

<b>Name:</b> Bernice M. Peña	<b>Date Performed:</b> 10/17/2023
<b>Course/Section:</b> Managing Enterprise Servers / CPE31S5	<b>Date Submitted:</b> 10/17/2023
<b>Instructor:</b> Engr. Roman Richard	<b>Semester and SY:</b> 1 <sup>st</sup> semester, SY 2023-2024

## Activity 7: Managing Files and Creating Roles in Ansible

### 1. Objectives:

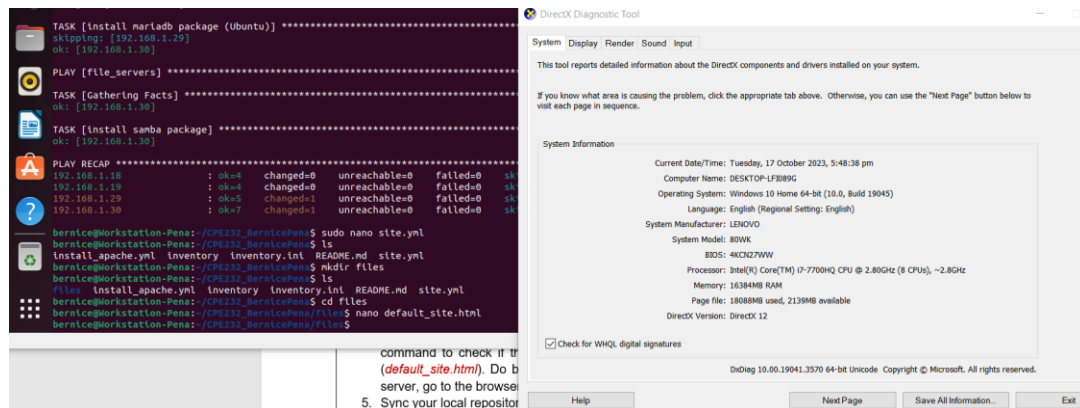
- 1.1 Manage files in remote servers
- 1.2 Implement roles in ansible

### 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.

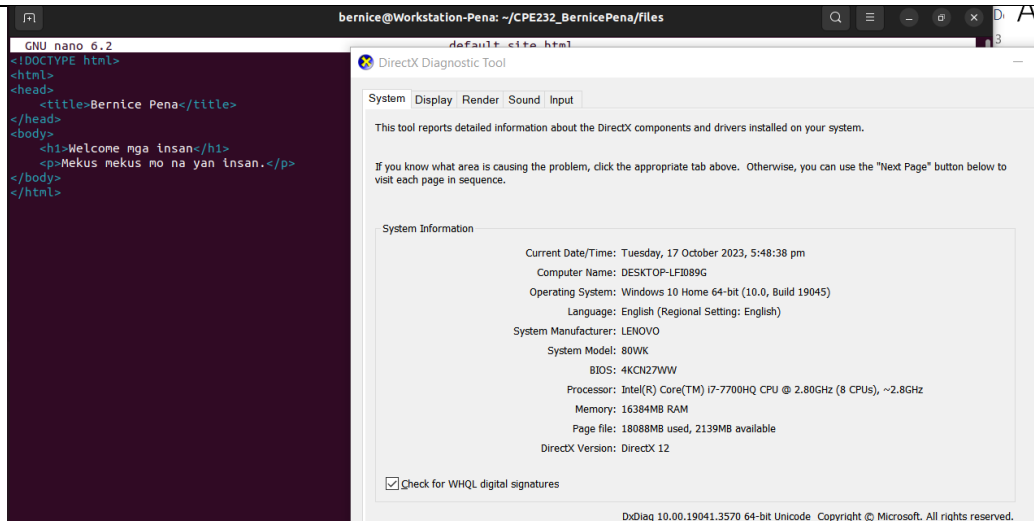
### Task 1: Create a file and copy it to remote servers

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.



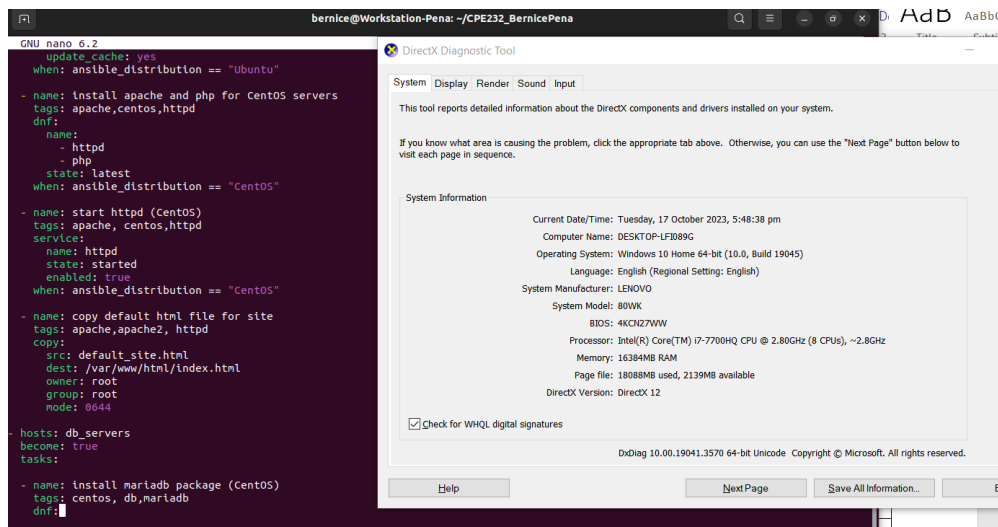
The image shows two side-by-side screenshots. The left screenshot is a terminal window running Ansible commands. It shows the installation of the mariadb package on Ubuntu, gathering facts, and installing the samba package. A play recap shows three hosts (192.168.1.10, 192.168.1.19, 192.168.1.29) with various status indicators. Below the recap, the user is in a shell on the first host, creating a directory 'files', installing the apache.yml inventory, and creating a file 'default\_site.html'.

The right screenshot is a Windows System Information window. It displays system details such as the current date and time (Tuesday, 17 October 2023, 5:48:38 pm), computer name (DESKTOP-LF3089G), operating system (Windows 10 Home 64-bit (18.0, Build 19045)), language (English), system manufacturer (LENOVO), system model (80WK), BIOS (4KCN27WW), processor (Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz (8 CPUs), ~2.8GHz), memory (16384MB RAM), and page file (18068MB used, 2139MB available). The Direct X version is listed as Direct X 12.



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



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

```
bernice@Workstation-Pena:~/CPE232_BernicePena$ ansible-playbook -i inventory site.yml --ask-become-pass
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [192.168.1.30]
ok: [192.168.1.19]
ok: [192.168.1.18]
ok: [192.168.1.29]

TASK [install updates (CentOS)] *****
skipping: [192.168.1.18]
skipping: [192.168.1.19]
skipping: [192.168.1.30]
ok: [192.168.1.29]

TASK [install update (Ubuntu)] *****
skipping: [192.168.1.29]
ok: [192.168.1.19]
ok: [192.168.1.18]
ok: [192.168.1.30]

PLAY [web_servers] *****

TASK [Gathering Facts] *****
ok: [192.168.1.19]
ok: [192.168.1.29]
ok: [192.168.1.18]

TASK [install apache and php for Ubuntu servers] *****
skipping: [192.168.1.29]
ok: [192.168.1.19]
ok: [192.168.1.18]

TASK [install apache and php for CentOS servers] *****
skipping: [192.168.1.18]
skipping: [192.168.1.19]
changed: [192.168.1.29]

TASK [start httpd (CentOS)] *****
skipping: [192.168.1.18]
skipping: [192.168.1.19]
changed: [192.168.1.29]

TASK [copy default html file for site] *****
ok: [192.168.1.18]
ok: [192.168.1.19]
changed: [192.168.1.29]

PLAY [db_servers] *****

TASK [Gathering Facts] *****
ok: [192.168.1.29]
ok: [192.168.1.30]

TASK [install mariadb package (CentOS)] *****
skipping: [192.168.1.30]
ok: [192.168.1.29]

TASK [Mariadb- Restarting/Enabling] *****
changed: [192.168.1.30]
changed: [192.168.1.29]

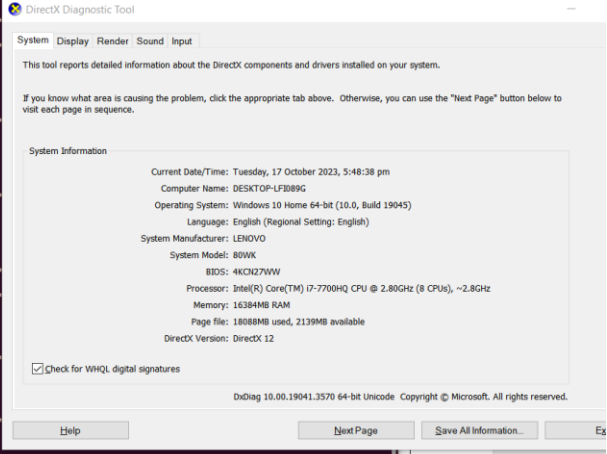
TASK [install mariadb package (Ubuntu)] *****
skipping: [192.168.1.29]
ok: [192.168.1.30]

PLAY [file_servers] *****

TASK [Gathering Facts] *****
ok: [192.168.1.30]

TASK [install samba package] *****
ok: [192.168.1.30]

[...]
```



DirectX Diagnostic Tool

System | Display | Render | Sound | Input

This tool reports detailed information about the DirectX components and drivers installed on your system.

If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the "Next Page" button below to visit each page in sequence.

System Information

Current Date/Time: Tuesday, 17 October 2023, 5:48:38 pm  
Computer Name: DESKTOP-LFJ889G  
Operating System: Windows 10 Home 64-bit (10.0, Build 19045)  
Language: English (Regional Setting: English)  
System Manufacturer: LENOVO  
System Model: 80WK  
BIOS: 4KC27WW  
Processor: Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz (8 CPUs), ~2.8GHz  
Memory: 16384MB RAM  
Page file: 18088MB used, 2139MB available  
DirectX Version: DirectX 12

☒ Check for WHQL digital signatures

DxDiag 10.00.19041.3570 64-bit Unicode Copyright © Microsoft. All rights reserved.

Help Next Page Save All Information... Exit

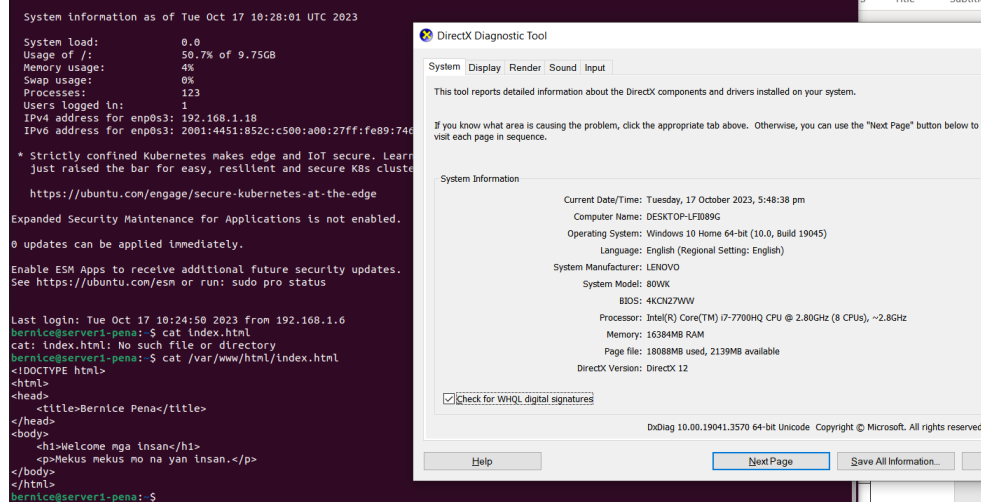
```
bernice@Workstation-Pena:~/CPE232_BernicePena$
```

PLAY RECAP	ok=	changed=	unreachable=	failed=	skipped=	rescued=	ignored=
192.168.1.18	ok=5	changed=0	unreachable=0	failed=0	skipped=3	rescued=0	ignored=0
192.168.1.19	ok=5	changed=0	unreachable=0	failed=0	skipped=3	rescued=0	ignored=0
192.168.1.29	ok=9	changed=4	unreachable=0	failed=0	skipped=3	rescued=0	ignored=0
192.168.1.30	ok=7	changed=1	unreachable=0	failed=0	skipped=2	rescued=0	ignored=0

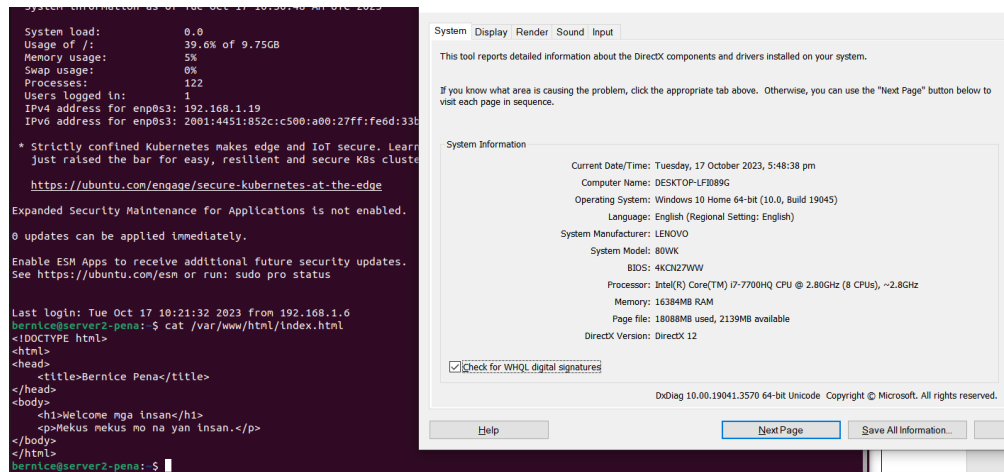
Notice that there were changes in the recently added code for copying the default file under web\_servers. The task to copy the default HTML file to the web\_servers was executed leading to servers 1 and 2 to have the

ok status and CentOS server having the change status since I added its ip under my web\_servers.

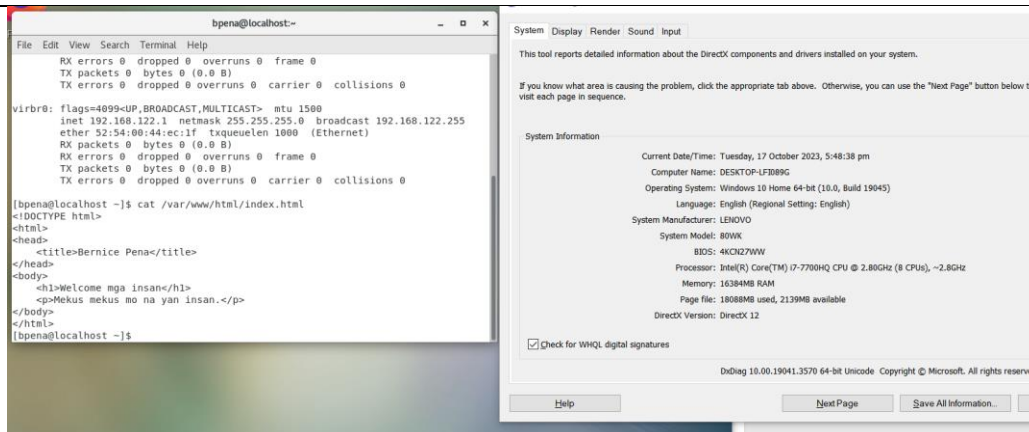
4. Go to the remote servers (*web\_servers*) listed in your inventory. Use cat command to check if the index.html is the same as the local repository file (*default\_site.html*). Do both for Ubuntu and CentOS servers. On the CentOS server, go to the browser and type its IP address. Describe the output.



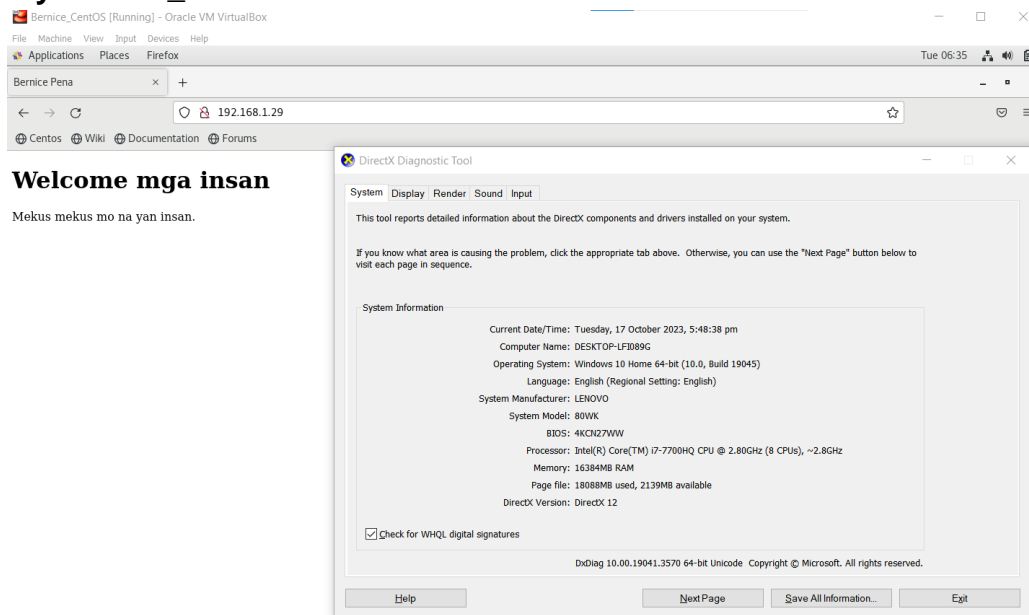
I used ssh command to go to the remove server 1 and used the cat command to check if the index.html is the same as my default\_site.html. It shows in the screenshot above that it is the same with my default\_site.html.



I used the same process for server 2, and it shows that it is also same with my default\_site.html.

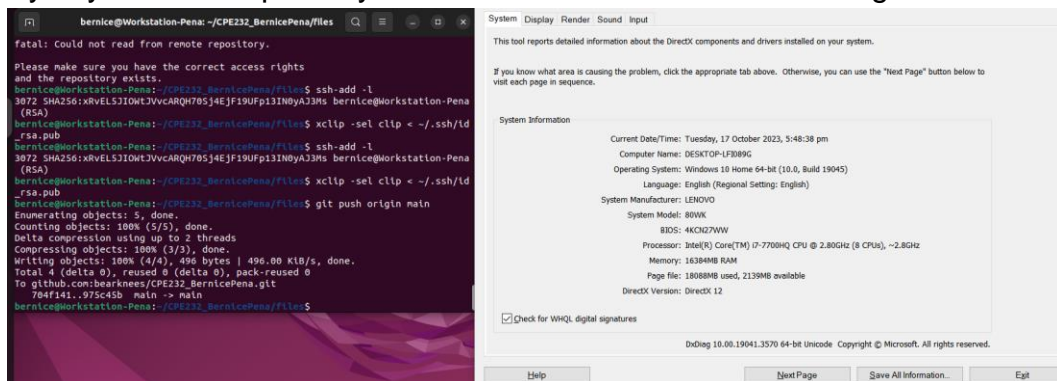


I did the same process for my CentOS server and the result was same with my default\_site.html.

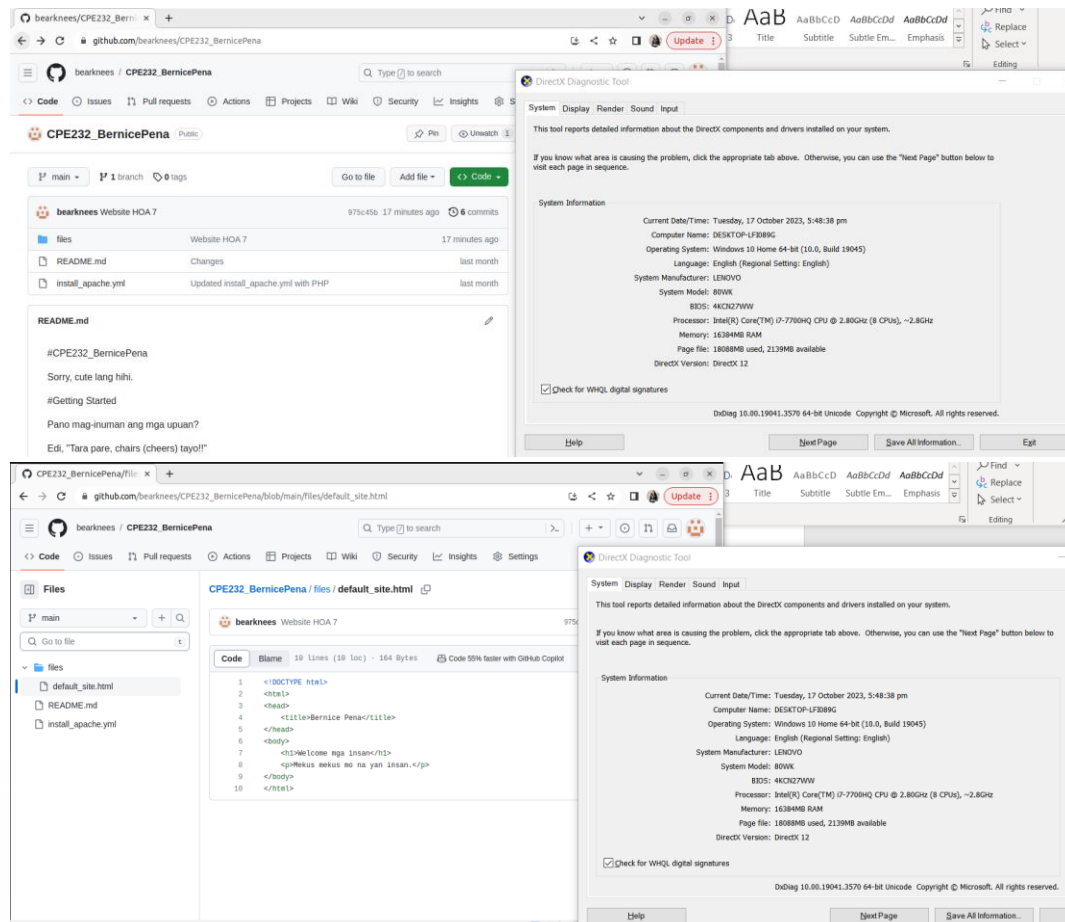


To check it using my browser, I opened a browser on my CentOS and typed its ip address and this was the result, it's the same content as what I have in my default\_site.html.

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



**I created first a new SSH key in order to push what I committed in my GitHub.**



**The files directory including the contents of the directory was successfully synched in my GitHub.**

## **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](https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_linux_amd64.zip)

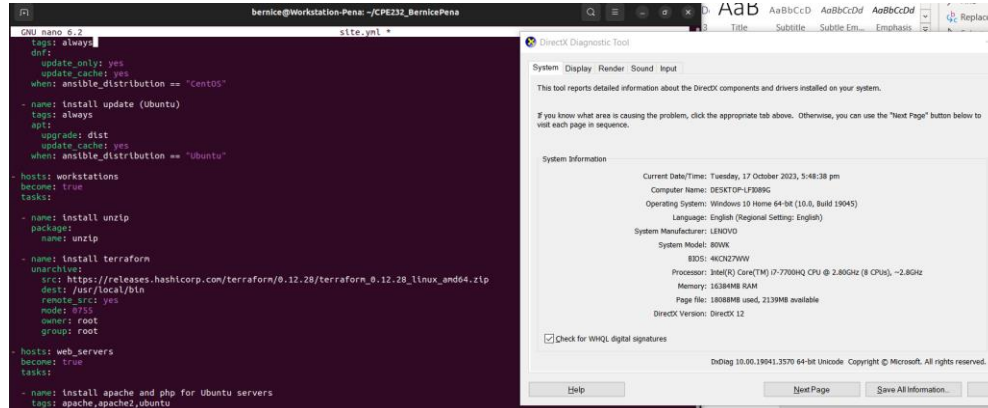
dest: /usr/local/bin

remote\_src: yes

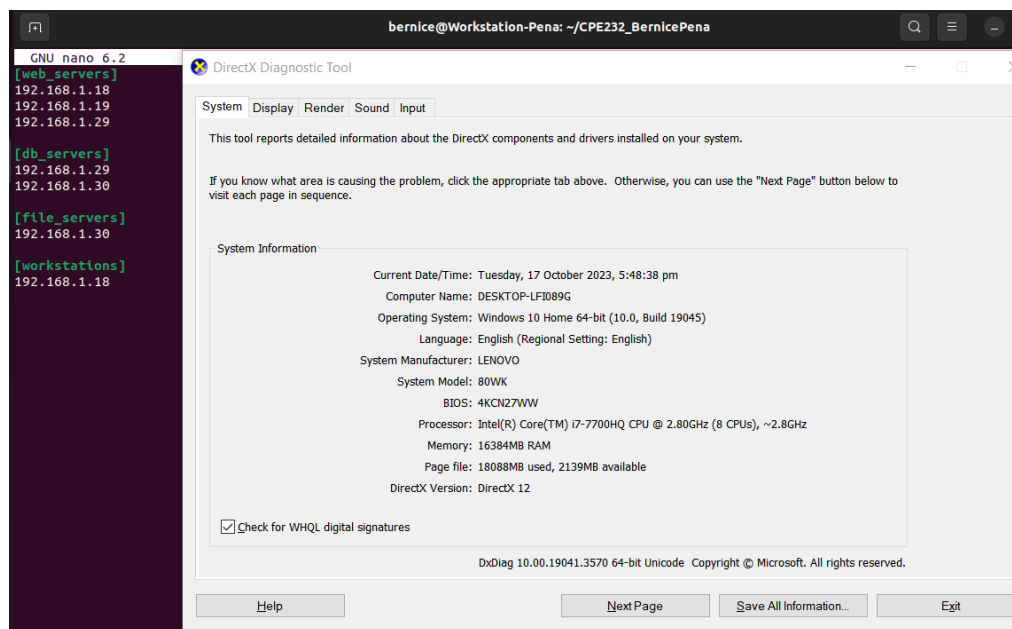
mode: 0755

owner: root

group: root

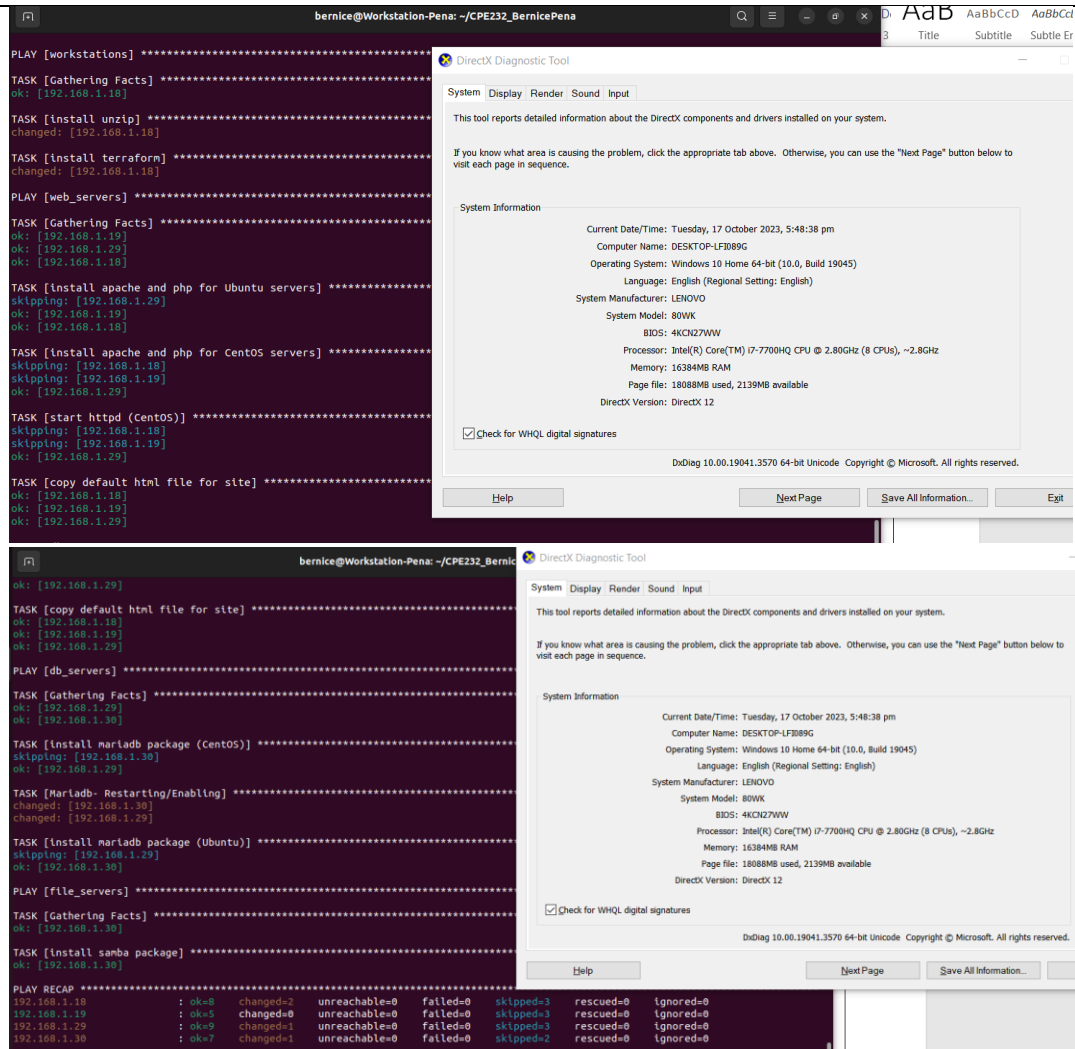


2. Edit the inventory file and add workstations group. Add any Ubuntu remote server. Make sure to remember the IP address.



3. Run the playbook. Describe the output.

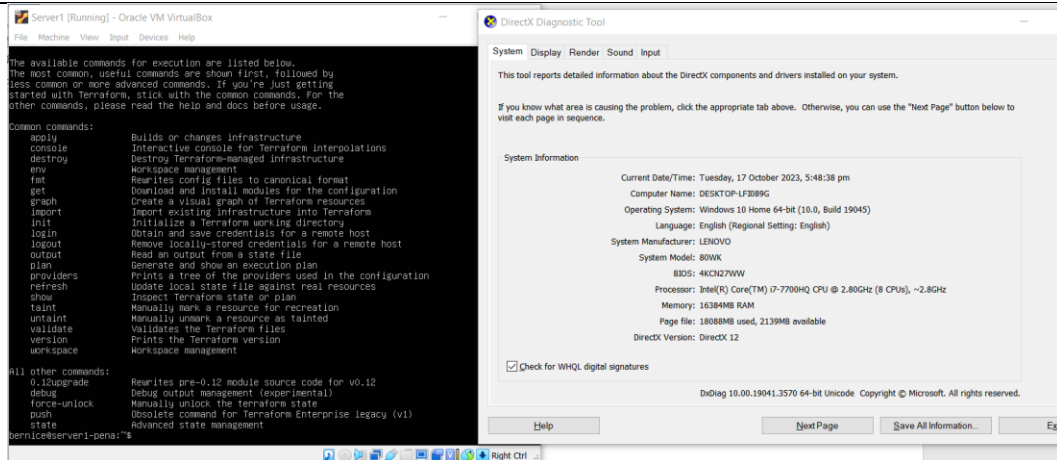




As you can see in my recently added play named workstations, it read my server 1 ip address since that was the address that I added in my recently added group named workstations in my inventory. There was also a change happened in that server since the code is meant to make changes for that server.

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





After executing the command to verify the installation of terraform, the usage of terraform was shown including the commands for execution list.

### 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

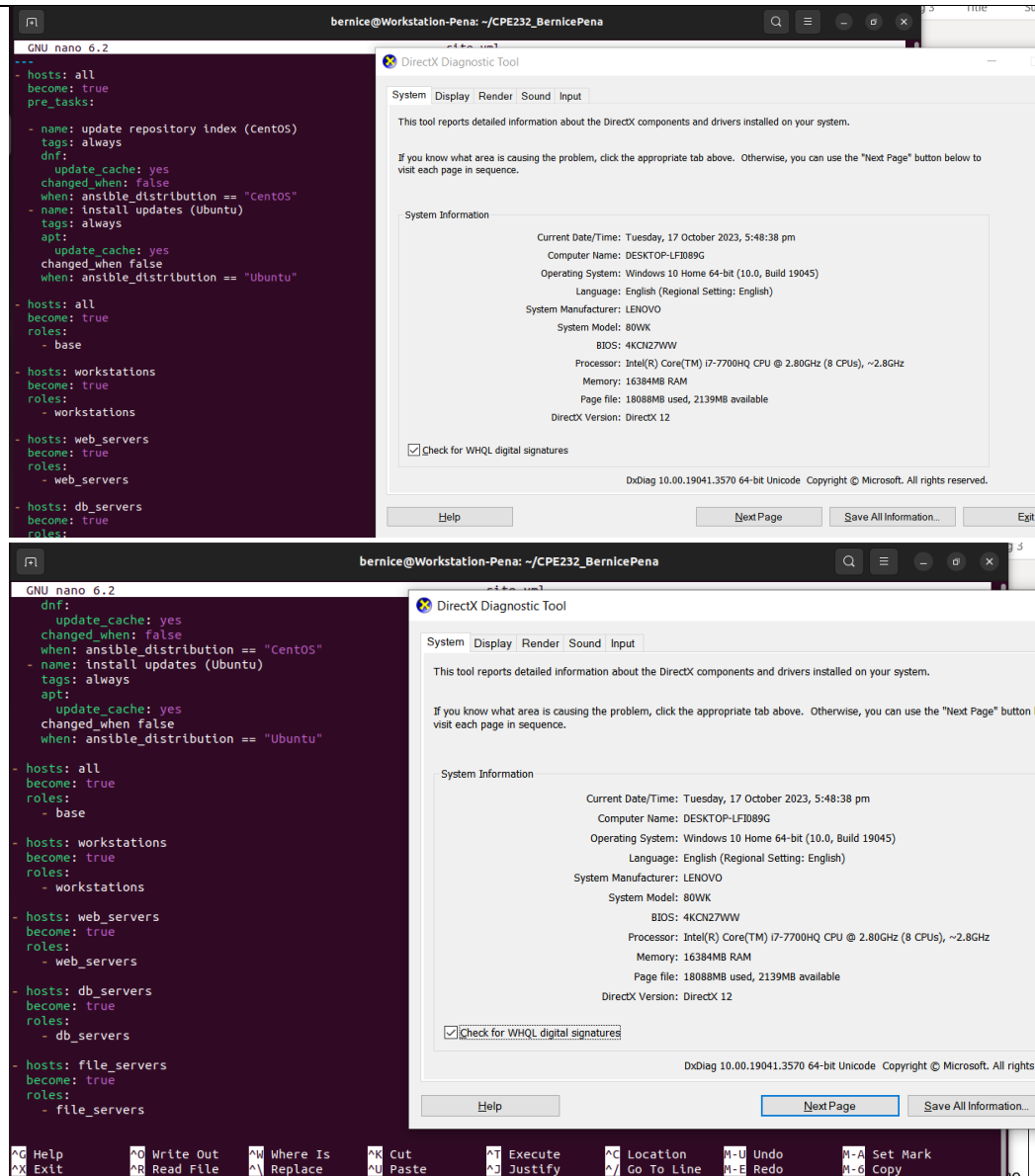
- hosts: web_servers
  become: true
  roles:
    - web_servers

- hosts: db_servers
  become: true
  roles:
    - db_servers

- hosts: file_servers
  become: true
  roles:
    - file_servers

```

Save the file and exit.

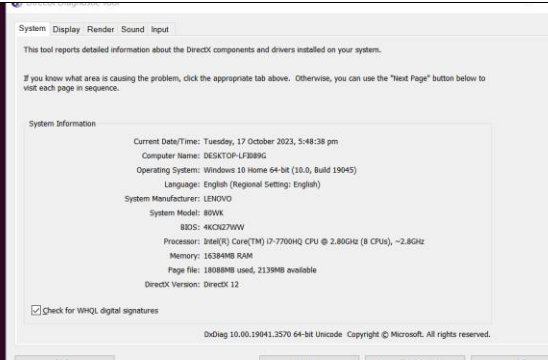


- Under the same directory, create a new directory and name it roles. Enter the roles directory and create new directories: base, web\_servers, file\_servers, db\_servers and workstations. For each directory, create a directory and name it tasks.

```

bernice@Workstation-Pena: ~/CPE232_BernicePena$ sudo nano site.yml
bernice@Workstation-Pena: ~/CPE232_BernicePena$ sudo nano copy_site.yml
bernice@Workstation-Pena: ~/CPE232_BernicePena$ sudo nano site.yml
bernice@Workstation-Pena: ~/CPE232_BernicePena$ mkdir roles
bernice@Workstation-Pena: ~/CPE232_BernicePena$ cd roles
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ mkdir base
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ cd base
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/base$ mkdir tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/base$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ mkdir web_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ cd web_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/web_servers$ mkdir tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/web_servers$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ mkdir file_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ cd file_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/file_servers$ mkdir tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/file_servers$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ mkdir db_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ cd db_servers
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/db_servers$ mkdir tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/db_servers$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ mkdir workstations
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ cd workstations
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/workstations$ mkdir tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/workstations$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/workstations$ ls
tasks
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles/workstations$ cd ..
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$ ls
base db_servers file_servers web_servers workstations
bernice@Workstation-Pena: ~/CPE232_BernicePena/roles$

```



- 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.

### base:

```

GNU nano 6.2
name: install updates (CentOS)
tags: always
dnf:
  update_only: yes
  update_cache: yes
  when: ansible_distribution == "CentOS"

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

```

The screenshot shows the DirectX Diagnostic Tool window with the 'System' tab selected. It displays the following information:

- Current Date/Time: Tuesday, 17 October 2023, 9:12:40 pm
- Computer Name: DESKTOP-LFJ089G
- Operating System: Windows 10 Home 64-bit (10.0, Build 19045)
- Language: English (Regional Setting: English)
- System Manufacturer: LENOVO
- System Model: 80WK
- BIOS: 4KCNZ7WW
- Processor: Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz (8 CPUs), ~2.8GHz
- Memory: 16384MB RAM
- Page file: 17851MB used, 14792MB available
- DirectX Version: DirectX 12

There is a checkbox for 'Check for WHQL digital signatures' which is currently checked. At the bottom, it says 'DxDiag 10.00.19041.3570 64-bit Unicode Copyright © Microsoft. All rights reserved.'

### db\_servers:

```

GNU nano 6.2
- 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 package (Ubuntu)
tags: db, mariadb, ubuntu
apt:
  name: mariadb-server
  state: latest
  when: ansible_distribution == "Ubuntu"

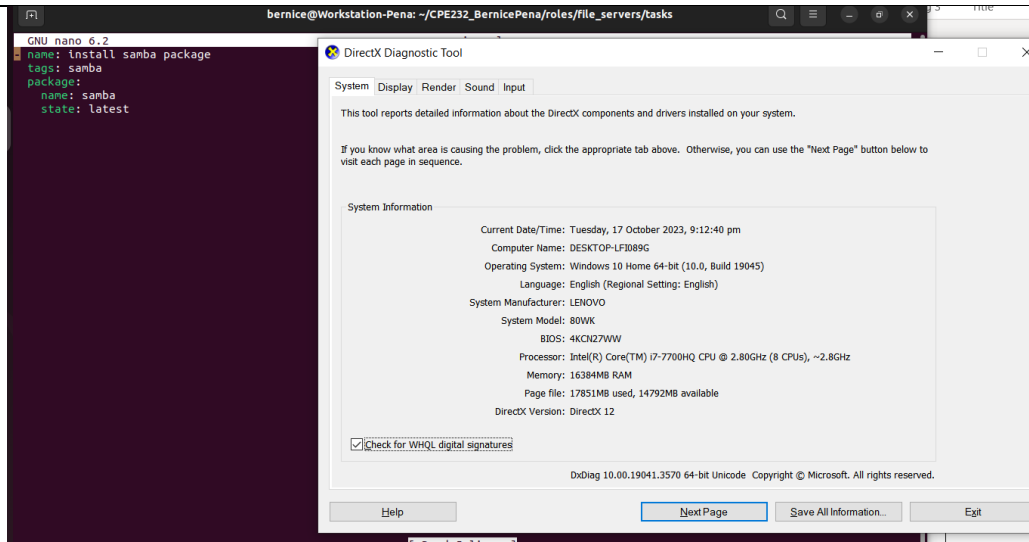
```

The screenshot shows the DirectX Diagnostic Tool window with the 'System' tab selected. It displays the following information:

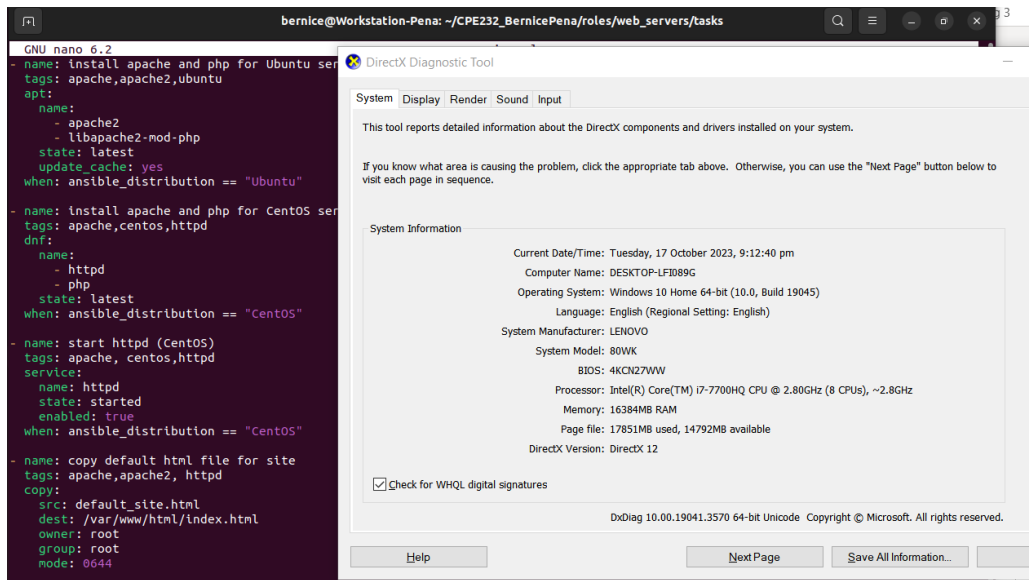
- Current Date/Time: Tuesday, 17 October 2023, 9:12:40 pm
- Computer Name: DESKTOP-LFJ089G
- Operating System: Windows 10 Home 64-bit (10.0, Build 19045)
- Language: English (Regional Setting: English)
- System Manufacturer: LENOVO
- System Model: 80WK
- BIOS: 4KCNZ7WW
- Processor: Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz (8 CPUs), ~2.8GHz
- Memory: 16384MB RAM
- Page file: 17851MB used, 14792MB available
- DirectX Version: DirectX 12

There is a checkbox for 'Check for WHQL digital signatures' which is currently checked. At the bottom, it says 'DxDiag 10.00.19041.3570 64-bit Unicode Copyright © Microsoft. All rights reserved.'

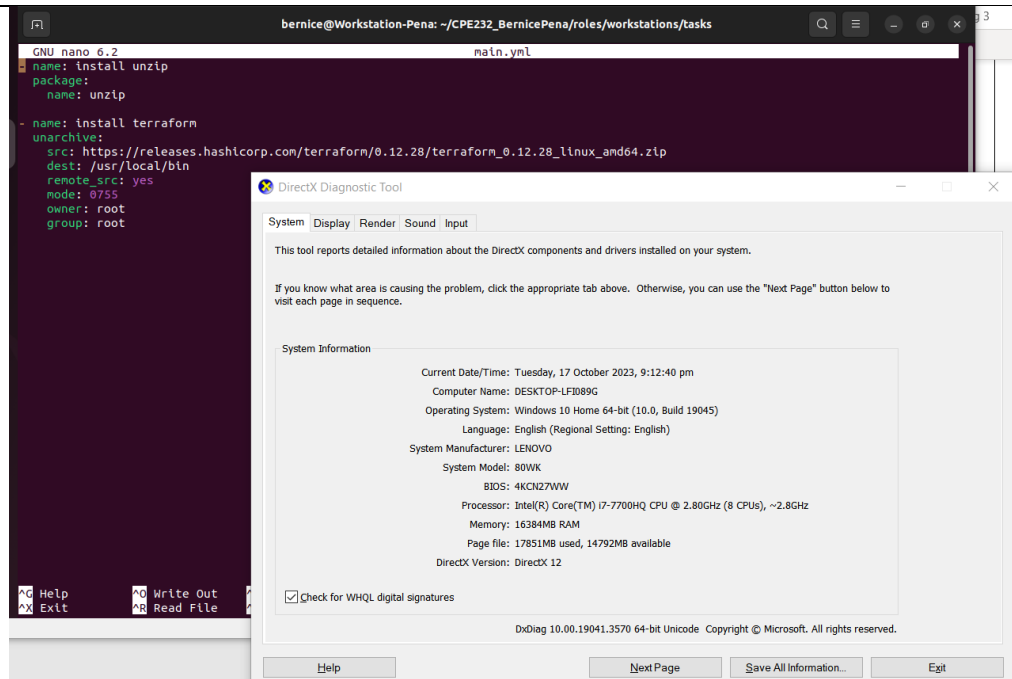
### file\_servers:



## web\_servers:

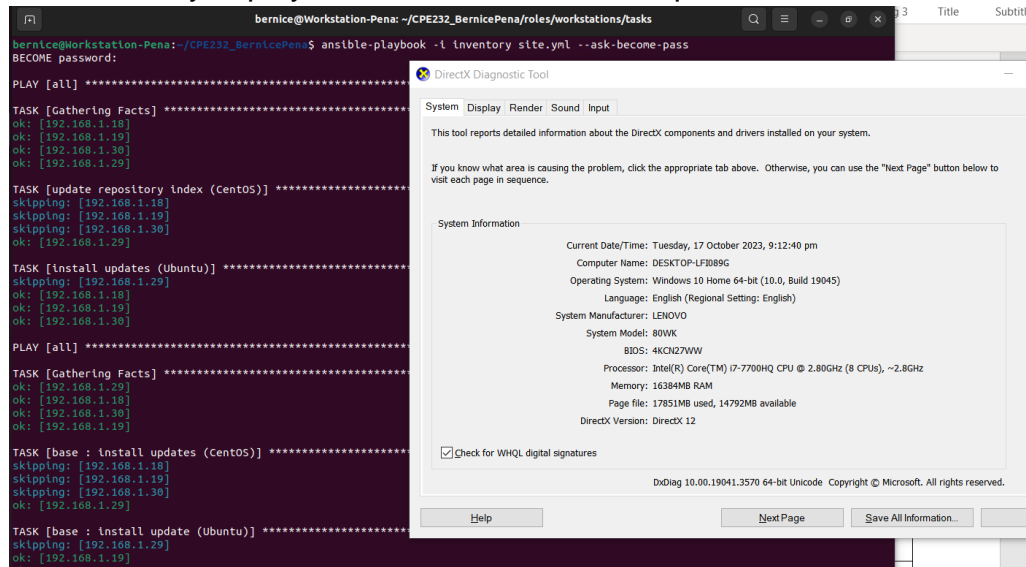


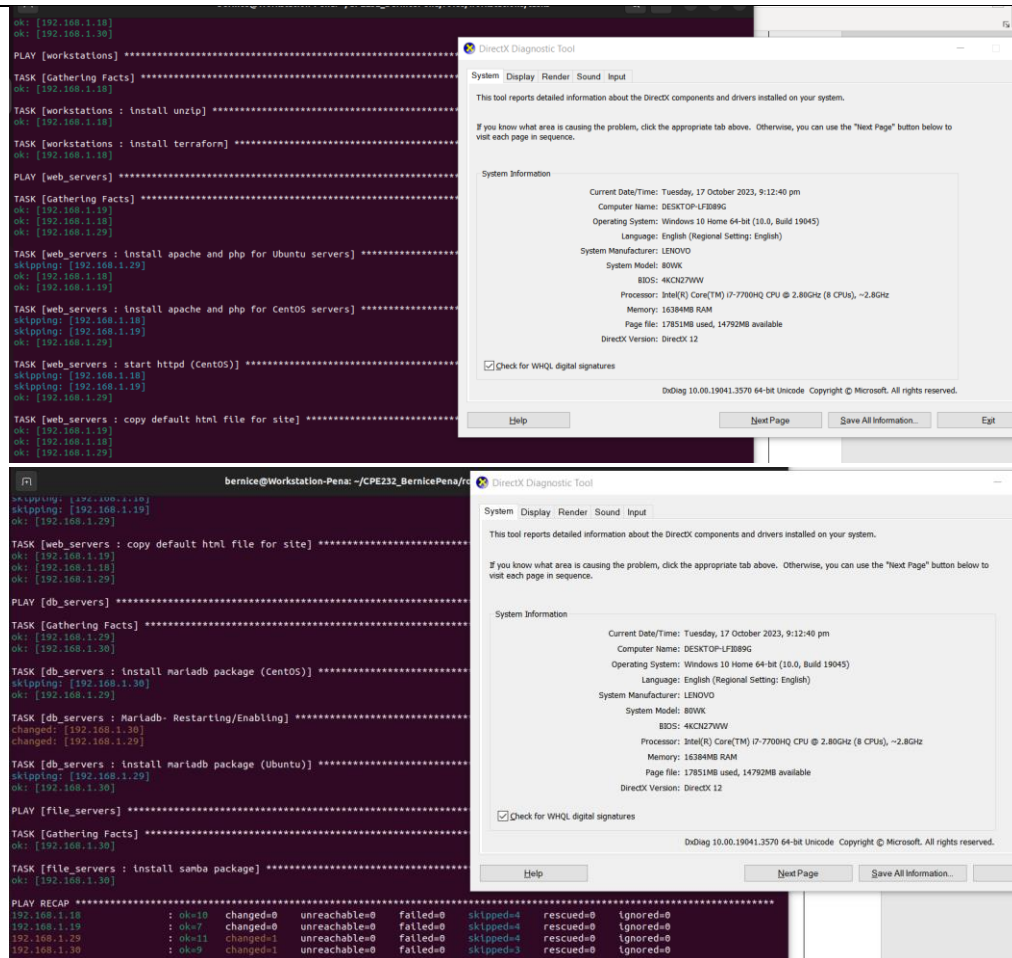
## workstations:



What I did for each `main.yml` is that I copied the tasks meant for each group, I have groups such as `base`, `db_servers`, `file_servers`, `web_servers`, `workstations`. Each of this have their own tasks in the old `site.yml`, I copied the tasks for each group and paste it on the appropriate `yml`.

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





The plays were executed successfully without any fails. Even though I removed all the tasks from the site.yml and just replace it with the use of roles command, the tasks were still executed successfully.

## Reflections:

Answer the following:

1. What is the importance of creating roles?

Creating roles makes it easier and less complex for users when managing and organizing multiple tasks with multiple groups. This helps in breaking down complex automations into smaller parts making it organized just like what I did in the previous tasks, I removed all the tasks written in my old site.yml and then created tasks files that only contains tasks for a specific group. With the use of roles, it makes it easier to identify and resolve issues, instead of analyzing the whole tasks in a single yml, you can just navigate to a specific task inside of a specific directory to fix errors, the separation of tasks allows simplifying the debugging process.

2. What is the importance of managing files?



**Managing files is important when maintaining organized and efficient system, it ensures integrity as well as implementing backup files, this helps when navigating specific files, it is basically a time-saver for the users to locate specific files when needed. A well-organized file provides conveniency and less prone to data loss.**