Name: De Omampo, Julius Mark A.	**Date Performed:** 10/06/24
Course/Section: CPE212-CPE31S2	**Date Submitted:** 10/09/24
Instructor: Engr. Robin Valenzuela	**Semester and SY:** 1st 2024 - 2025

Activity 7: Managing Files and Creating Roles in Ansible

1. **Objectives:**

1.1 Manage files in remote servers

1.2 Implement roles in ansible

2. **Discussion**:

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.

Task 1: Create a file and copy it to remote servers

1. Using the previous directory we created, create a directory, and named it "*files*." Create a file inside that directory and name it "*default_site.html*." 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.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ mkdir files
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ls
files  inventory.yaml  README.md  site.yml
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$
```

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ cd files
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/files$ sudo nano default_site.html
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/files$
```

```
  GNU nano 7.2                                    default_site.html *
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>HTML DOCUMENT</title>
</head>
<body>
    <h1>The snow goose need not to bathe to make itself white<br></h1>
    <h1>Neither need you do anything but be yourself.</h1>
</body>
</html>
```

2. Edit the *site.yml* file and just below the *web_servers* play, create a new file to copy the default html file for site:
   - 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

```
- 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
```

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

```
TASK [copy default html file for site] ************
changed: [192.168.56.104]
```

The task that I did earlier successfully changed in the remote servers.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ sudo nano site.yml
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] ********************************************************************************************
ok: [192.168.56.104]
ok: [192.168.56.106]
ok: [192.168.56.108]

TASK [install updates (Centos)] ***********************************************************************************
skipping: [192.168.56.104]
skipping: [192.168.56.106]
ok: [192.168.56.108]

TASK [install updates (Ubuntu)] ***********************************************************************************
skipping: [192.168.56.108]
ok: [192.168.56.106]
ok: [192.168.56.104]

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

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

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

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

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

TASK [copy default html file for site] ******************************************************************************
ok: [192.168.56.104]

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

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

TASK [install mariadb package (CentOS)] *****************************************************************************
skipping: [192.168.56.106]

TASK [install mariadb package (Ubuntu)] *****************************************************************************
changed: [192.168.56.106]

TASK [Mariadb- Restarting/Enabling] *********************************************************************************
changed: [192.168.56.106]

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

TASK [Gathering Facts] **********************************************************************************************
ok: [192.168.56.108]

TASK [install samba package] ****************************************************************************************
changed: [192.168.56.108]

PLAY RECAP **********************************************************************************************************
192.168.56.104             : ok=5    changed=0    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0
192.168.56.106             : ok=5    changed=2    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
192.168.56.108             : ok=4    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$
```

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.

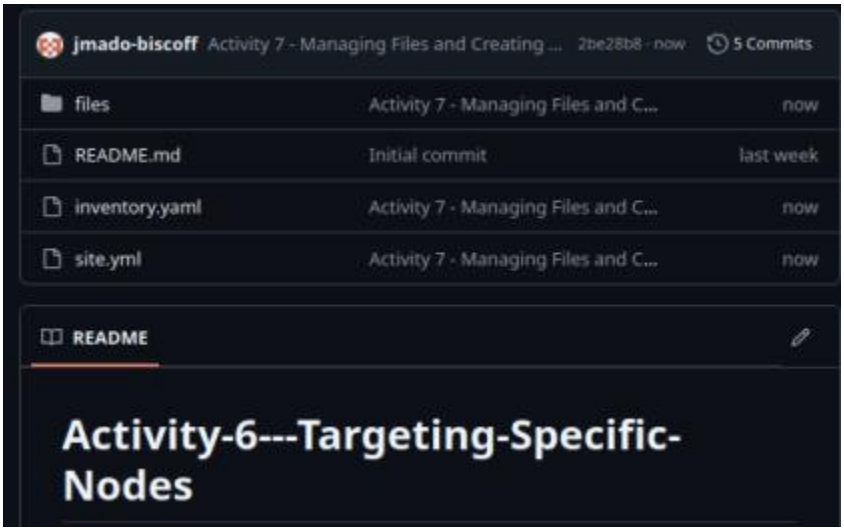
```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ssh julius-de-omampo@server1
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Wed Oct  9 00:48:51 2024 from 192.168.56.105
julius-de-omampo@server1:~$ cat /var/www/html/index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>HTML DOCUMENT</title>
</head>
<body>
    <h1>The snow goose need not to bathe to make itself white<br></h1>
    <h1>Neither need you do anything but be yourself.</h1>
</body>
</html>
julius-de-omampo@server1:~$
```

```
[julius-de-omampo@localhost ~]$ cat /var/www/html/index.html
cat: /var/www/html/index.html: No such file or directory
[julius-de-omampo@localhost ~]$
```

As you can see, the change only went to the web_server address as it was the only one appointed for it, the CentOS does not posses the changes made in the playbook.

5. Sync your local repository with GitHub and describe the changes.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ git add .
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ls
files  inventory.yaml  README.md  site.yml
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   files/default_site.html
        modified:   inventory.yaml
        modified:   site.yml

julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ git commit -m "Activity 7 - Managing Files and Cre
ating Roles in Ansible"
[main 2be28b8] Activity 7 - Managing Files and Creating Roles in Ansible
 3 files changed, 31 insertions(+), 8 deletions(-)
 create mode 100644 files/default_site.html
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ git push origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 3 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 912 bytes | 912.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:jmado-biscoff/Activity-6---Targeting-Specific-Nodes.git
   d925c32..2be28b8  main -> main
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$
```



The syncing process successfully override changes made from the previous activity.

https://github.com/jmado-biscoff/Activity-6---Targeting-Specific-Nodes.git

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_amd64.zip
         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.

```
  GNU nano 7.2                                          inventory.yaml *
[web_servers]
192.168.56.104

[db_servers]
192.168.56.106

[file_servers]
192.168.56.108

[workstations]
192.168.56.104
```

3. Run the playbook. Describe the output.

```
PLAY [workstations] ************************************************************

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

TASK [install unzip] ***********************************************************
ok: [192.168.56.104]

TASK [install terraform] *******************************************************
changed: [192.168.56.104]

PLAY RECAP *********************************************************************
192.168.56.104             : ok=8    changed=1    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0
192.168.56.106             : ok=5    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
192.168.56.108             : ok=4    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

4. On the Ubuntu remote workstation, type terraform to verify installation of terraform. Describe the output.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ssh julius-de-omampo@server1
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Wed Oct  9 01:08:42 2024 from 192.168.56.105
julius-de-omampo@server1:~$
```

Terraform has been properly installed in the workstation (Ubuntu server1).

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.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ mv site.yml old_site.yml
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ls
files  inventory.yaml  old_site.yml  README.md
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ sudo nano site.yml
```

```
 GNU nano 7.2                                                site.yml *
---
- 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
```

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.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/roles$ mkdir web_servers file_servers db_servers wo
rkstations
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/roles$ ls
db_servers  file_servers  web_servers  workstations
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/roles$ mkdir -p web_servers/tasks file_servers/task
s db_servers/tasks workstations/tasks
```

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/roles$ ls -aR
.:
.  ..  base  db_servers  file_servers  web_servers  workstations

./base:
.  ..  tasks

./base/tasks:
.  ..  main.yml

./db_servers:
.  ..  tasks

./db_servers/tasks:
.  ..  main.yml

./file_servers:
.  ..  tasks

./file_servers/tasks:
.  ..  main.yml

./web_servers:
.  ..  tasks

./web_servers/tasks:
.  ..  main.yml

./workstations:
.  ..  tasks

./workstations/tasks:
.  ..  main.yml
```

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.

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes/roles$ ls -Ra
.:
.  ..  db_servers  file_servers  web_servers  workstations

./db_servers:
.  ..  tasks

./db_servers/tasks:
.  ..  main.yml

./file_servers:
.  ..  tasks

./file_servers/tasks:
.  ..  main.yml

./web_servers:
.  ..  tasks

./web_servers/tasks:
.  ..  main.yml

./workstations:
.  ..  tasks

./workstations/tasks:
.  ..  main.yml
```

```
 GNU nano 7.2                                  roles/base/tasks/main.yml *
---
- name: install updates (Centos)
  tags: always
  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"
```

```
 GNU nano 7.2                              roles/file_servers/tasks/main.yml *
---
- name: install samba package
  tags: samba
  package:
    name: samba
    state: latest
```

```
 GNU nano 7.2                                  roles/web_servers/tasks/main.yml *
---
- name: install apache and php 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,apache2,ubuntu
  dnf:
    name:
      - httpd
      - php
    state: latest
  when: ansible_distribution == "CentOS"

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

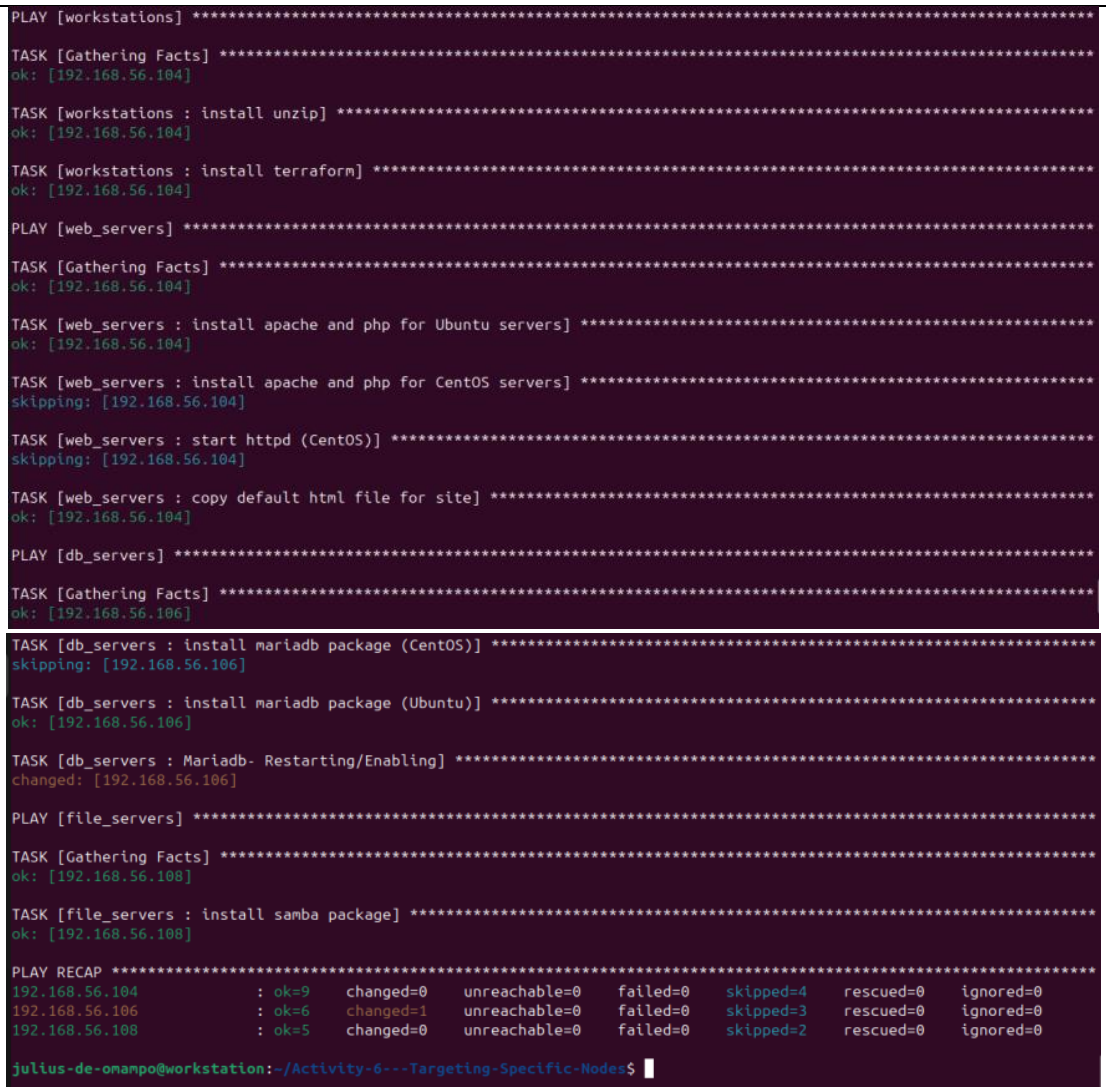
```
- name: copy default html file for site
  tags: apache,apache2,httpd
  copy:
    src: default_site.html
    dest: /var/www/html/index.html
    owner: root
    group: root
    mode: 0644
```

```
 GNU nano 7.2                                  roles/db_servers/tasks/main.yml
---
- 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
```

```
 GNU nano 7.2                                  roles/workstations/tasks/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_amd64.zip
    dest: /usr/local/bin
    remote_src: yes
    mode: 0755
    owner: root
    group: root
```

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

```
julius-de-omampo@workstation:~/Activity-6---Targeting-Specific-Nodes$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************************************
ok: [192.168.56.104]
ok: [192.168.56.106]
ok: [192.168.56.108]

TASK [update repository index (CentOS)] ****************************************************************************
skipping: [192.168.56.104]
skipping: [192.168.56.106]
ok: [192.168.56.108]

TASK [install updates (Ubuntu)] ************************************************************************************
skipping: [192.168.56.108]
ok: [192.168.56.106]
ok: [192.168.56.104]

TASK [base : install updates (Centos)] *****************************************************************************
skipping: [192.168.56.104]
skipping: [192.168.56.106]
ok: [192.168.56.108]

TASK [base : install updates (Ubuntu)] *****************************************************************************
skipping: [192.168.56.108]
ok: [192.168.56.106]
ok: [192.168.56.104]
```



Executing the ansible playbook with assigned roles was a success.

Reflections:

Answer the following:

1. What is the importance of creating roles?

   Creating roles in Ansible is important because it promotes modularity and reusability in playbooks. Roles allow you to organize tasks, variables, files, templates, and handlers into a structured format, making it easier to manage complex configurations. This modular approach not only enhances readability and maintainability but also facilitates collaboration, as roles can be shared and reused across different projects. By encapsulating related tasks, roles simplify the development process and enable efficient scaling of automation efforts.

2. What is the importance of managing files?

   Managing files is crucial in any system or application as it ensures organization, security, and efficient access to data. Proper file management allows for easy

retrieval, modification, and sharing of information, enhancing productivity and collaboration. It also plays a vital role in maintaining system performance and integrity by preventing data loss and redundancy. Effective file management practices, such as version control and backup strategies, help safeguard sensitive information and streamline workflows, ultimately contributing to smoother operations and better resource allocation.