

Name: Magboo, Matt Clemence C.	Date Performed: 08/27/2025
Course/Section: CPE212 - CPE32S2	Date Submitted: 08/27/2025
Instructor: Engr. Robin Valenzuela	Semester and SY: 2025 - 2026

### Activity 5: Consolidating Playbook plays

#### 1. Objectives:

- 1.1 Use **when** command in playbook for different OS distributions
- 1.2 Apply refactoring techniques in cleaning up the playbook codes

#### 2. Discussion:

We are going to look at a way that we can differentiate a playbook by a host in terms of which distribution the host is running. It's very common in most Linux shops to run multiple distributions, for example, Ubuntu shop or Debian shop and you need a different distribution for a one off-case or perhaps you want to run plays only on certain distributions.

It is a best practice in ansible when you are working in a collaborative environment to use the command git pull. git pull is a Git command used to update the local version of a repository from a remote. By default, git pull does two things. Updates the current local working branch (currently checked out branch) and updates the remote-tracking branches for all other branches. git pull essentially pulls down any changes that may have happened since the last time you worked on the repository.

#### Requirement:

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

#### Task 1: Use when command for different distributions

1. In the local machine, make sure you are in the local repository directory (**CPE232\_yourname**). Issue the command git pull. When prompted, enter the correct passphrase or password. Describe what happened when you issue this command. Did something happen? Why?

```
Magboo@LocalMachine:~/CPE232$ git pull
Already up to date.
```

2. Edit the inventory file and add the IP address of the Centos VM. Issue the command we used to execute the playbook (the one we used in the last activity): **ansible-playbook --ask-become-pass install\_apache.yml**. After executing this command, you may notice that it did not become successful in the Centos VM. You can see that the Centos VM has failed=1. Only the two remote

servers have been changed. The reason is that Centos VM does not support "apt" as the package manager. The default package manager for Centos is "yum."

```
Magboo@LocalMachine: ~/CPE232
Magboo@LocalMachine:~/CPE232$ ansible-playbook --ask-become-pass install_apache.
yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
Fatal: [192.168.56.102]: UNREACHABLE! => {"changed": false, "msg": "Failed to co
nnect to the host via ssh: no such identity: /home/Magboo/.ssh/ansible: No such
file or directory\r\nMagboo@192.168.56.102: Permission denied (publickey,password
).", "unreachable": true}
Fatal: [192.168.56.105]: UNREACHABLE! => {"changed": false, "msg": "Failed to co
nnect to the host via ssh: no such identity: /home/Magboo/.ssh/ansible: No such
file or directory\r\nMagboo@192.168.56.105: Permission denied (publickey,gssapi-
keyex,gssapi-with-mic,password).", "unreachable": true}
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [update repository index] *****
changed: [192.168.56.104]
changed: [192.168.56.103]

TASK [install apache2 package] *****
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [add PHP support for apache] *****
ok: [192.168.56.103]
ok: [192.168.56.104]

PLAY RECAP *****
192.168.56.102 : ok=0 changed=0 unreachable=1 failed=0 s
```

3. Edit the *install\_apache.yml* file and insert the lines shown below.

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

    - name: update repository index
      apt:
        update_cache: yes
      when: ansible_distribution == "Ubuntu"

    - name: install apache2 package
      apt:
        name: apache2
      when: ansible_distribution == "Ubuntu"

    - name: add PHP support for apache
      apt:
        name: libapache2-mod-php
      when: ansible_distribution == "Ubuntu"
```

CN\_Magboo [Running] - Oracle VirtualBox

File Machine View Input Devices Help

Magboo@LocalMachine: ~/CPE232

GNU nano 7.2 install\_apache.yml

```
---
- hosts: all
  become: true
  tasks:
    - name: update repository index
      apt:
        update_cache: yes
      when: ansible_distribution == "ubuntu"
    - name: install apache2 package
      apt:
        name: apache2
      when: ansible_distribution == "ubuntu"
    - name: add PHP support for apache
      apt:
        name: libapache2-mod-php
      when: ansible_distribution == "ubuntu"
```

Make sure to save the file and exit.

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```

Magboo@LocalMachine: ~/CPE212_Magboo
file or directory\r\nMagboo@192.168.56.111: Permission denied (publickey,password).", "unreachable": true}
ok: [192.168.56.110]
ok: [192.168.56.105]
ok: [192.168.56.109]

TASK [update repository index] *****
skipping: [192.168.56.105]
changed: [192.168.56.110]
changed: [192.168.56.109]

TASK [install apache2 package] *****
skipping: [192.168.56.105]
ok: [192.168.56.109]
ok: [192.168.56.110]

TASK [add PHP support for apache] *****
skipping: [192.168.56.105]
ok: [192.168.56.109]
changed: [192.168.56.110]

PLAY RECAP *****
192.168.56.105      : ok=1    changed=0    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.109      : ok=4    changed=1    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0
192.168.56.110      : ok=4    changed=2    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0
192.168.56.111      : ok=0    changed=0    unreachable=1    failed=0    s
kipped=0    rescued=0    ignored=0
Magboo@LocalMachine: ~/CPE212_Magboo$

```

If you have a mix of Debian and Ubuntu servers, you can change the configuration of your playbook like this.

- name: update repository index
  - apt:
    - update\_cache: yes
    - when: ansible\_distribution in ["Debian", "Ubuntu"]

*Note:* This will work also if you try. Notice the changes are highlighted.

4. Edit the *install\_apache.yml* file and insert the lines shown below.

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

    - name: update repository index
      apt:
        update_cache: yes
      when: ansible_distribution == "Ubuntu"

    - name: install apache2 package
      apt:
        name: apache2
        state: latest
      when: ansible_distribution == "Ubuntu"

    - name: add PHP support for apache
      apt:
        name: libapache2-mod-php
        state: latest
      when: ansible_distribution == "Ubuntu"

    - name: update repository index
      dnf:
        update_cache: yes
      when: ansible_distribution == "CentOS"

    - name: install apache2 package
      dnf:
        name: httpd
        state: latest
      when: ansible_distribution == "CentOS"

    - name: add PHP support for apache
      dnf:
        name: php
        state: latest
      when: ansible_distribution == "CentOS"
```



CN\_Magboo [Running] - Oracle VirtualBox

File Machine View Input Devices Help

GNU nano 7.2

i

```
- name: update repository index
  apt:
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: install apache2 package
  apt:
    name: apache2
  when: ansible_distribution == "Ubuntu"

- name: add PHP support for apache
  apt:
    name: libapache2-mod-php
  when: ansible_distribution == "Ubuntu"

- name: update repository index
  dnf:
    update_cache: yes
  when: ansible_distribution == "CentOS"

- name: install apache2 package
  dnf:
    name: httpd
    state: latest
  when: ansible_distribution == "CentOS"

- name: add PHP support for apache
  dnf:
    name: php
    state: latest
  when: ansible_distribution == "CentOS"
```

Make sure to save and exit.

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```
CN_Magboo [Running] - Oracle VirtualBox
File Machine View Input Devices Help

skipping: [192.168.56.105]
changed: [192.168.56.104]
changed: [192.168.56.103]

TASK [install apache2 package] *****
skipping: [192.168.56.105]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [add PHP support for apache] *****
skipping: [192.168.56.105]
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [update repository index] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
ok: [192.168.56.105]

TASK [install apache2 package] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
changed: [192.168.56.105]

TASK [add PHP support for apache] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
changed: [192.168.56.105]

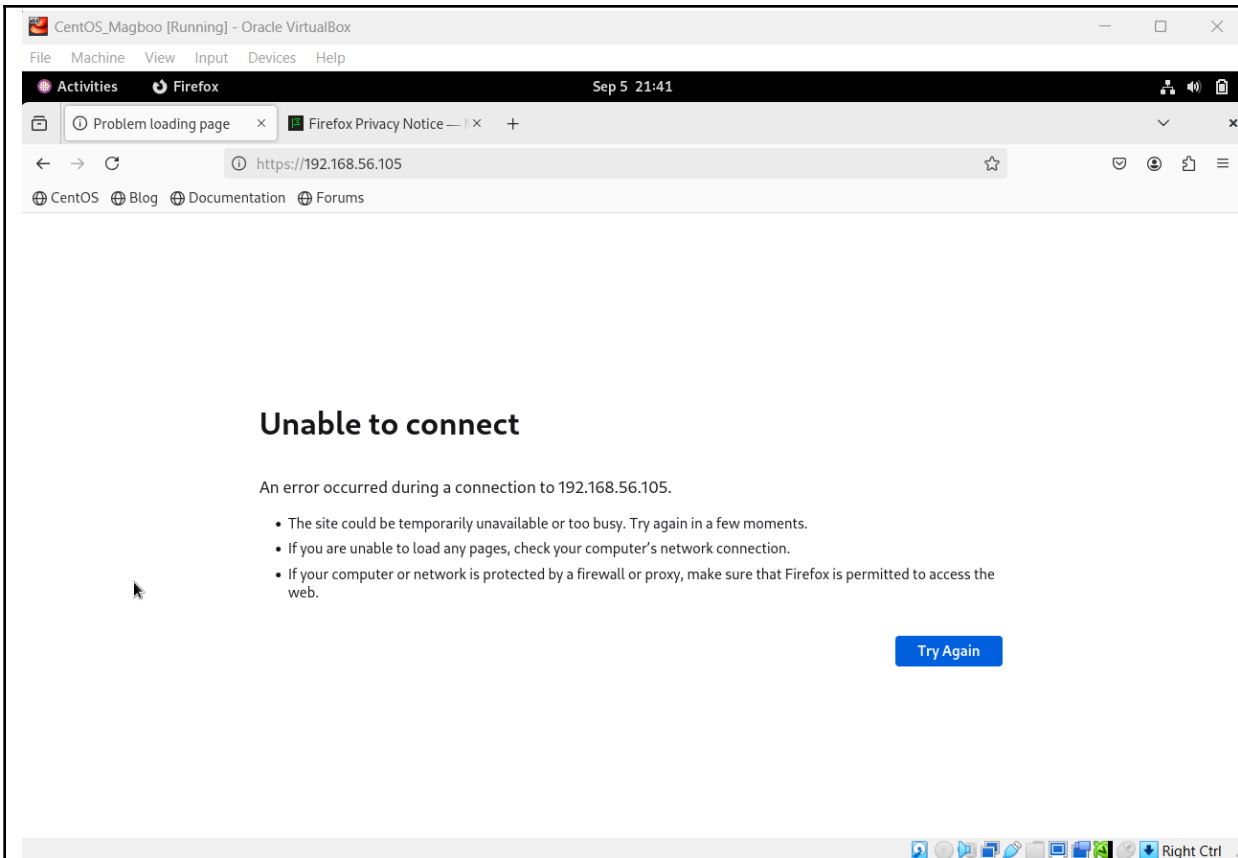
PLAY RECAP *****
192.168.56.102      : ok=0    changed=0    unreachable=1    failed=0    skipped=0    rescued=0    ignored=0
192.168.56.103      : ok=4    changed=1    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0
192.168.56.104      : ok=4    changed=1    unreachable=0    failed=0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

lagboo@LocalMachine:~/CPE232$
```

it skipped the unspecified distribution only the specified passed through the process

5. To verify the installations, go to CentOS VM and type its IP address on the browser. Was it successful? The answer is no. It's because the httpd service or the Apache HTTP server in the CentOS is not yet active. Thus, you need to activate it first.





5.1 To activate, go to the CentOS VM terminal and enter the following:

***systemctl status httpd***

The result of this command tells you that the service is inactive.

```
[magboo@vbox ~]$ systemctl status httpd
○ httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d>
   Drop-In: /usr/lib/systemd/system/httpd.service.d
            └─php-fpm.conf
   Active: inactive (dead)
   Docs: man:httpd.service(8)

lines 1-6/6 (END)
```

5.2 Issue the following command to start the service:

***sudo systemctl start httpd***

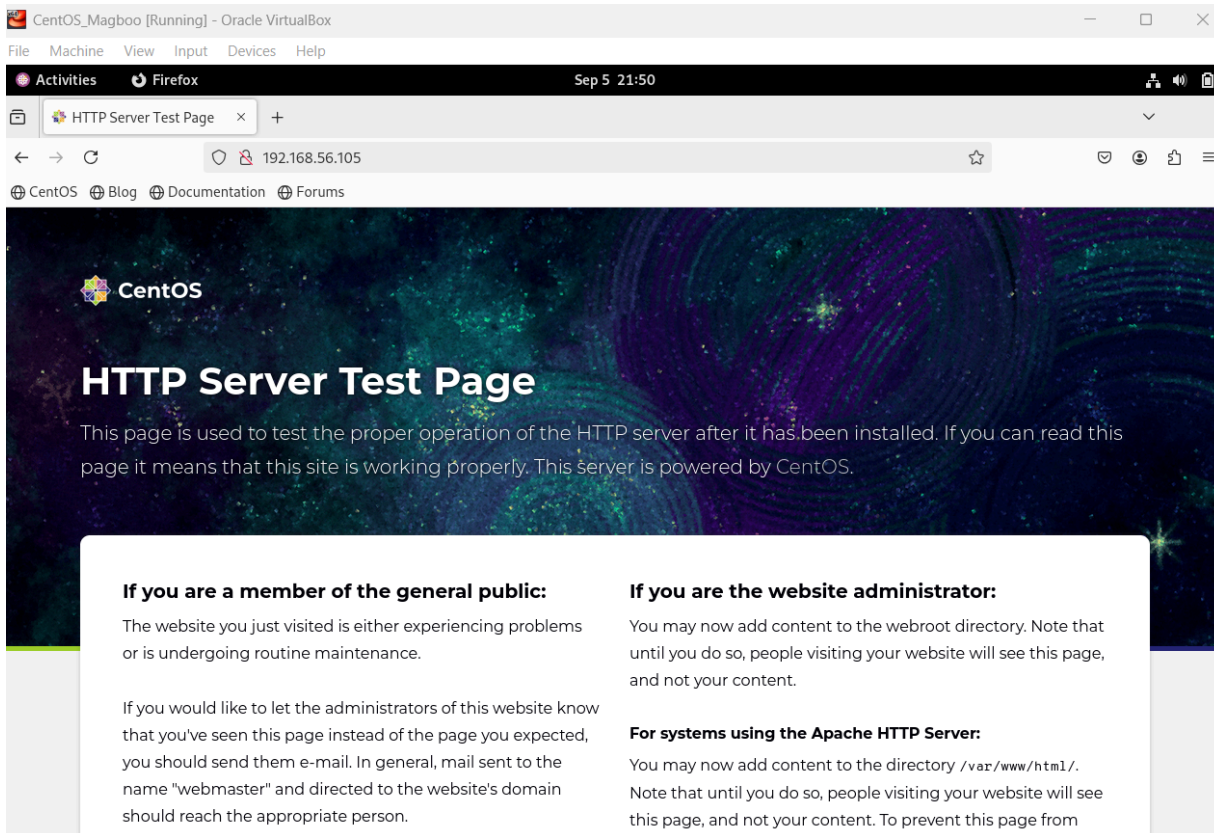
(When prompted, enter the sudo password)

***sudo firewall-cmd --add-port=80/tcp***

(The result should be a success)

```
[magboo@vbox ~]$ sudo systemctl start httpd
[sudo] password for magboo:
[magboo@vbox ~]$ sudo firewall-cmd --add-port=80/tcp
success
[magboo@vbox ~]$
```

5.3 To verify the service is already running, go to CentOS VM and type its IP address on the browser. Was it successful? (Screenshot the browser)



## Task 2: Refactoring playbook

This time, we want to make sure that our playbook is efficient and that the codes are easier to read. This will also makes run ansible more quickly if it has to execute fewer tasks to do the same thing.

1. Edit the playbook *install\_apache.yml*. Currently, we have three tasks targeting our Ubuntu machines and 3 tasks targeting our CentOS machine. Right now, we try to consolidate some tasks that are typically the same. For example, we can consolidate two plays that install packages. We can do that by creating a list of installation packages as shown below:

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

    - name: update repository index Ubuntu
      apt:
        update_cache: yes
        when: ansible_distribution == "Ubuntu"

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

    - name: update repository index for CentOS
      dnf:
        update_cache: yes
        when: ansible_distribution == "CentOS"

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

Make sure to save the file and exit.

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

- name: update repository index CentOS
  apt:
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

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

- name: update repository index for CentOS
  dnf:
    update_cache: yes
  when: ansible_distribution == "CentOS"

- name: install apache2 and php packages for CentOS
  dnf:
    name:
      - httpd
      - php
    state: latest
  when: ansible_distribution == "CentOS"
```

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```
CN_Magboo [Running] - Oracle VirtualBox
File Machine View Input Devices Help

TASK [Gathering Facts] *****
fatal: [192.168.56.102]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: no such identity: /
home/Magboo/.ssh/ansible: No such file or directory\r\nMagboo@192.168.56.102: Permission denied (publickey,password).", "unrea
chable": true}
ok: [192.168.56.105]
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [update repository index CentOS] *****
skipping: [192.168.56.105]
changed: [192.168.56.104]
changed: [192.168.56.103]

TASK [install apache2 and php packages for Ubuntu] *****
skipping: [192.168.56.105]
ok: [192.168.56.103]
ok: [192.168.56.104]

TASK [update repository index for CentOS] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
ok: [192.168.56.105]

TASK [install apache2 and php packages for CentOS] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
ok: [192.168.56.105]

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

Magboo@localMachine: ~/CDE232$
```

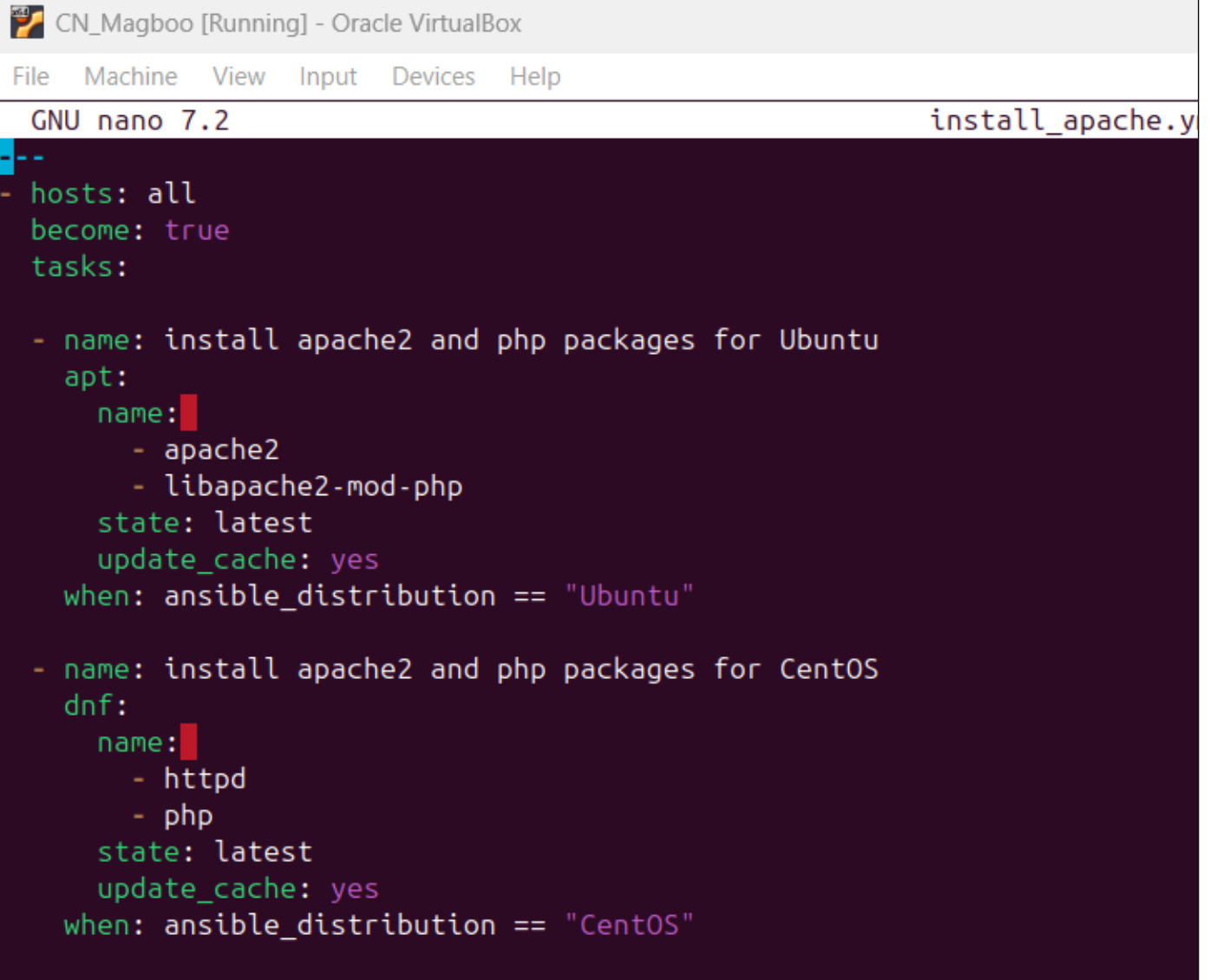
it works the same but the playbook became shorter because it joined 2 actions

2. Edit the playbook *install\_apache.yml* again. In task 2.1, we consolidated the plays into one play. This time we can actually consolidated everything in just 2 plays. This can be done by removing the update repository play and putting the command *update\_cache: yes* below the command *state: latest*. See below for reference:

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

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

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

The image shows a screenshot of an Oracle VM VirtualBox window titled 'CN\_Magboo [Running] - Oracle VirtualBox'. The window contains a terminal running the GNU nano 7.2 text editor, editing a file named 'install\_apache.y'. The terminal background is dark purple. The code in the editor is an Ansible playbook with the following structure:

```
--  
- hosts: all  
  become: true  
  tasks:  
    - name: install apache2 and php packages for Ubuntu  
      apt:  
        name:  
          - apache2  
          - libapache2-mod-php  
        state: latest  
        update_cache: yes  
        when: ansible_distribution == "Ubuntu"  
    - name: install apache2 and php packages for CentOS  
      dnf:  
        name:  
          - httpd  
          - php  
        state: latest  
        update_cache: yes  
        when: ansible_distribution == "CentOS"
```

Make sure to save the file and exit.

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```

Magboo@LocalMachine:~/CPE232$ ansible-playbook --ask-become-pass install_apache.yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
fatal: [192.168.56.102]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: no such identity: /
home/Magboo/.ssh/ansible: No such file or directory\r\nMagboo@192.168.56.102: Permission denied (publickey,password).", "unrea
chable": true}
ok: [192.168.56.103]
ok: [192.168.56.105]
ok: [192.168.56.104]

TASK [install apache2 and php packages for Ubuntu] *****
skipping: [192.168.56.105]
ok: [192.168.56.104]
ok: [192.168.56.103]

TASK [install apache2 and php packages for CentOS] *****
skipping: [192.168.56.103]
skipping: [192.168.56.104]
ok: [192.168.56.105]

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

```

the commands are compressed but it still proceed

- Finally, we can consolidate these 2 plays in just 1 play. This can be done by declaring variables that will represent the packages that we want to install. Basically, the `apache_package` and `php_package` are variables. The names are arbitrary, which means we can choose different names. We also take out the line `when: ansible_distribution`. Edit the playbook *install\_apache.yml* again and make sure to follow the below image. Make sure to save the file and exit.

```

- hosts: all
  become: true
  tasks:

  - name: install apache and php
    apt:
      name:
        - "{{ apache_package }}"
        - "{{ php_package }}"
      state: latest
      update_cache: yes

```





CN\_Magboo [Running] - Oracle VirtualBox

File Machine View Input Devices Help

GNU nano 7.2

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

  - name: install apache and php
    apt:
      name:
        - "{{ apache_package }}"
        - "{{ php_package }}"
      state: latest
      update_cache: yes
```

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```

Magboo@LocalMachine:~/CPE232$ ansible-playbook --ask-become-pass install_apache.yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
fatal: [192.168.56.102]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: no such identity: /home/Magboo/.ssh/ansible: No such file or directory\r\nMagboo@192.168.56.102: Permission denied (publickey,password).", "unreachable": true}
ok: [192.168.56.104]
ok: [192.168.56.105]
ok: [192.168.56.103]

TASK [install apache and php] *****
fatal: [192.168.56.103]: FAILED! => {"msg": "The task includes an option with an undefined variable. The error was: 'apache_package' is undefined. 'apache_package' is undefined\n\nThe error appears to be in '/home/Magboo/CPE232/install_apache.yml': line 6, column 5, but may\nbe elsewhere in the file depending on the exact syntax problem.\n\nThe offending line appears to be:\n\n\n - name: install apache and php\n    ^ here\n"}
fatal: [192.168.56.104]: FAILED! => {"msg": "The task includes an option with an undefined variable. The error was: 'apache_package' is undefined. 'apache_package' is undefined\n\nThe error appears to be in '/home/Magboo/CPE232/install_apache.yml': line 6, column 5, but may\nbe elsewhere in the file depending on the exact syntax problem.\n\nThe offending line appears to be:\n\n\n - name: install apache and php\n    ^ here\n"}
fatal: [192.168.56.105]: FAILED! => {"msg": "The task includes an option with an undefined variable. The error was: 'apache_package' is undefined. 'apache_package' is undefined\n\nThe error appears to be in '/home/Magboo/CPE232/install_apache.yml': line 6, column 5, but may\nbe elsewhere in the file depending on the exact syntax problem.\n\nThe offending line appears to be:\n\n\n - name: install apache and php\n    ^ here\n"}

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

```

## it failed on all the play

- Unfortunately, task 2.3 was not successful. It's because we need to change something in the inventory file so that the variables we declared will be in place. Edit the *inventory* file and follow the below configuration:

```

192.168.56.120 apache_package=apache2 php_package=libapache2-mod-php
192.168.56.121 apache_package=apache2 php_package=libapache2-mod-php
192.168.56.122 apache_package=httpd php_package=php

```

Make sure to save the *inventory* file and exit.

```
GNU nano 7.2 inventory.yaml *
[dbserver]
192.168.56.102
[workstation]
192.168.56.102
192.168.56.103
192.168.56.104
192.168.56.105
[webserver]
192.168.56.103 apache_package=apache2 php_package=libapache2-mod-php
192.168.56.104 apache_package=apache2 php_package=libapache2-mod-php
192.168.56.105 apache_package=httpd php_package=php
[CentOS]
192.168.56.105 ansible_user=magboo ansible_ssh_private_key_file=~/.ssh/id_rsa
```

**Finally**, we still have one more thing to change in our *install\_apache.yml* file. In task 2.3, you may notice that the package is assign as *apt*, which will not run in CentOS. Replace the *apt* with *package*. Package is a module in ansible that is generic, which is going to use whatever package manager the underlying host or the target server uses. For Ubuntu it will automatically use *apt*, and for CentOS it will automatically use *dnf*. Make sure to save the file and exit. For more details about the ansible package, you may refer to this documentation: [ansible.builtin.package — Generic OS package manager — Ansible Documentation](#)

CN\_Magboo [Running] - Oracle VirtualBox

File Machine View Input Devices Help

GNU nano 7.2

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

    - name: install apache and php
      package:
        name:
          - "{{ apache_package }}"
          - "{{ php_package }}"
        state: latest
        update_cache: yes
```

Run *ansible-playbook --ask-become-pass install\_apache.yml* and describe the result.

```
Magboo@LocalMachine:~/CPE232$ ansible-playbook --ask-become-pass install_apache.yml
BECOME password:
```

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

```
TASK [Gathering Facts] *****
fatal: [192.168.56.102]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: no such identity: /home/Magboo/.ssh/ansible: No such file or directory\r\nMagboo@192.168.56.102: Permission denied (publickey,password).", "unreachable": true}
```

```
ok: [192.168.56.103]
ok: [192.168.56.105]
ok: [192.168.56.104]
```

```
TASK [install apache and php] *****
```

```
ok: [192.168.56.105]
ok: [192.168.56.103]
ok: [192.168.56.104]
```

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

192.168.56.102	: ok=0	changed=0	unreachable=1	failed=0	skipped=0	rescued=0	ignored=0
192.168.56.103	: ok=2	changed=0	unreachable=0	failed=0	skipped=0	rescued=0	ignored=0
192.168.56.104	: ok=2	changed=0	unreachable=0	failed=0	skipped=0	rescued=0	ignored=0
192.168.56.105	: ok=2	changed=0	unreachable=0	failed=0	skipped=0	rescued=0	ignored=0

**in one play it accommodated 2 installation and update from 2 different packages or 2 different distributions**

**Supplementary Activity:**

1. Create a playbook that could do the previous tasks in Red Hat OS. \

```
Magboo@LocalMachine:~/CPE232$ ansible-playbook --ask-become-pass activity5sup.yaml
BECOME password:

PLAY [Install Apache and PHP on CentOS] *****

TASK [Gathering Facts] *****
ok: [192.168.56.105]

TASK [Install Apache and PHP packages] *****
ok: [192.168.56.105]

TASK [Start and enable Apache] *****
ok: [192.168.56.105]

TASK [Open HTTP port in firewall] *****
ok: [192.168.56.105]

PLAY RECAP *****
192.168.56.105      : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



CN\_Magboo [Running] - Oracle VirtualBox

File Machine View Input Devices Help

GNU nano 7.2

activity5sup.yaml

```
---
- name: Install Apache and PHP on CentOS
  hosts: CentOS
  become: yes
  tasks:
    - name: Install Apache and PHP packages
      package:
        name:
          - httpd
          - php
        state: latest

    - name: Start and enable Apache
      service:
        name: "{{ apache_package }}"
        state: started
        enabled: true

    - name: Open HTTP port in firewall
      firewalld:
        port: 80/tcp
        permanent: yes
        state: enabled
        immediate: yes
```

**Reflections:**

Answer the following:

1. Why do you think refactoring of playbook codes is important?

**it makes the code short and clean and doubles as it makes the playbook faster**

2. When do we use the “when” command in playbook?

**it is used to run task only if the task is true**