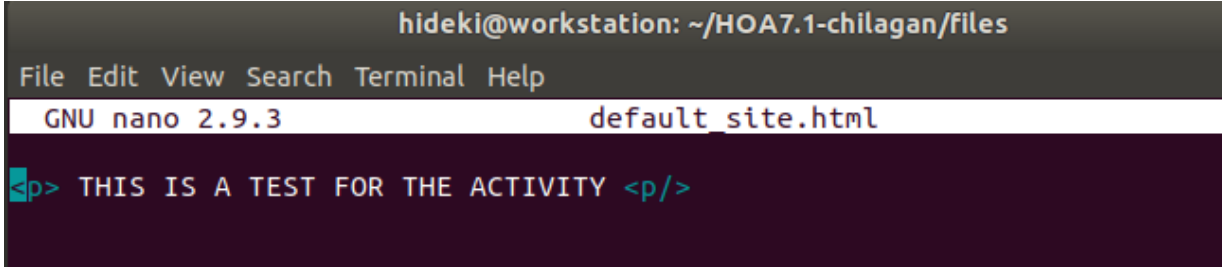


Name: Ilagan, Carlo Hideki D.	Date Performed: 10-7-2024
Course/Section: CPE31S2	Date Submitted: 10-10-2024
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st sem/2024-2025
<b>Activity 7: Managing Files and Creating Roles in Ansible</b>	
<b>1. Objectives:</b> 1.1 Manage files in remote servers 1.2 Implement roles in ansible	
<b>2. Discussion:</b> <p>In this activity, we look at the concept of copying a file to a server. We are going to create a file into our git repository and use Ansible to grab that file and put it into a particular place so that we could do things like customize a default website, or maybe install a default configuration file. We will also implement roles to consolidate plays.</p>	
<b>Task 1: Create a file and copy it to remote servers</b>  1. Using the previous directory we created, create a directory, and named it " <b>files</b> ." Create a file inside that directory and name it " <b>default_site.html</b> ." Edit the file and put basic HTML syntax. Any content will do, as long as it will display text later. Save the file and exit.	
	
2. Edit the <b>site.yml</b> file and just below the <b>web_servers</b> play, create a new file to copy the default html file for site: <ul style="list-style-type: none"> <li>- name: copy default html file for site</li> </ul> tags: apache, apache2, httpd copy: <ul style="list-style-type: none"> <li>src: default_site.html</li> <li>dest: /var/www/html/index.html</li> <li>owner: root</li> <li>group: root</li> <li>mode: 0644</li> </ul>	

```
hideki@workstation: ~/HOA7.1-chilagan
File Edit View Search Terminal Help
GNU nano 2.9.3 site.yml

when: ansible_distribution == "CentOS"

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

- name: copy default html file for site
  tags: apache, apache2, httpd
  copy:
    src: default_site.html
    dest: /var/www/html/index.html
    owner: root
    group: root
    mode: 0644

- hosts: db_servers
```

3. Run the playbook *site.yml*. Describe the changes.

```
hideki@workstation: ~/HOA7.1-chilagan
File Edit View Search Terminal Help
skipping: [192.168.56.111]

TASK [install apache and php for CENTOS servers] *****
ok: [192.168.56.111]

TASK [start httpd (CentOS)] *****
changed: [192.168.56.111]

TASK [copy default html file for site] *****
changed: [192.168.56.111]

PLAY [db_servers] *****

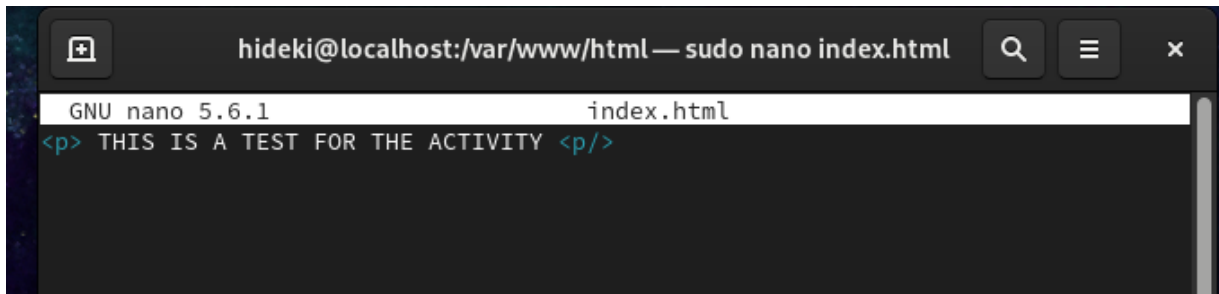
TASK [Gathering Facts] *****
ok: [192.168.56.111]

TASK [install mariadb package in CENTOS] *****
ok: [192.168.56.111]

TASK [install mariadb package in Ubuntu] *****
skipping: [192.168.56.111]

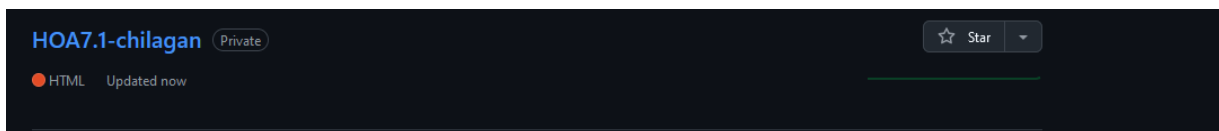
TASK [Mariadb- Restart/Enable] *****
changed: [192.168.56.111]
```

- There were added files for the servers inside the group `web_servers`.
4. Go to the remote servers (*web\_servers*) listed in your inventory. Use `cat` command to check if the `index.html` is the same as the local repository file (*default\_site.html*). Do both for Ubuntu and CentOS servers. On the CentOS server, go to the browser and type its IP address. Describe the output.



```
hideki@localhost:/var/www/html — sudo nano index.html
GNU nano 5.6.1 index.html
<p> THIS IS A TEST FOR THE ACTIVITY <p/>
```

- The text inside the `default_site.html` was copied to the remote server file named `index.html` since this is where the text was directed to.
5. Sync your local repository with GitHub and describe the changes.



- In checking repositories, this will now provide information that HOA 7.1 includes an HTML file inside the repository.

## Task 2: Download a file and extract it to a remote server

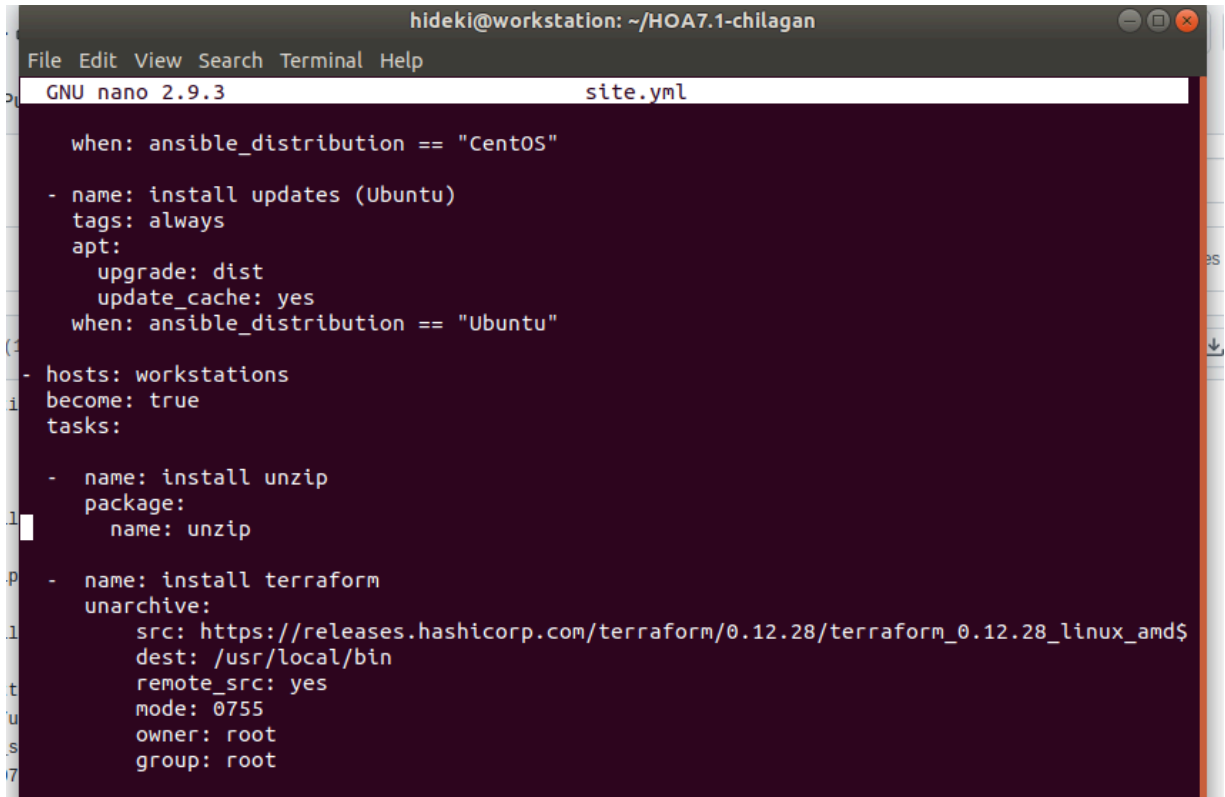
1. Edit the `site.yml`. Just before the `web_servers` play, create a new play:

- `hosts: workstations`  
`become: true`  
`tasks:`
  - `name: install unzip`  
`package:`
    - `name: unzip`
  - `name: install terraform`  
`unarchive:`

src:

[https://releases.hashicorp.com/terraform/0.12.28/terraform\\_0.12.28\\_linux\\_a  
md64.zip](https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_linux_amd64.zip)

```
dest: /usr/local/bin
remote_src: yes
mode: 0755
owner: root
group: root
```



```
hideki@workstation: ~/HOA7.1-chilagan
File Edit View Search Terminal Help
GNU nano 2.9.3 site.yml

  when: ansible_distribution == "CentOS"

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

- hosts: workstations
  become: true
  tasks:

    - name: install unzip
      package:
        name: unzip

    - name: install terraform
      unarchive:
        src: https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_linux_and$
        dest: /usr/local/bin
        remote_src: yes
        mode: 0755
        owner: root
        group: root
```

2. Edit the inventory file and add workstations group. Add any Ubuntu remote server. Make sure to remember the IP address.

```
hideki@workstation: ~/
File Edit View Search Terminal Help
GNU nano 2.9.3 inven

[db_servers]
192.168.56.111
[web_servers]
192.168.56.106
192.168.56.111
[file_servers]
192.168.56.107
[workstations]
192.168.56.106
```

3. Run the playbook. Describe the output.

```
hideki@workstation: ~/HOA7.1-chilagan
File Edit View Search Terminal Help

TASK [install updates (CentOS)] *****
skipping: [192.168.56.107]
skipping: [192.168.56.106]
ok: [192.168.56.111]

TASK [install updates (Ubuntu)] *****
skipping: [192.168.56.111]
ok: [192.168.56.107]
ok: [192.168.56.106]

PLAY [workstations] *****

TASK [Gathering Facts] *****
ok: [192.168.56.106]

TASK [install unzip] *****
ok: [192.168.56.106]

TASK [install terraform] *****
changed: [192.168.56.106]
```

- There were changes after install updates for the ubuntu which was the installation of terraform to the ubuntu remote server.
4. On the Ubuntu remote workstation, type terraform to verify installation of terraform. Describe the output.

```
hideki@server1:~$ terraform version
Terraform v0.12.28
```

- Terraform was installed to the ubuntu server

### **Task 3: Create roles**

1. Edit the site.yml. Configure roles as follows: (make sure to create a copy of the old site.yml file because you will be copying the specific plays for all groups)

```

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

    - name: update repository index (CentOS)
      tags: always
      dnf:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "CentOS"
    - name: install updates (Ubuntu)
      tags: always
      apt:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "Ubuntu"

- hosts: all
  become: true
  roles:
    - base

- hosts: workstations
  become: true
  roles:
    - workstations

- hosts: web_servers
  become: true
  roles:
    - web_servers

- hosts: db_servers
  become: true
  roles:
    - db_servers

- hosts: file_servers
  become: true
  roles:
    - file_servers

```

Save the file and exit.

2. Under the same directory, create a new directory and name it roles. Enter the roles directory and create new directories: base, web\_servers, file\_servers,

db\_servers and workstations. For each directory, create a directory and name it tasks.

```
hideki@workstation:~/H0A7.1-chilagan$ tree roles
roles
├── base
│   └── tasks
├── db_servers
│   └── tasks
├── file_servers
│   └── tasks
├── web_servers
│   └── tasks
└── workstations
    └── tasks
```

3. Go to tasks for all directory and create a file. Name it main.yml. In each of the tasks for all directories, copy and paste the code from the old site.yml file. Show all contents of main.yml files for all tasks.

```
hideki@workstation:~/H0A7.1-chilagan$ cat roles/base/tasks/main.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: workstations
  become: true
  tasks:

  - name: install unzip
    package:
      name: unzip
```



```
hideki@workstation:~/HOA7.1-chilagan$ tree roles
```

```
roles
├── base
│   └── tasks
│       └── main.yml
├── db_servers
│   └── tasks
│       └── main.yml
├── file_servers
│   └── tasks
│       └── main.yml
├── web_servers
│   └── tasks
│       └── main.yml
└── workstations
    └── tasks
        └── main.yml
```

```
10 directories, 5 files
```

main.yml for base:

```
hideki@workstation: ~/HOA7.1-chilagan/roles/base/tasks
```

```
File Edit View Search Terminal Help
```

```
GNU nano 2.9.3 main.yml
```

```
--
```

```
- 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"
```

```
[ Read 16 lines ]
```

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

main.yml for db\_servers:

```
hideki@workstation: ~/HOA7.1-chilagan/roles/db_servers/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml
---
- name: install mariadb package in CENTOS
  tags: centos, db, mariadb
  yum:
    name: mariadb-server
    state: latest
  when: ansible_distribution == "CentOS"

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

- name: "Mariadb- Restart/Enable"
  service:
    name: mariadb

[ Read 22 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

main.yml for file\_servers:

```
hideki@workstation: ~/HOA7.1-chilagan/roles/file_servers/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml
---
- name: install samba package
  tags: samba
  package:
    name: samba
    state: latest
```

### main.yml for web\_servers:

```
hideki@workstation: ~/HOA7.1-chilagan/roles/web_servers/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml
---
- name: install apache and ph for Ubuntu servers
  tags: apache, apache2, ubuntu
  apt:
    name:
      - apache2
      - libapache2-mod-php
    state: latest
    update_cache: yes
  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: start httpd (CentOS)
  tags: apache, centos, httpd
  service:
    name: httpd
    state: started

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

### main.yml for workstations:

```
hideki@workstation: ~/HOA7.1-chilagan/roles/workstations/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml
---
- name: install unzip
  package:
    name: unzip

- name: install terraform
  unarchive:
    src: https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_linux_and64.zip
    dest: /usr/local/bin
    remote_src: yes
    mode: 0755
    owner: root
    group: root
```

4. Run the site.yml playbook and describe the output.

```
hideki@workstation: ~/HOA7.1-chilagan
File Edit View Search Terminal Help
skipping: [192.168.56.106]
ok: [192.168.56.111]

TASK [web_servers : start httpd (CentOS)] *****
skipping: [192.168.56.106]
ok: [192.168.56.111]

TASK [web_servers : copy default html file for site] *****
ok: [192.168.56.106]
ok: [192.168.56.111]

PLAY [db_servers] *****

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

TASK [db_servers : install mariadb package in CentOS] *****
ok: [192.168.56.111]

TASK [db_servers : install mariadb package in Ubuntu] *****
skipping: [192.168.56.111]

TASK [db_servers : Mariadb- Restart/Enable] *****
changed: [192.168.56.111]

PLAY [file_servers] *****

TASK [Gathering Facts] *****
ok: [192.168.56.107]

TASK [file_servers : install samba package] *****
ok: [192.168.56.107]

PLAY RECAP *****
192.168.56.106      : ok=10   changed=0    unreachable=0    failed=0    skipped=4    rescued=0    ignored=0
192.168.56.107      : ok=6    changed=0    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
192.168.56.111      : ok=11   changed=1    unreachable=0    failed=0    skipped=4    rescued=0    ignored=0

hideki@workstation:~/HOA7.1-chilagan$
```

- Each host ran its specific tasks included.

## Reflections:

In this activity, we were able to utilize and learn how to copy content of files to a server's file in which we copied an html content from control node to the managed nodes without accessing the server directly. We were also able to run different tasks for different servers using roles. Not only is this type of process resistant to errors, but it is also a time efficient way to do commands.

Answer the following:

1. What is the importance of creating roles?

Creating roles will provide an efficiency for the administrator in managing its task per server. For example, if the role of server 1 and 2 is to have a mariadb, you can add a command or inventory to install mariadb servers in this 2 server by using roles.

2. What is the importance of managing files?

- Managing files can also provide an efficiency for the administrator since in this type of process, files from different servers can be easily manipulated, modified, and saved or deleted using the control node.