

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

Only the parts with the db tag are performed which resulted in faster time in running the playbook

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --tags db --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] ******************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ******************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

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

TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]
```

```
TASK [Gathering Facts] ***********************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (CentOS)] ******************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (Ubuntu)] ******************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]

PLAY [file_server] ***************************************************************************

TASK [Gathering Facts] ***********************************************************************
ok: [192.168.56.102]

PLAY RECAP ***********************************************************************************
192.168.56.102             : ok=4    changed=0    unreachable=0    failed=0    skipped=1    rescued=0
   ignored=0
192.168.56.103             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=5    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
```

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

Only the parts with the apache tag are performed which resulted in faster time in running the playbook

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --tags apache --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] ***********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] **************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] **************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] ***********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *********************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *********************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

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

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

PLAY [file_server] **********************************************************************************

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.102]

PLAY RECAP ******************************************************************************************
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=3    changed=0    unreachable=0    failed=0    skipped=1    rescued=0
   ignored=0
192.168.56.104             : ok=5    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
```

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

Only the parts with the tags apache and db are run which made the overall run time of the playbook shorter than running the entire playbook

```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --tags "apache,db" --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] *********************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] *********************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for Ubuntu servers] ****************************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] ****************************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

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

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (CentOS)] *************************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (Ubuntu)] *************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]

PLAY [file_server] **********************************************************************************

TASK [Gathering Facts] ******************************************************************************
ok: [192.168.56.102]

PLAY RECAP ******************************************************************************************
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.103             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=6    changed=0    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
```

**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: 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.

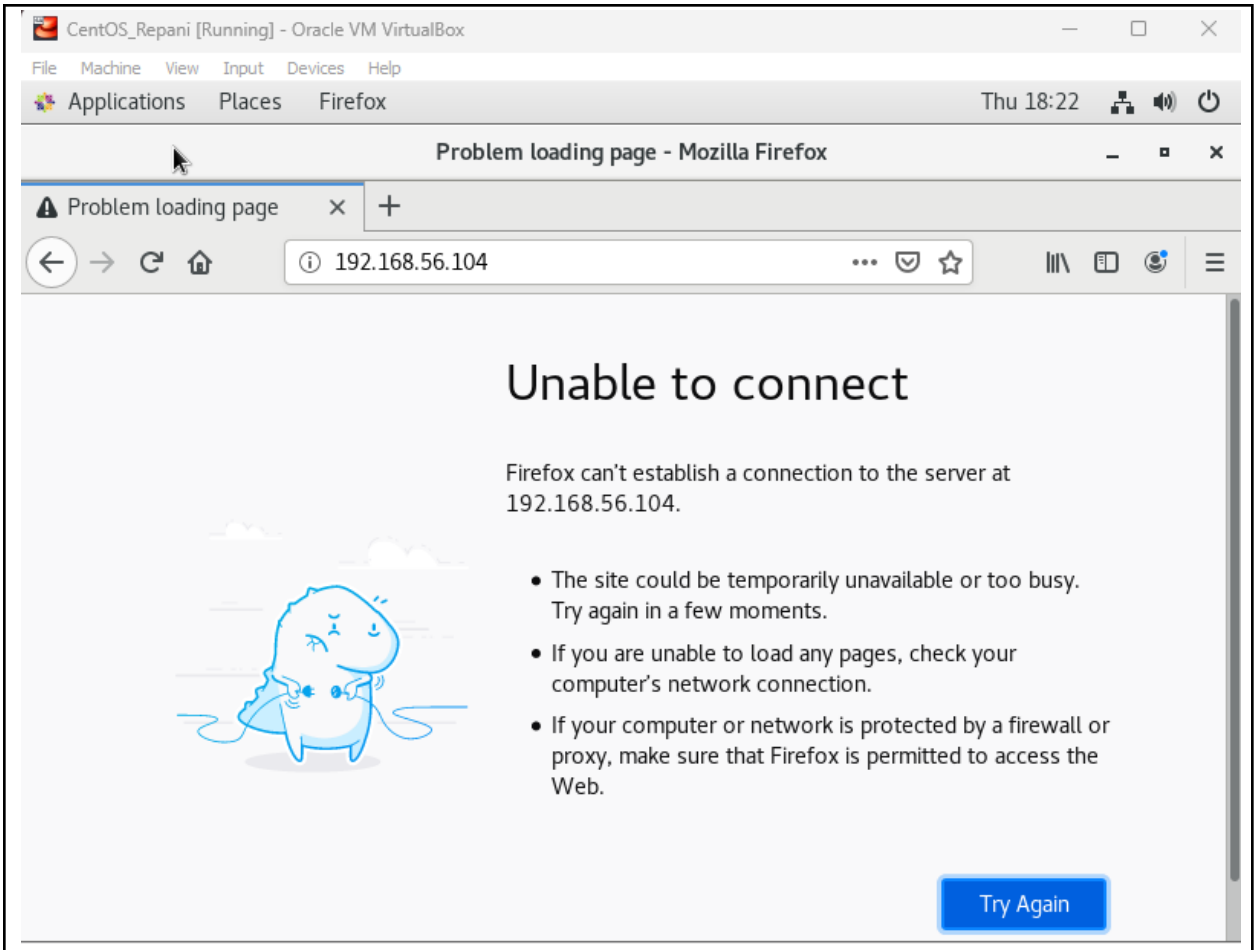
```
- hosts: db_servers
  become: true
  tasks:

  - 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.

```
[jello@localhost ~]$ sudo systemctl stop httpd
[sudo] password for jello:
```



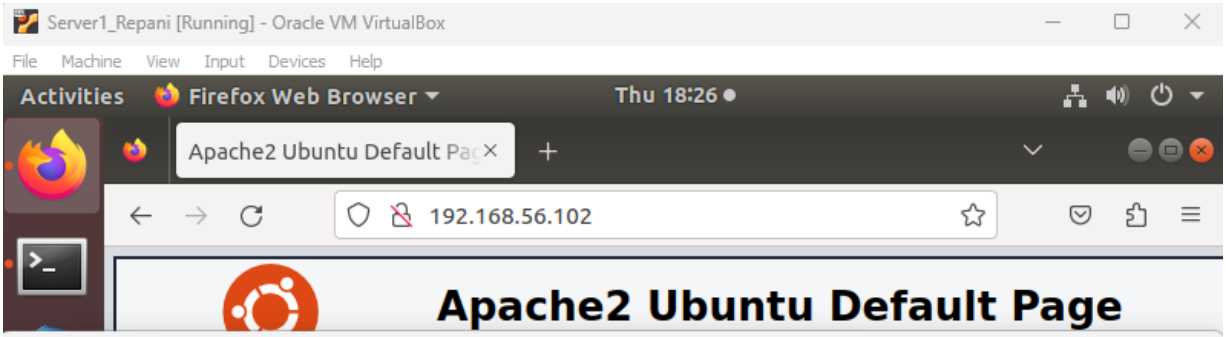

3. Go to the local machine and this time, run the *site.yml* file. Then after running the file, go again to the CentOS server and enter its IP address on the browser. Describe the result.
   - A

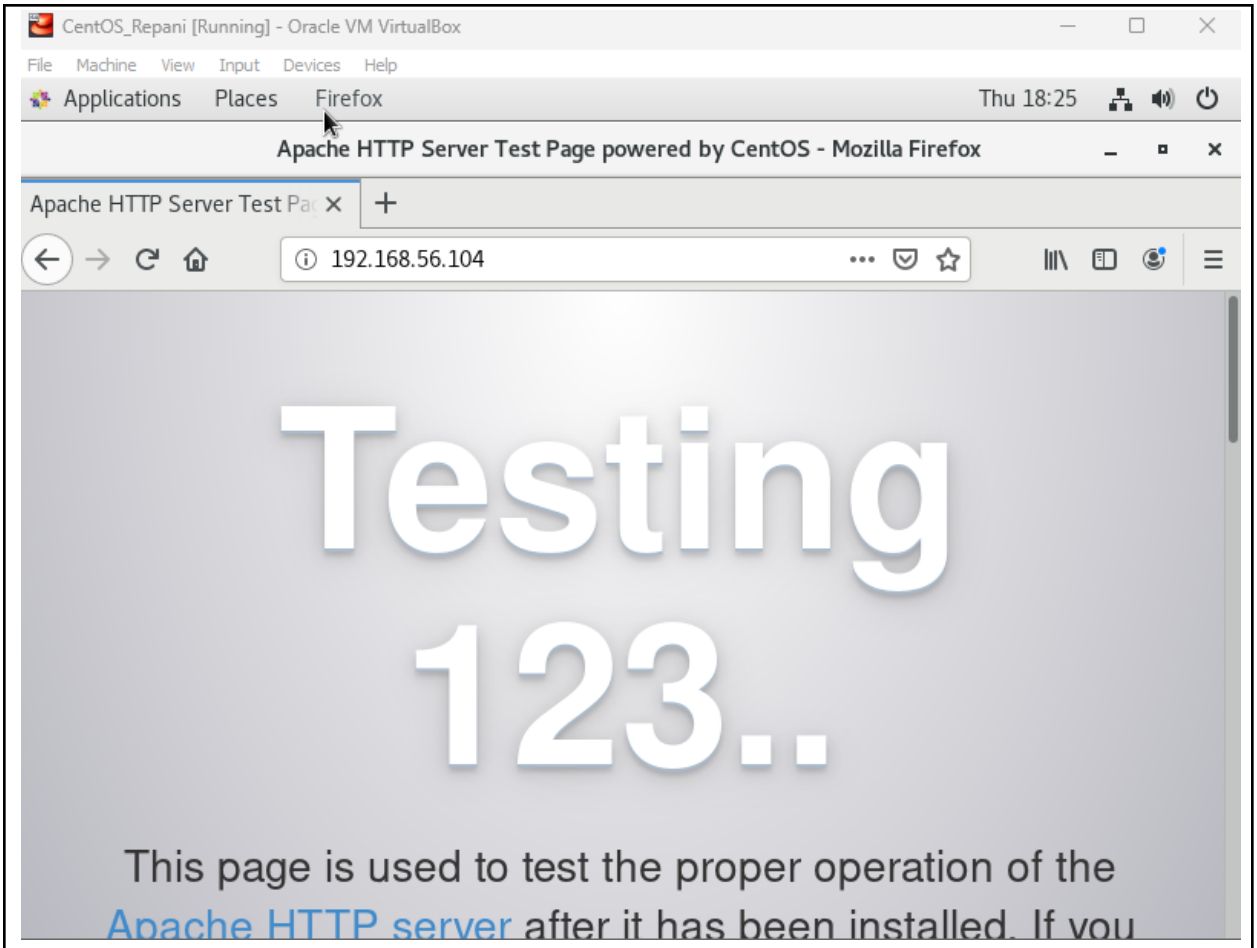
```
jello@workstation:~/CPE232_hoa6$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]
ok: [192.168.56.102]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [install apache and php for Ubuntu servers] *******************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *******************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]
```

```
TASK [start httpd (CentOS)] ****************************************************************
skipping: [192.168.56.102]
changed: [192.168.56.104]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [install mariadb package (CentOS)] ****************************************************
skipping: [192.168.56.103]
ok: [192.168.56.104]

TASK [install mariadb package (Ubuntu)] ****************************************************
skipping: [192.168.56.104]
ok: [192.168.56.103]

TASK [Mariadb- Restarting/Enabling] ********************************************************
changed: [192.168.56.103]
changed: [192.168.56.104]

PLAY [file_server] *************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]

TASK [install samba package] ***************************************************************
ok: [192.168.56.102]
```

```
PLAY RECAP *********************************************************************************************
192.168.56.102             : ok=6    changed=0    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0    skipped=2    rescued=0
   ignored=0
192.168.56.104             : ok=8    changed=2    unreachable=0    failed=0    skipped=3    rescued=0
   ignored=0
```

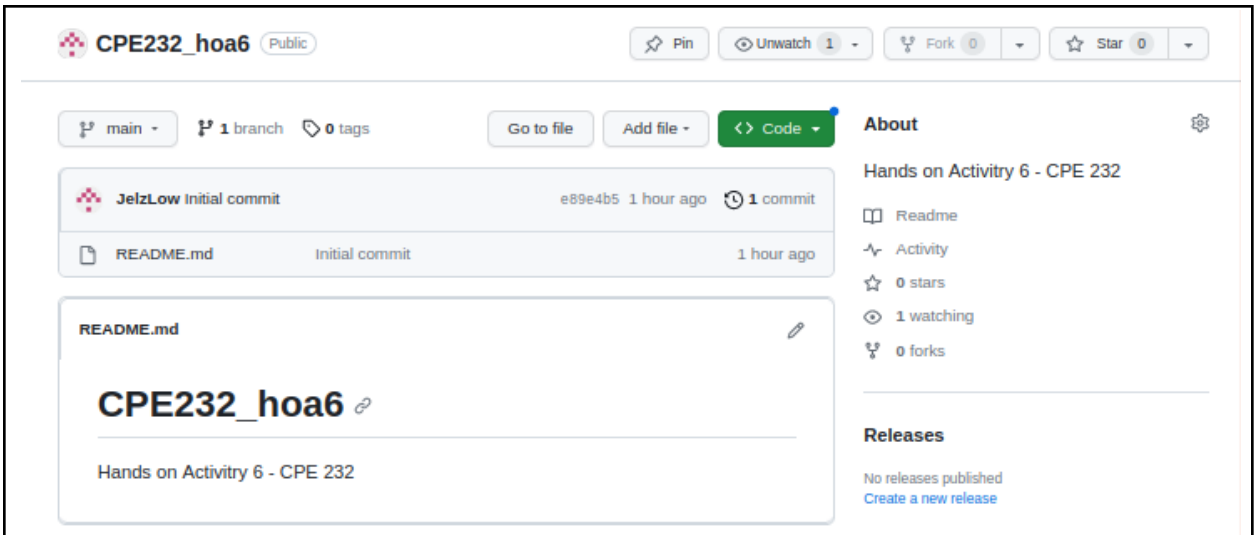
## Apache Proof



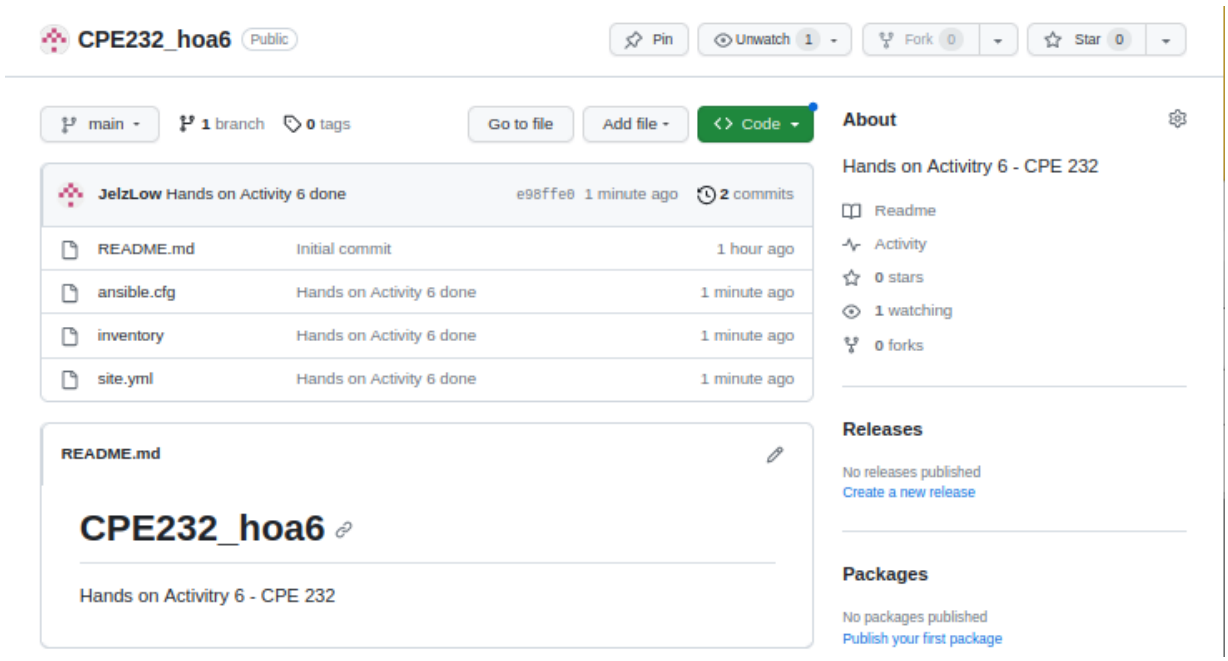




To automatically enable the service every time we run the playbook, use the command *enabled: true* similar to Figure 7.1.2 and save the playbook.

**Git repository**




```
jello@workstation:~/CPE232_hoa6$ git add *
jello@workstation:~/CPE232_hoa6$ git commit -m "Hands on Activity 6 done"
[main e98ffe0] Hands on Activity 6 done
 3 files changed, 105 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 inventory
 create mode 100644 site.yml
jello@workstation:~/CPE232_hoa6$ git push origin
Counting objects: 5, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 955 bytes | 955.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0)
To github.com:JelzLow/CPE232_hoa6.git
   e89e4b5..e98ffe0  main -> main
```




Git repository link:
[https://github.com/JelzLow/CPE232_hoa6](https://github.com/JelzLow/CPE232_hoa6)

**Reflections:**

Answer the following:

1. What is the importance of putting our remote servers into groups?
   - It is important to put remote servers into groups in order to easily classify the purpose and function of each server. This is very helpful when managing many different servers so that we can easily manage the different servers without getting confused.
2. What is the importance of tags in playbooks?
   - Tags in playbooks help in finding the appropriate command, this is helpful when the playbook is long and consists of many commands, by searching the tags we can filter and narrow down the commands that we will actually need. This also servers as a quick way to identify the purpose and function of a command rather than reading the entire name.
3. Why do think some services need to be managed automatically in playbooks?
   - Some services needs to automatically be managed in playbooks because there are a lot of tasks and programs which have to be manually performed and started usually once a system starts up. By adding commands that can automatically manage services in playbooks, it can increase the efficiency of the system because we no longer have to perform each and every single step, saving time.

**Conclusion**

In this hands-on activity 6, the topic is all about the playbook and additional commands and syntaxes used in it. For this one I specifically learned about the different user groups in the inventory file which allows the remote servers to be categorized accordingly. Additionally, the commands in ansible playbook such as tags are also introduced. Adding tags allows us to search for specific parts of the playbook and only run those specific commands with the tags attached to it.

**Honor Pledge**

*"I affirm that I have not given or received any unauthorized help on this assignment, and that this work is my own."*