

<b>Name:</b> John Renzo L. Mendoza	<b>Date Performed:</b> October 17, 2023
<b>Course/Section:</b> CPE31S5	<b>Date Submitted:</b> October 18, 2023
<b>Instructor:</b> Engr. Roman Richard	<b>Semester and SY:</b> 1st Semester 2023-2024

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

#### 1. Objectives:

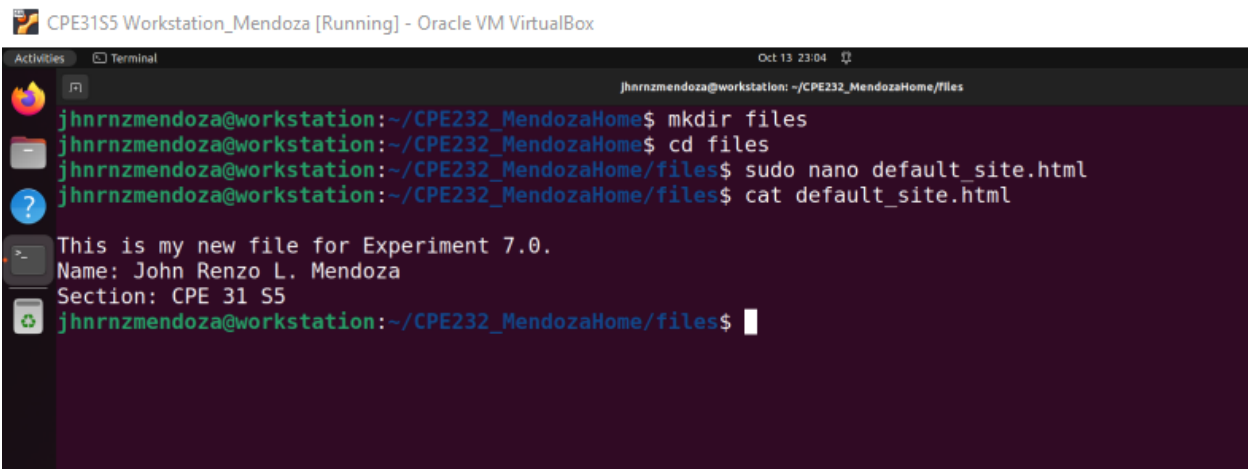
- 1.1 Manage files in remote servers
- 1.2 Implement roles in ansible

#### 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, created 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.



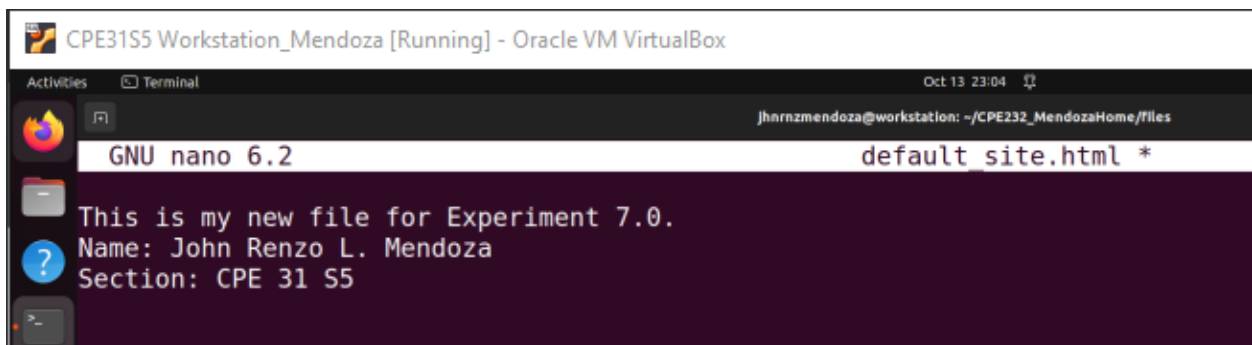
CPE31S5 Workstation\_Mendoza [Running] - Oracle VM VirtualBox

```

jhnrmendoza@workstation: ~/CPE232_MendozaHome/files
jhnrmendoza@workstation:~/CPE232_MendozaHome$ mkdir files
jhnrmendoza@workstation:~/CPE232_MendozaHome$ cd files
jhnrmendoza@workstation:~/CPE232_MendozaHome/files$ sudo nano default_site.html
jhnrmendoza@workstation:~/CPE232_MendozaHome/files$ cat default_site.html

This is my new file for Experiment 7.0.
Name: John Renzo L. Mendoza
Section: CPE 31 S5
jhnrmendoza@workstation:~/CPE232_MendozaHome/files$

```



CPE31S5 Workstation\_Mendoza [Running] - Oracle VM VirtualBox

```

GNU nano 6.2 default_site.html *

This is my new file for Experiment 7.0.
Name: John Renzo L. Mendoza
Section: CPE 31 S5

```

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:

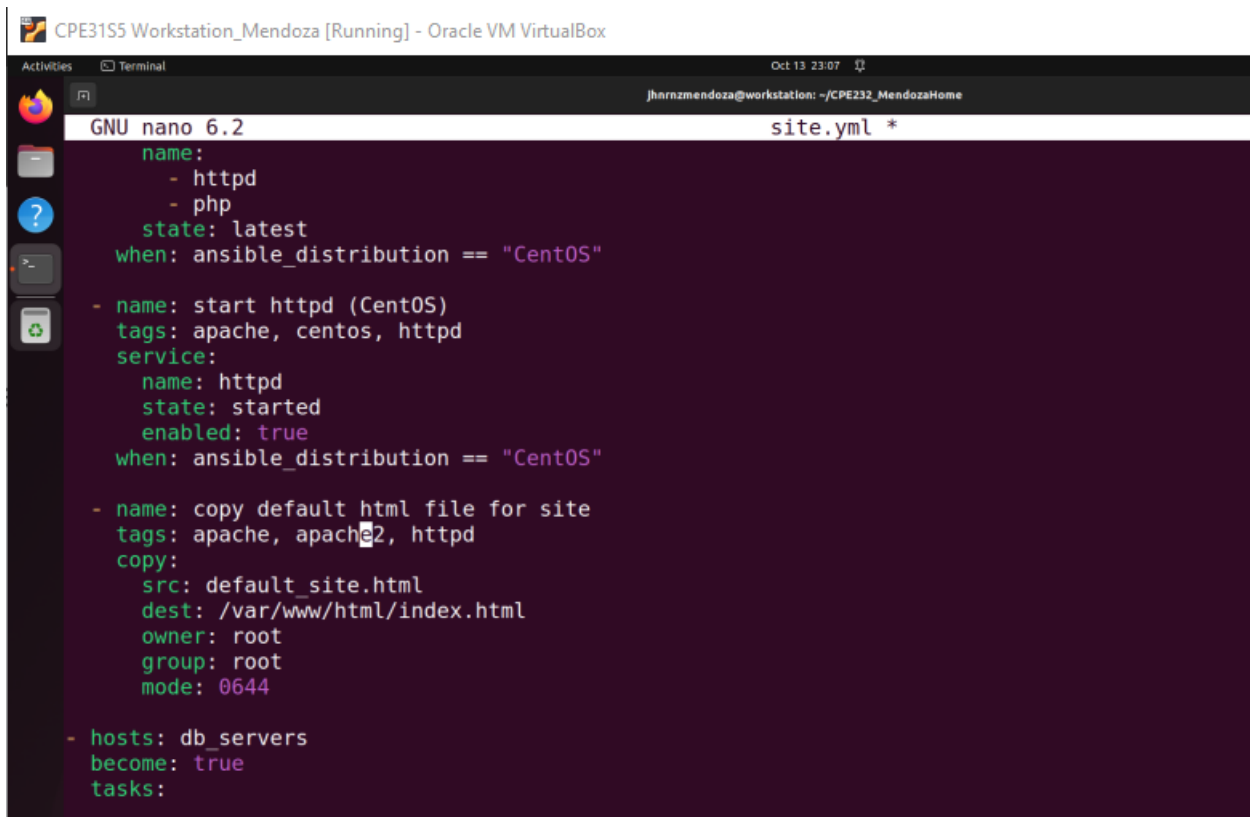
src: default\_site.html

dest: /var/www/html/index.html

owner: root

group: root

mode: 0644



```
GNU nano 6.2 site.yml *
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"

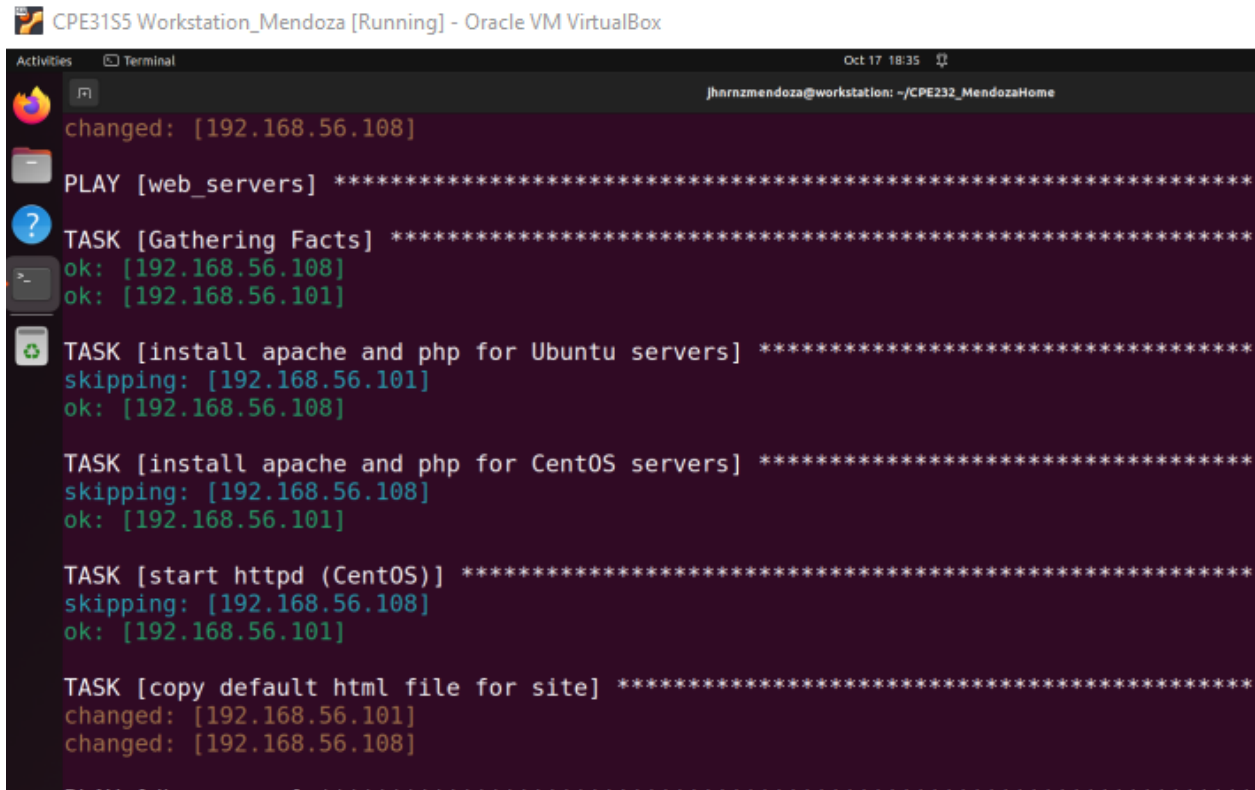
- name: copy default html file for site
  tags: apache, apache2, httpd
  copy:
    src: default_site.html
    dest: /var/www/html/index.html
    owner: root
    group: root
    mode: 0644

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

Observation:

- The given new task was added under the `web_servers` play. This the task would basically copy the file defined in `src`, and send it to the absolute path defined by the `dst` on the remote servers.

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



The screenshot shows a terminal window titled "CPE31S5 Workstation\_Mendoza [Running] - Oracle VM VirtualBox". The terminal output displays the execution of an Ansible playbook named *site.yml*. The output is as follows:

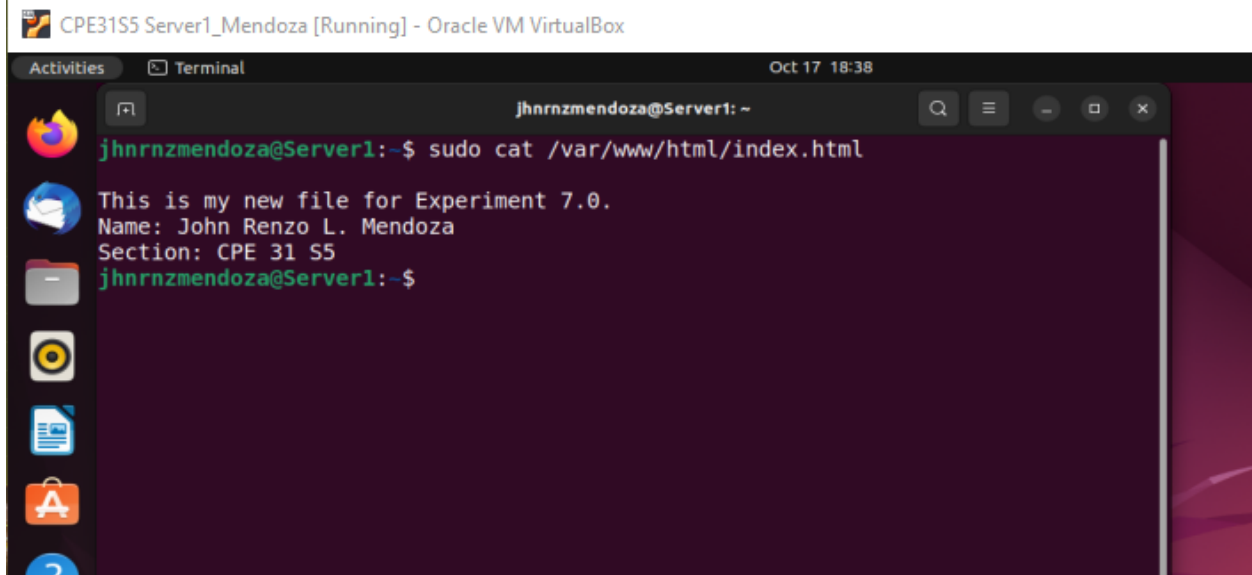
```
changed: [192.168.56.108]
PLAY [web_servers] *****
TASK [Gathering Facts] *****
ok: [192.168.56.108]
ok: [192.168.56.101]
TASK [install apache and php for Ubuntu servers] *****
skipping: [192.168.56.101]
ok: [192.168.56.108]
TASK [install apache and php for CentOS servers] *****
skipping: [192.168.56.108]
ok: [192.168.56.101]
TASK [start httpd (CentOS)] *****
skipping: [192.168.56.108]
ok: [192.168.56.101]
TASK [copy default html file for site] *****
changed: [192.168.56.101]
changed: [192.168.56.108]
```

Observation:

- On this play, we can observe that the new task was successfully performed on the target remote nodes which are the ubuntu server (Server 1) and centos server. we can assume that the file *index.html* has been created on the remote servers specifically on the *dst* absolute path.

- 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.

### Server 1: Web Server

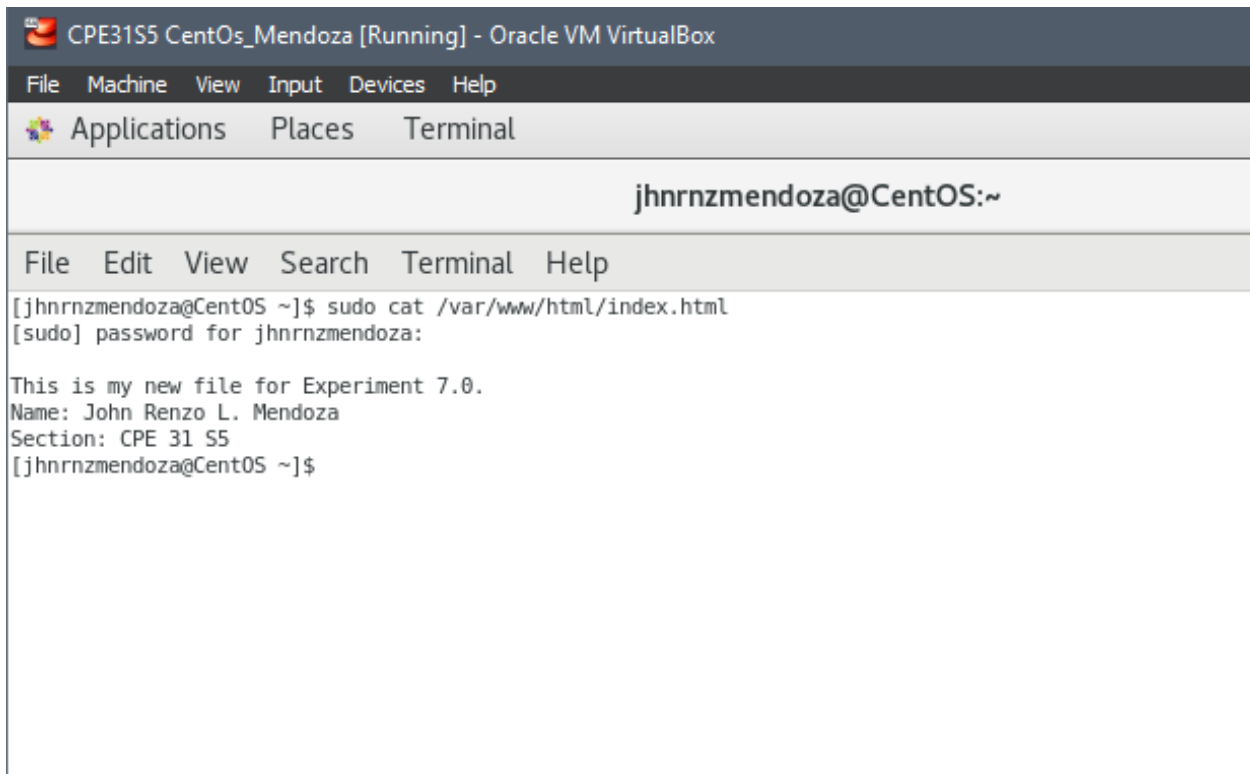


```
CPE31S5 Server1_Mendoza [Running] - Oracle VM VirtualBox
Activities Terminal Oct 17 18:38
jhnrmendoza@Server1: ~
jhnrmendoza@Server1:~$ sudo cat /var/www/html/index.html
This is my new file for Experiment 7.0.
Name: John Renzo L. Mendoza
Section: CPE 31 S5
jhnrmendoza@Server1:~$
```

### Observation:

- In server 1, I have concatenated the contents of the file using the cat command and the absolute path defined earlier. As observed we can see the contents of the file that we have created earlier.

## CentOS: Web\_Server



```
CPE31S5 CentOS_Mendoza [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal

jhnrnmendoza@CentOS:~

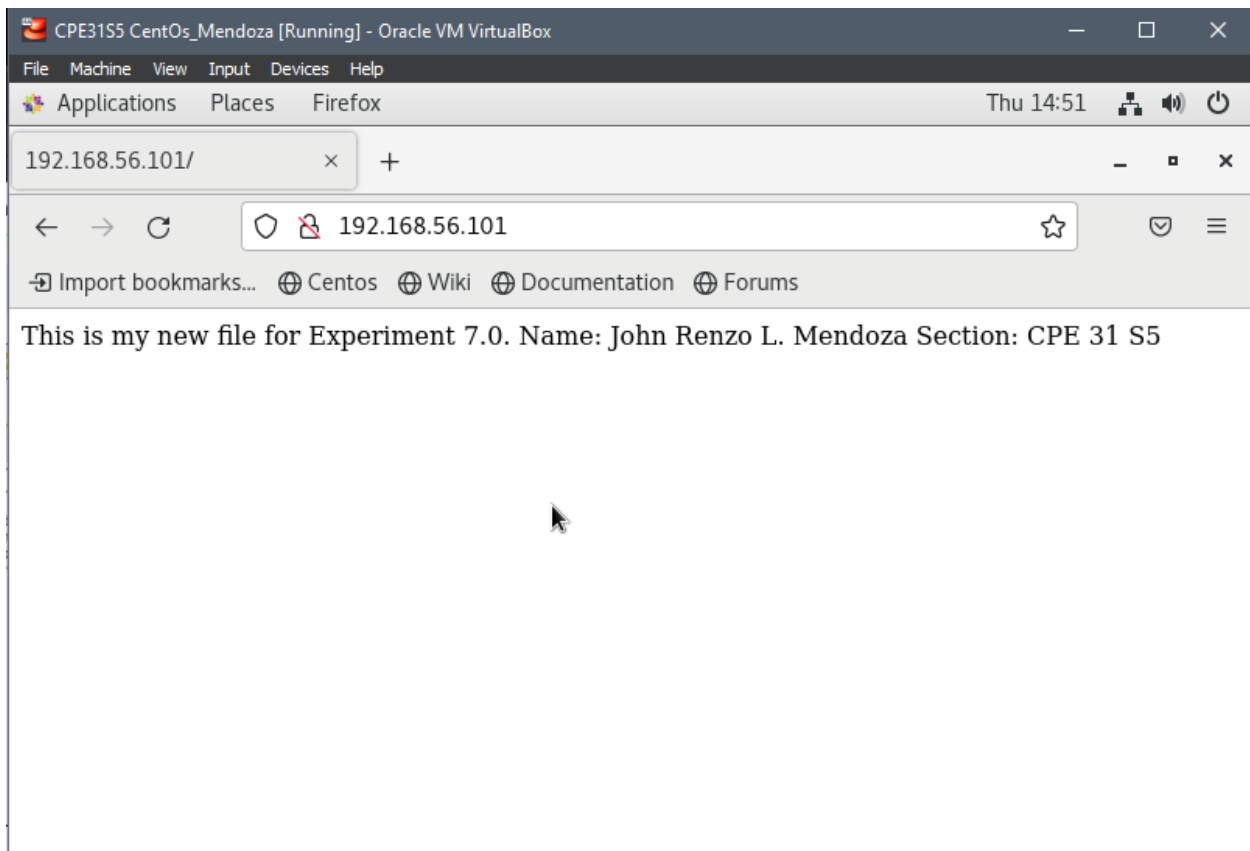
File Edit View Search Terminal Help
[jhnrnmendoza@CentOS ~]$ sudo cat /var/www/html/index.html
[sudo] password for jhnrnmendoza:

This is my new file for Experiment 7.0.
Name: John Renzo L. Mendoza
Section: CPE 31 S5
[jhnrnmendoza@CentOS ~]$
```

### Observation:

- In server 1, I have concatenated the contents of the file using the cat command and the absolute path defined earlier. As observed we can see the contents of the file that we have created earlier.

## Checking of IP Address



### Observation:

- The contents of the file was also seen on the browser by entering the IP address of the db server. I believe this was possible since the absolute path defined is for the internet browsers /var/www/html.

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

```
CPE31S5 Workstation_Mendoza [Running] - Oracle VM VirtualBox
jhnrmendoza@workstation: ~/CPE232_MendozaHome
jhnrmendoza@workstation:~/CPE232_MendozaHome$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   ansible.cfg
        modified:   hosts

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        files/
        site.yml

no changes added to commit (use "git add" and/or "git commit -a")
jhnrmendoza@workstation:~/CPE232_MendozaHome$ git add ~/CPE232_MendozaHome
jhnrmendoza@workstation:~/CPE232_MendozaHome$ 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)
        modified:   ansible.cfg
        new file:   files/default_site.html
        modified:   hosts
        new file:   site.yml

jhnrmendoza@workstation:~/CPE232_MendozaHome$ git commit -m "Experiment 7 Task 1"
[main bca7eab] Experiment 7 Task 1
 4 files changed, 137 insertions(+), 8 deletions(-)
 create mode 100644 files/default_site.html
 rewrite hosts (70%)
 create mode 100644 site.yml
jhnrmendoza@workstation:~/CPE232_MendozaHome$ git push origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 2 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 1.31 KiB | 1.31 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:MendozaJRL/CPE232_MendozaHome.git
 68f3153..bca7eab  main -> main
jhnrmendoza@workstation:~/CPE232_MendozaHome$
```

Observation:

- Using the previously learned git commands, I have successfully committed and pushed the changes to the GitHub cloud repository.

## GitHub Repository

CPE31S5 Workstation\_Mendoza [Running] - Oracle VM VirtualBox

Activities Firefox Web Browser Oct 17 18:45

GitHub - MendozaJRL/CPE232\_MendozaHome

Product Solutions Open Source Pricing Search or jump to...

MendozaJRL / CPE232\_MendozaHome Public Notifications Fork

<> Code Issues Pull requests Actions Projects Security Insights

main 1 branch 0 tags Go to file Code About

JRLMendoza Experiment 7 Task 1 bca7eab 1 minute ago 8 commits

files	Experiment 7 Task 1	1 minute ago
README.md	This is done at home	last month
ansible.cfg	Experiment 7 Task 1	1 minute ago
hosts	Experiment 7 Task 1	1 minute ago
install_apache.yml	Experiment 5 Task 1 and 2	last month
site.yml	Experiment 7 Task 1	1 minute ago

README.md

### CPE232\_MendozaHome

This is a repository

No description, website, or topics provided.

Readme Activity 0 stars 1 watching 0 forks Report repository

Releases No releases published

Packages No packages published

Contributors 2 JRLMendoza

Observation:

- In the GitHub repository, we can observe that the new files site.yml is added and the other related files are also updated.



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

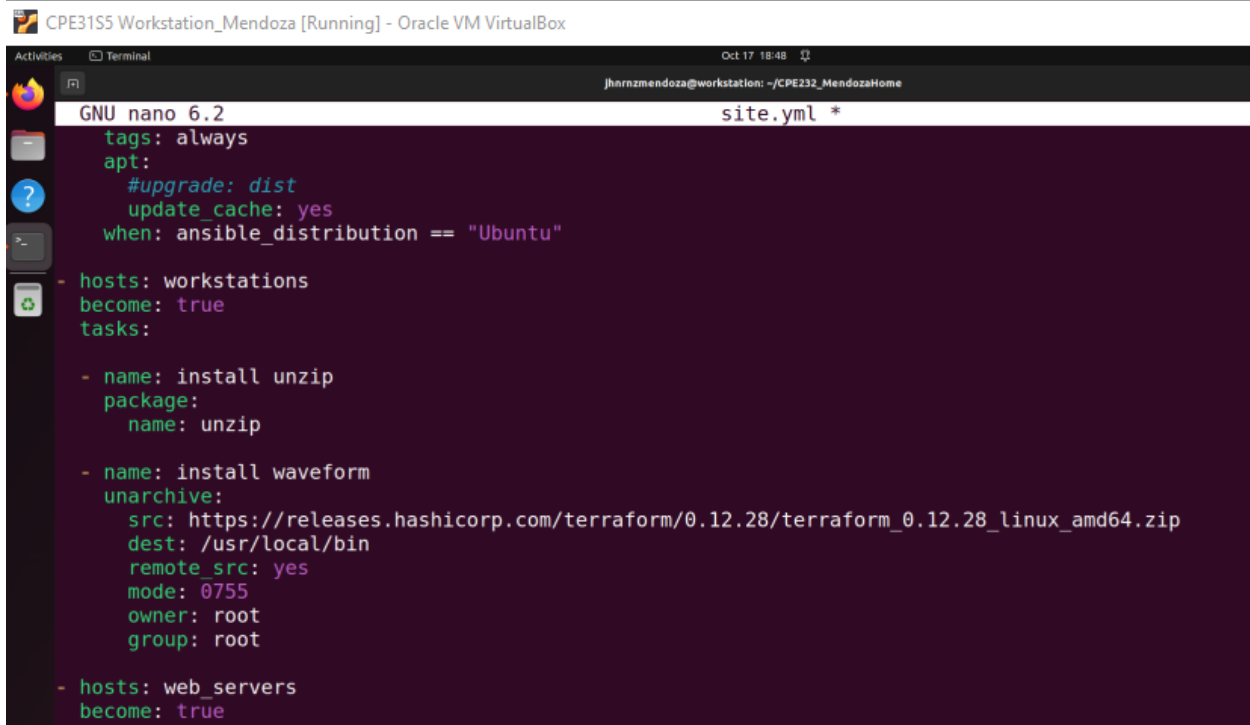
1. Edit the site.yml. Just before the web\_servers play, create a new play:

- 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



The screenshot shows a terminal window titled "CPE3155 Workstation\_Mendoza [Running] - Oracle VM VirtualBox". The terminal is running the GNU nano 6.2 editor, editing the file site.yml. The content of the file is as follows:

```
tags: always
apt:
  #upgrade: dist
  update_cache: yes
when: ansible_distribution == "Ubuntu"

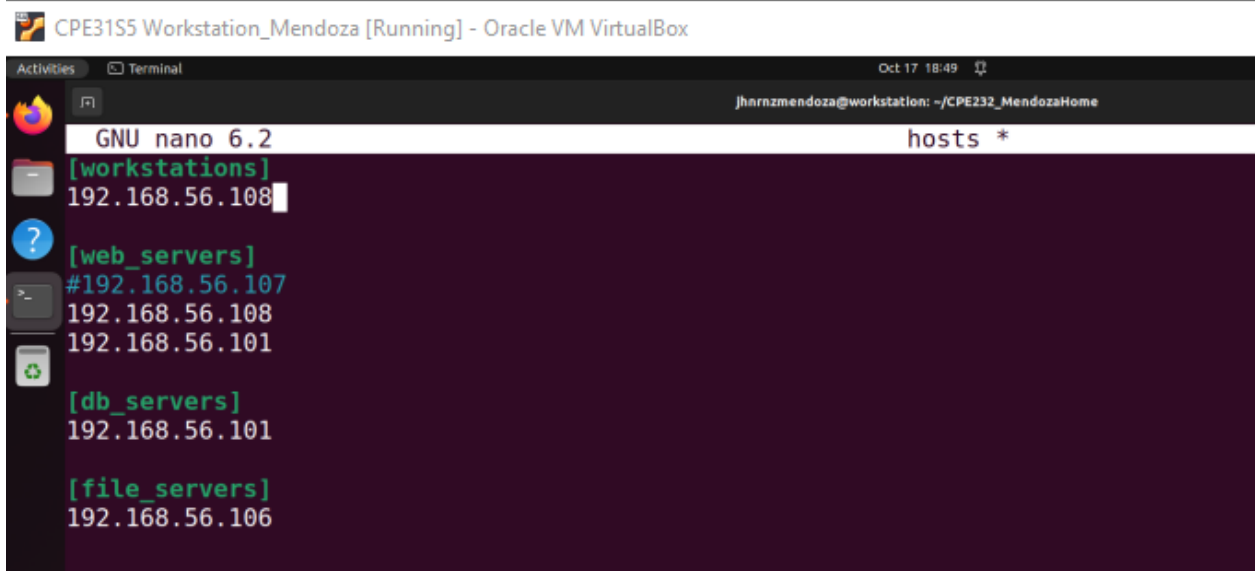
- hosts: workstations
  become: true
  tasks:

    - name: install unzip
      package:
        name: unzip

    - name: install waveform
      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

- hosts: web_servers
  become: true
```

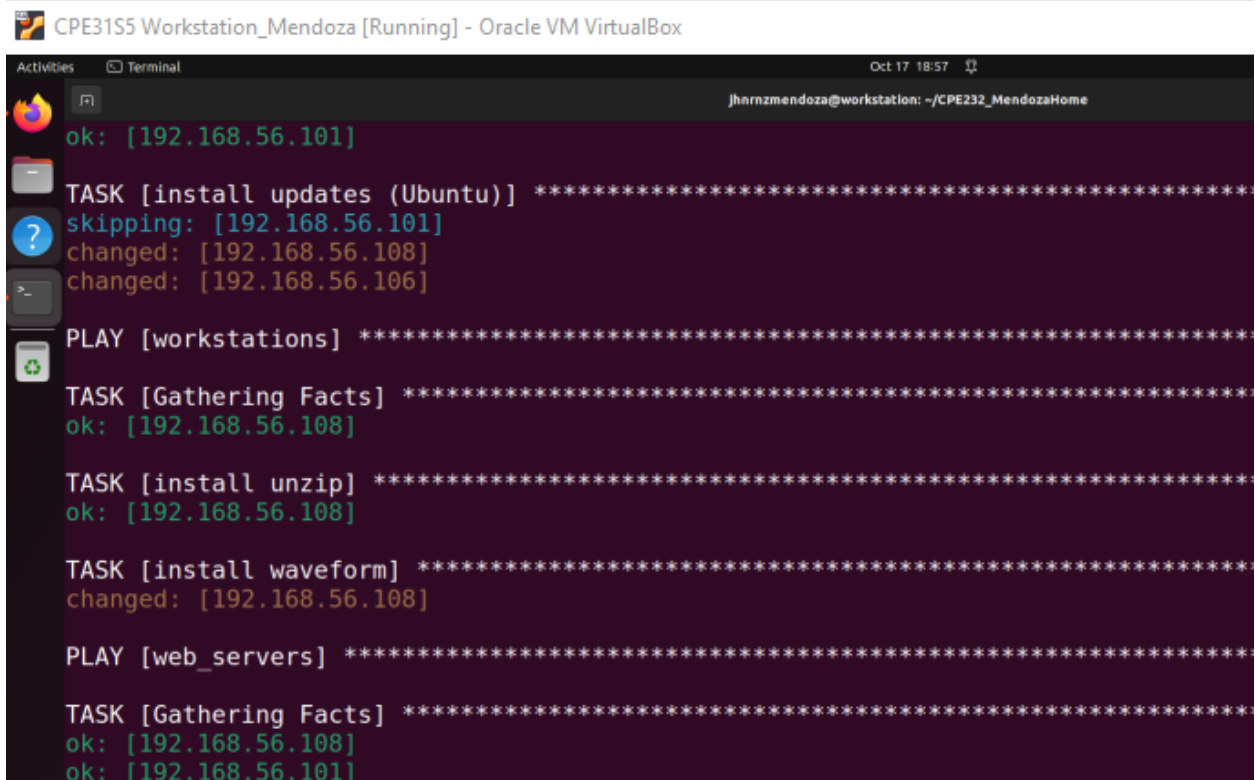
2. Edit the inventory file and add a workstation group. Add any Ubuntu remote server. Make sure to remember the IP address.



The screenshot shows a terminal window titled "CPE31S5 Workstation\_Mendoza [Running] - Oracle VM VirtualBox". The terminal is running the nano text editor, editing the file `hosts`. The content of the file is as follows:

```
GNU nano 6.2 hosts *
[workstations]
192.168.56.108
[web_servers]
#192.168.56.107
192.168.56.108
192.168.56.101
[db_servers]
192.168.56.101
[file_servers]
192.168.56.106
```

3. Run the playbook. Describe the output.



The screenshot shows the same terminal window as before, but now displaying the output of an Ansible playbook run. The output is as follows:

```
CPE31S5 Workstation_Mendoza [Running] - Oracle VM VirtualBox
jhnrmendoza@workstation: ~/CPE232_MendozaHome

ok: [192.168.56.101]

TASK [install updates (Ubuntu)] *****
skipping: [192.168.56.101]
changed: [192.168.56.108]
changed: [192.168.56.106]

PLAY [workstations] *****

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

TASK [install unzip] *****
ok: [192.168.56.108]

TASK [install waveform] *****
changed: [192.168.56.108]

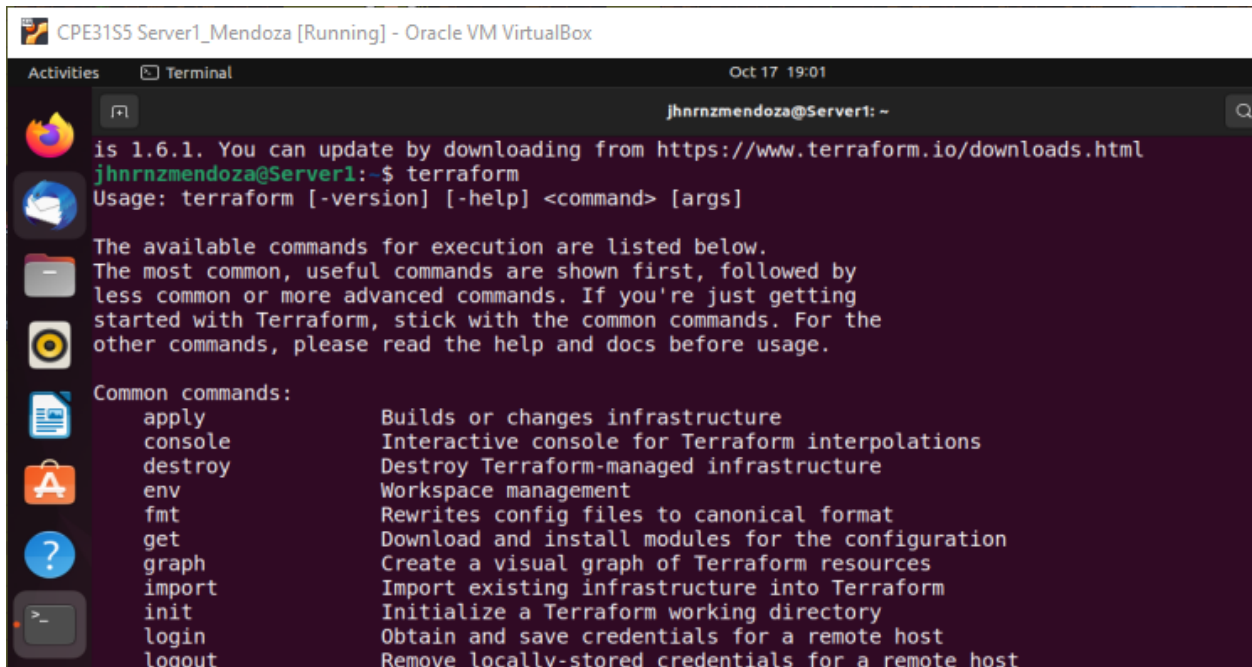
PLAY [web_servers] *****

TASK [Gathering Facts] *****
ok: [192.168.56.108]
ok: [192.168.56.101]
```

Observation:

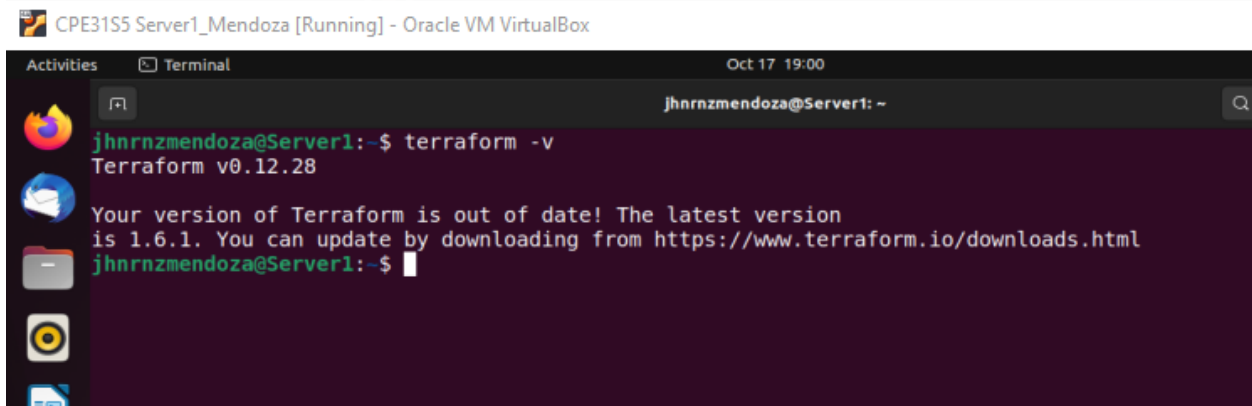
- By following the instructions, the new play for [workstations] has been successfully implemented and was able to perform the defined task to the remote ubuntu server which is server 1.

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



A terminal window titled "CPE31S5 Server1\_Mendoza [Running] - Oracle VM VirtualBox" showing the output of the 'terraform' command. The prompt is 'jhnrmendoza@Server1: ~'. The output includes the version '1.6.1', a link to the download page, and a list of common commands with their descriptions.

```
jhnrmendoza@Server1: ~  
is 1.6.1. You can update by downloading from https://www.terraform.io/downloads.html  
jhnrmendoza@Server1:~$ terraform  
Usage: terraform [-version] [-help] <command> [args]  
  
The available commands for execution are listed below.  
The most common, useful commands are shown first, followed by  
less common or more advanced commands. If you're just getting  
started with Terraform, stick with the common commands. For the  
other commands, please read the help and docs before usage.  
  
Common commands:  
  apply          Builds or changes infrastructure  
  console        Interactive console for Terraform interpolations  
  destroy        Destroy Terraform-managed infrastructure  
  env            Workspace management  
  fmt            Rewrites config files to canonical format  
  get            Download and install modules for the configuration  
  graph          Create a visual graph of Terraform resources  
  import         Import existing infrastructure into Terraform  
  init           Initialize a Terraform working directory  
  login          Obtain and save credentials for a remote host  
  logout         Remove locally-stored credentials for a remote host
```



A terminal window titled "CPE31S5 Server1\_Mendoza [Running] - Oracle VM VirtualBox" showing the output of the 'terraform -v' command. The prompt is 'jhnrmendoza@Server1: ~'. The output shows the version 'v0.12.28' and a message indicating that the version is out of date, with a link to the download page.

```
jhnrmendoza@Server1:~$ terraform -v  
Terraform v0.12.28  
  
Your version of Terraform is out of date! The latest version  
is 1.6.1. You can update by downloading from https://www.terraform.io/downloads.html  
jhnrmendoza@Server1:~$
```

Observation:

- On server 1, we can observe that the terraform was successfully installed by the executed playbook from the control node. This was proven by running the terraform command on server 1. The option -v displays the version of the command.

### 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

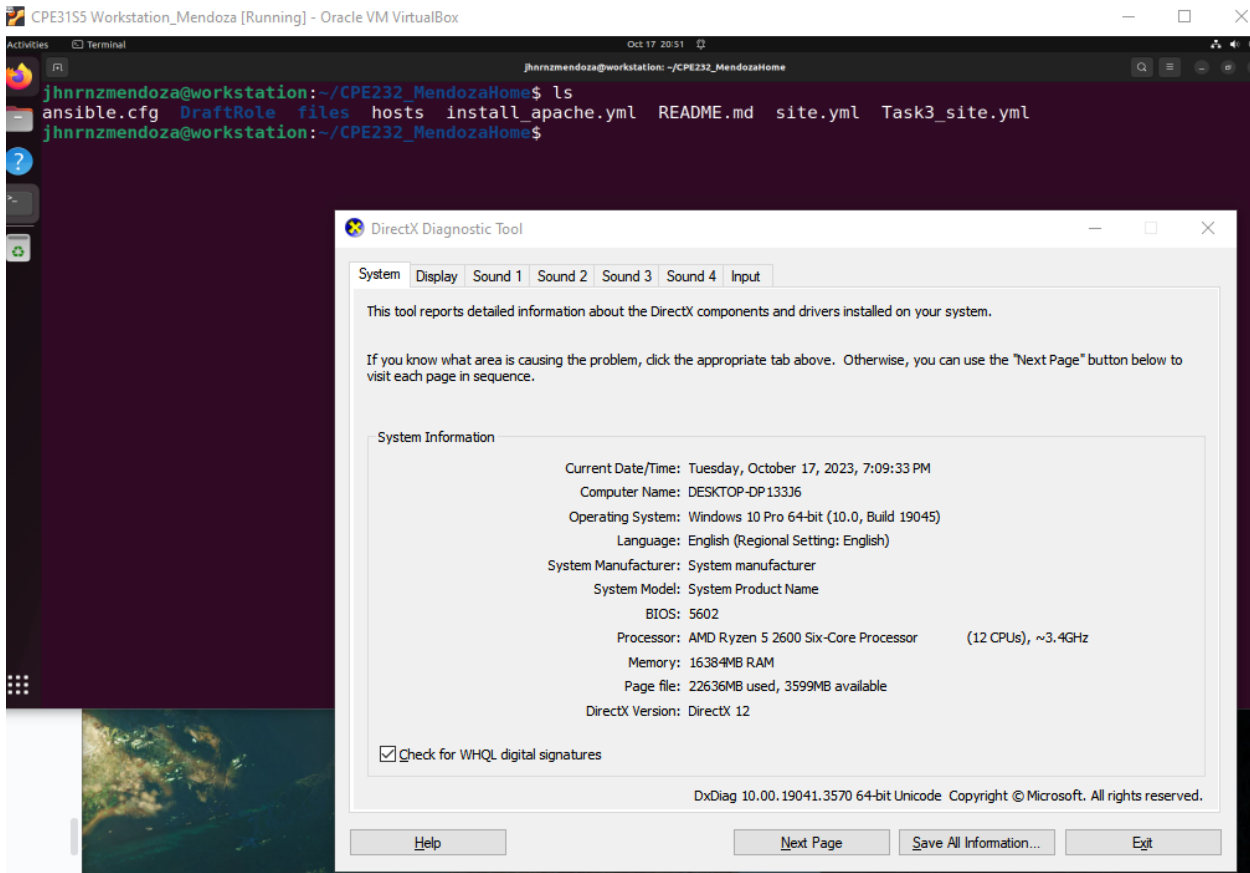
- hosts: web_servers
  become: true
  roles:
    - web_servers

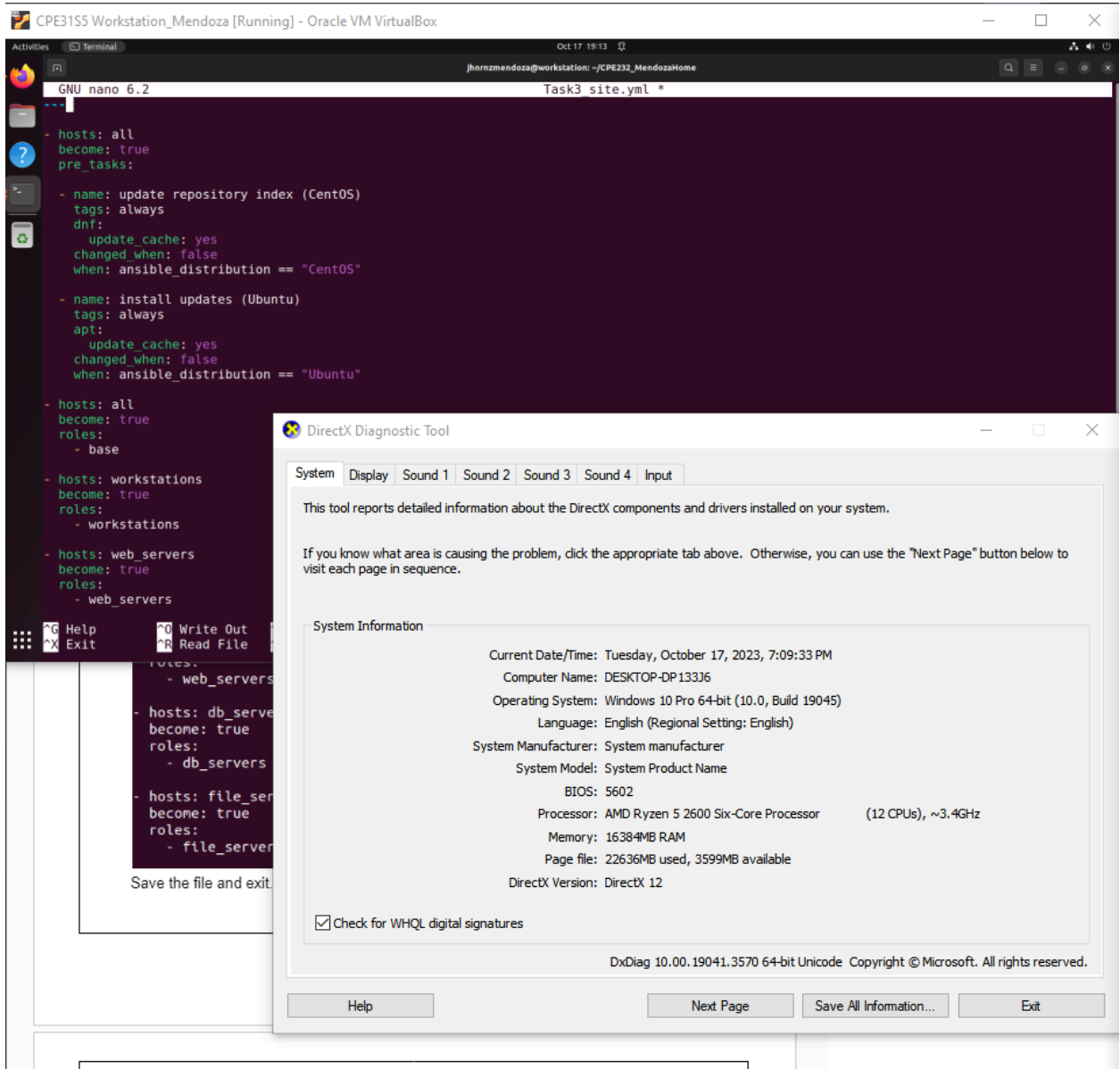
- hosts: db_servers
  become: true
  roles:
    - db_servers

- hosts: file_servers
  become: true
  roles:
    - file_servers
```

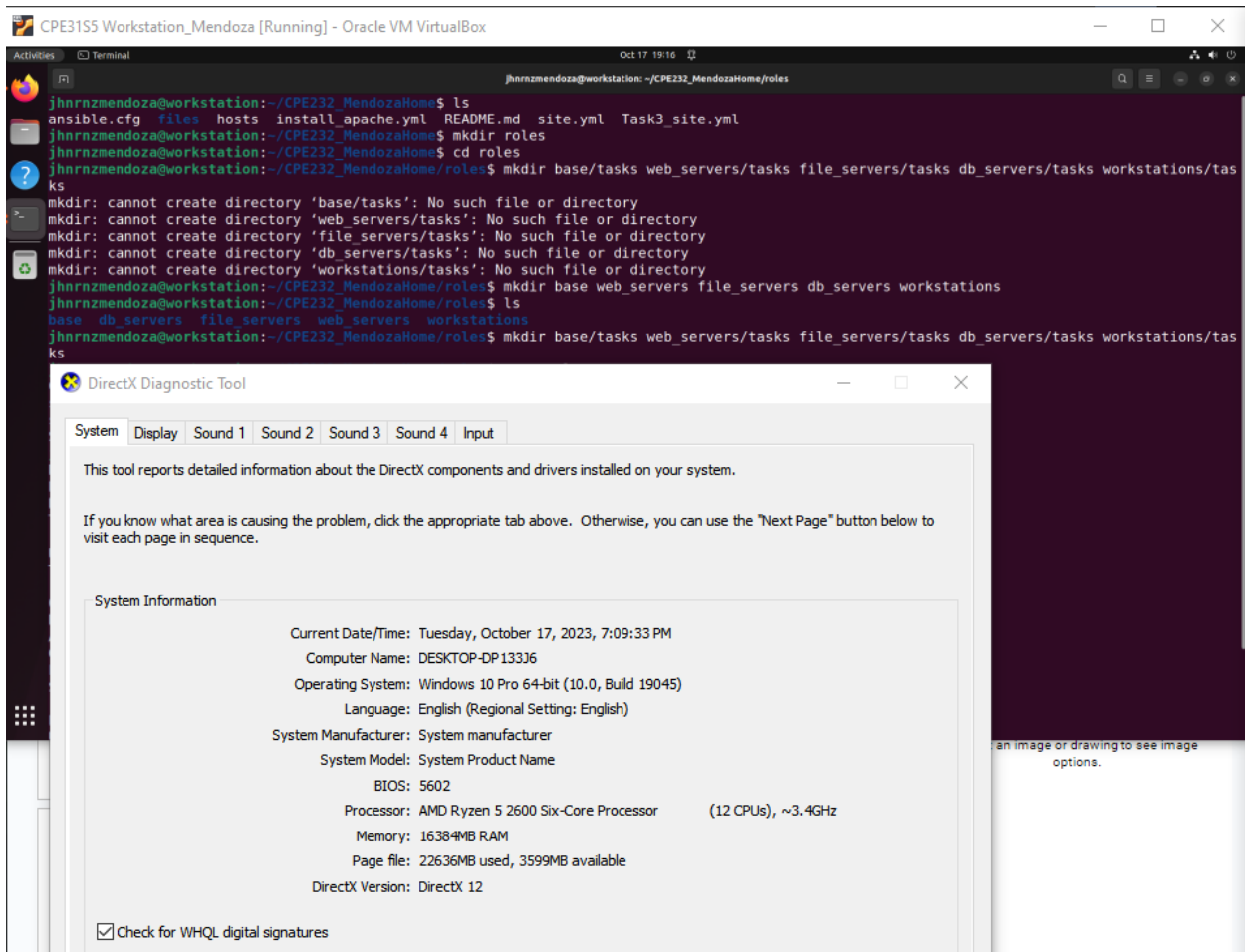
Save the file and exit.

Copying the contents of site.yml to a new file for backup.





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.



CPE3155 Workstation\_Mendoza [Running] - Oracle VM VirtualBox

Activities Terminal Oct 17 19:17 jhnrnmendoza@workstation: ~/CPE232\_MendozaHome/roles

```
jhnrnmendoza@workstation:~/CPE232_MendozaHome/roles$ ls -r roles
ls: cannot access 'roles': No such file or directory
jhnrnmendoza@workstation:~/CPE232_MendozaHome/roles$ ls -r *
workstations:
tasks
web_servers:
tasks
file_servers:
tasks
db_servers:
tasks
base:
tasks
jhnrnmendoza@workstation:~/CPE232_MendozaHome/roles$
```

DirectX Diagnostic Tool

System Display Sound 1 Sound 2 Sound 3 Sound 4 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, October 17, 2023, 7:09:33 PM  
Computer Name: DESKTOP-DP133J6  
Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)  
Language: English (Regional Setting: English)  
System Manufacturer: System manufacturer  
System Model: System Product Name  
BIOS: 5602  
Processor: AMD Ryzen 5 2600 Six-Core Processor (12 CPUs), ~3.4GHz  
Memory: 16384MB RAM  
Page file: 22636MB used, 3599MB available  
DirectX Version: DirectX 12

☒ Check for WHQL digital signatures

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

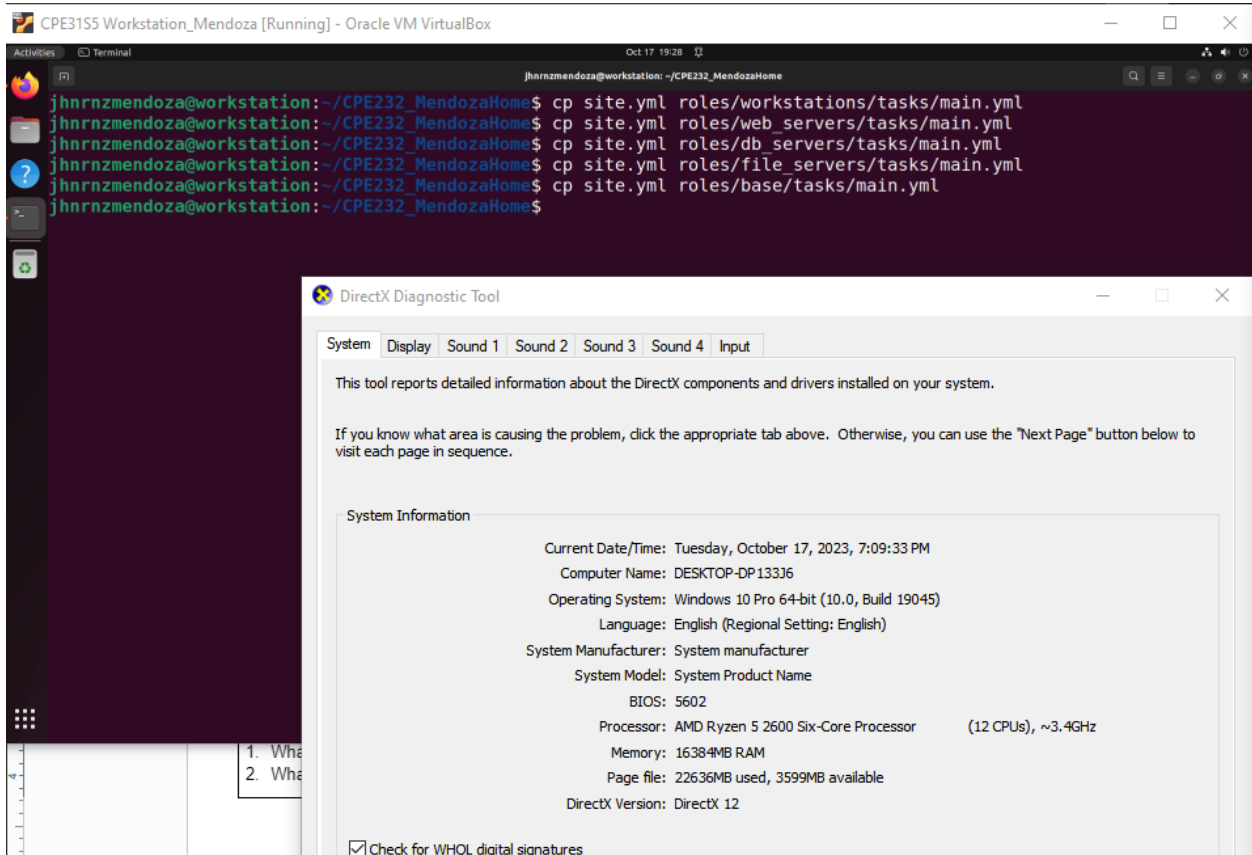
Help Next Page Save All Information... Exit



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.

Manually going to each directory and copy and pasting the contents of the site.yml playbook is hassle and inefficient. Therefore, I have used the cp command to copy the old site.yml to main.yml on each directory/tasks.

Here is the command that I have used:



The screenshot shows a terminal window titled "CPE3155 Workstation\_Mendoza [Running] - Oracle VM VirtualBox" with a dark purple background. The terminal displays the following commands and output:

```
jhnrmendoza@workstation: ~/CPE232_MendozaHome$ cp site.yml roles/workstations/tasks/main.yml
jhnrmendoza@workstation: ~/CPE232_MendozaHome$ cp site.yml roles/web_servers/tasks/main.yml
jhnrmendoza@workstation: ~/CPE232_MendozaHome$ cp site.yml roles/db_servers/tasks/main.yml
jhnrmendoza@workstation: ~/CPE232_MendozaHome$ cp site.yml roles/file_servers/tasks/main.yml
jhnrmendoza@workstation: ~/CPE232_MendozaHome$ cp site.yml roles/base/tasks/main.yml
jhnrmendoza@workstation: ~/CPE232_MendozaHome$
```

Overlaid on the terminal is the "DirectX Diagnostic Tool" window. It has tabs for "System", "Display", "Sound 1", "Sound 2", "Sound 3", "Sound 4", and "Input". The "System" tab is selected, showing the following information:

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.

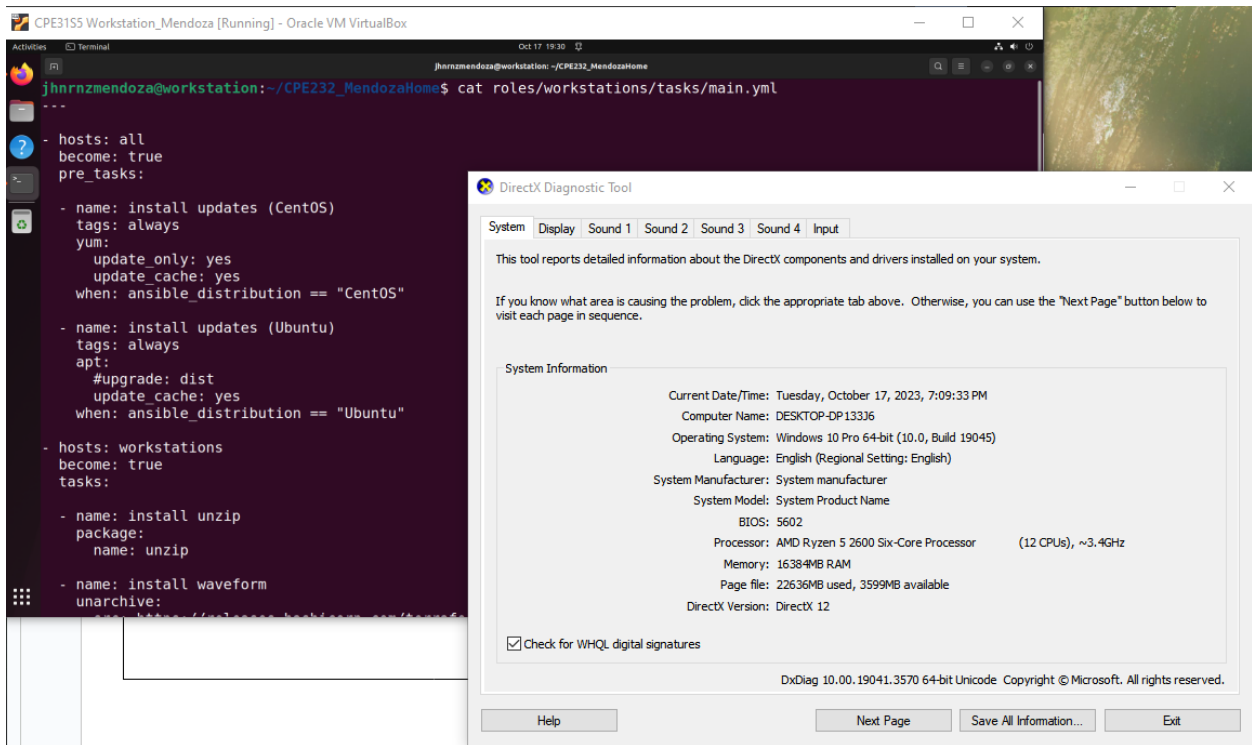
**System Information**

- Current Date/Time: Tuesday, October 17, 2023, 7:09:33 PM
- Computer Name: DESKTOP-DP133J6
- Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)
- Language: English (Regional Setting: English)
- System Manufacturer: System manufacturer
- System Model: System Product Name
- BIOS: 5602
- Processor: AMD Ryzen 5 2600 Six-Core Processor (12 CPUs), ~3.4GHz
- Memory: 16384MB RAM
- Page file: 22636MB used, 3599MB available
- DirectX Version: DirectX 12

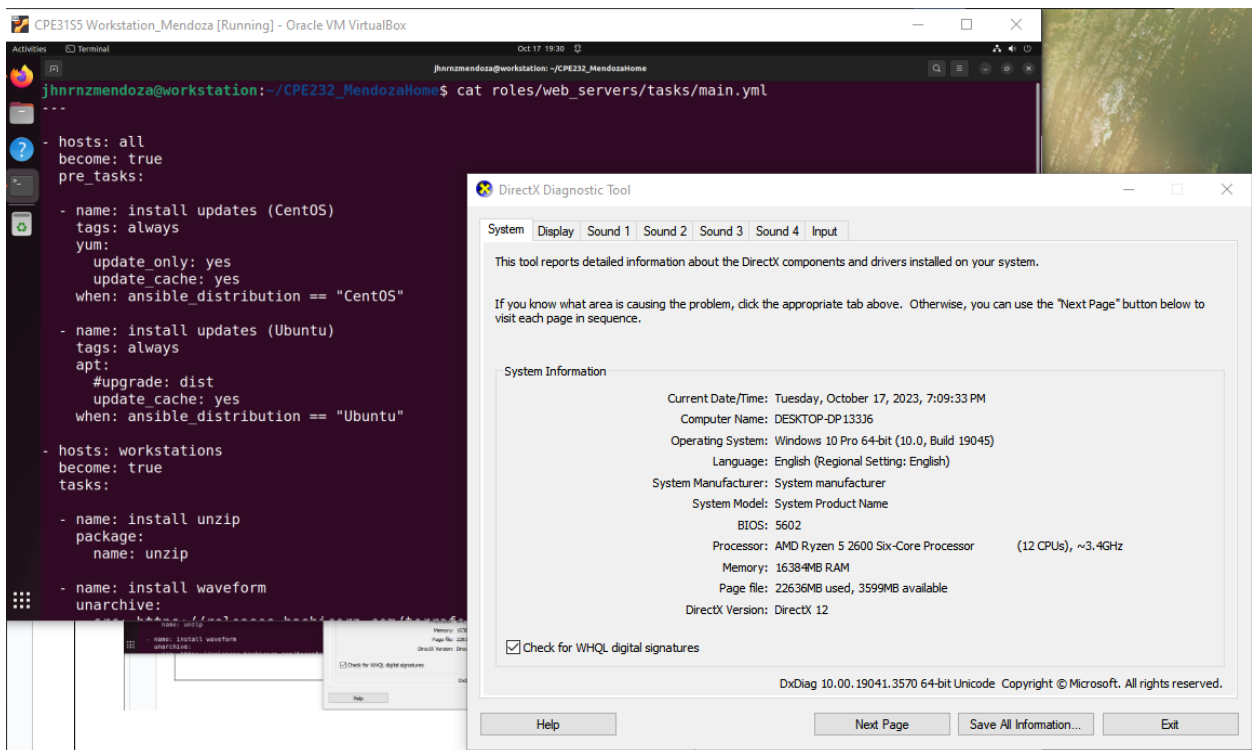
☒ Check for WHQL digital signatures

Checking if the new main.yml files are added with the appropriate contents.

## Directory: roles/workstations/tasks



## Directory: roles/web\_servers/tasks



## Directory: roles/db\_servers/tasks

The screenshot shows a terminal window with the following Ansible tasks:

```
---
- hosts: all
  become: true
  pre_tasks:
    - name: install updates (CentOS)
      tags: always
      yum:
        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

    - name: install waveform
      unarchive:
```

Overlaid on the terminal is the DirectX Diagnostic Tool window, which displays system information:

- Current Date/Time: Tuesday, October 17, 2023, 7:09:33 PM
- Computer Name: DESKTOP-DP133J6
- Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)
- Language: English (Regional Setting: English)
- System Manufacturer: System manufacturer
- System Model: System Product Name
- BIOS: 5602
- Processor: AMD Ryzen 5 2600 Six-Core Processor (12 CPUs), ~3.4GHz
- Memory: 16384MB RAM
- Page file: 22636MB used, 3599MB available
- DirectX Version: DirectX 12

At the bottom of the DirectX Diagnostic Tool window, there is a checkbox for "Check for WHQL digital signatures" and buttons for "Help", "Next Page", "Save All Information...", and "Exit".

## Directory: roles/file\_servers/tasks

The screenshot shows a terminal window with the following Ansible tasks:

```
---
- hosts: all
  become: true
  pre_tasks:
    - name: install updates (CentOS)
      tags: always
      yum:
        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

    - name: install waveform
      unarchive:
```

Overlaid on the terminal is the DirectX Diagnostic Tool window, which displays system information:

- Current Date/Time: Tuesday, October 17, 2023, 7:09:33 PM
- Computer Name: DESKTOP-DP133J6
- Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)
- Language: English (Regional Setting: English)
- System Manufacturer: System manufacturer
- System Model: System Product Name
- BIOS: 5602
- Processor: AMD Ryzen 5 2600 Six-Core Processor (12 CPUs), ~3.4GHz
- Memory: 16384MB RAM
- Page file: 22636MB used, 3599MB available
- DirectX Version: DirectX 12

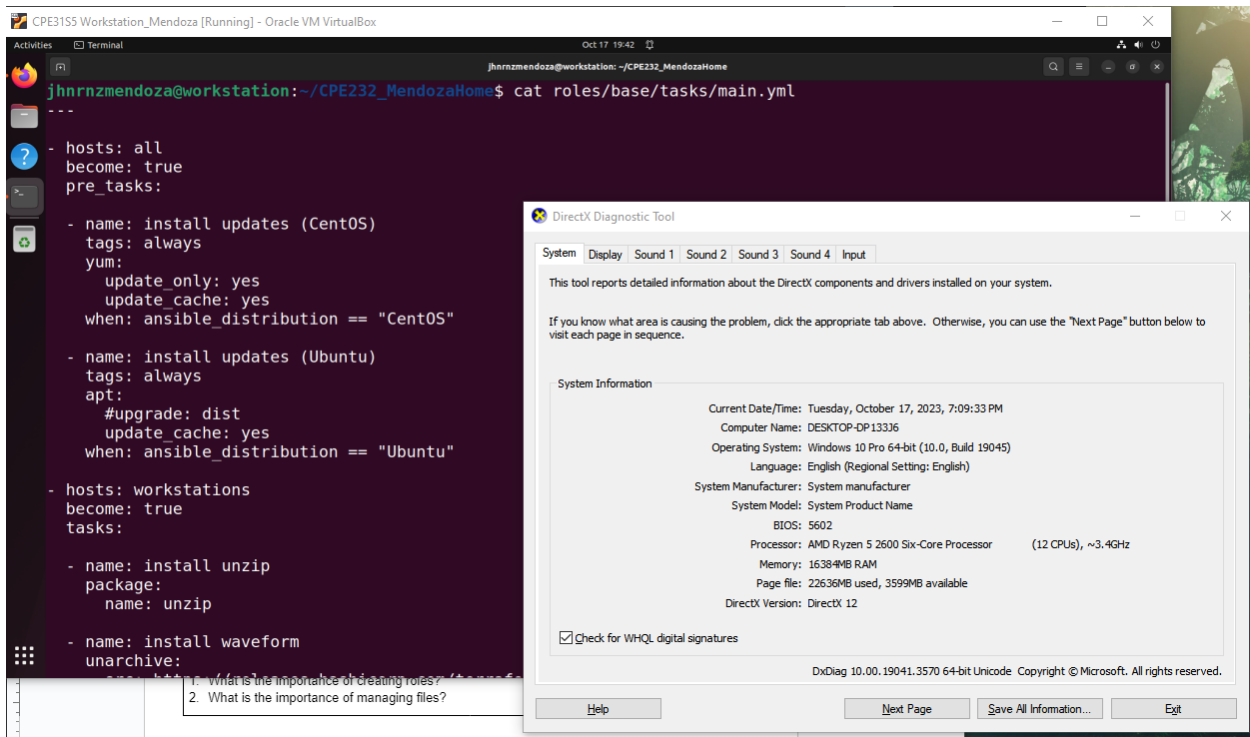
At the bottom of the DirectX Diagnostic Tool window, there is a checkbox for "Check for WHQL digital signatures" and buttons for "Help", "Next Page", "Save All Information...", and "Exit".

Below the terminal window, there is a section titled "Reflections:" with the following text:

Answer the following:

1. What is the importance of creating roles?
2. What is the importance of managing files?

## Directory: roles/base/tasks



### Observation:

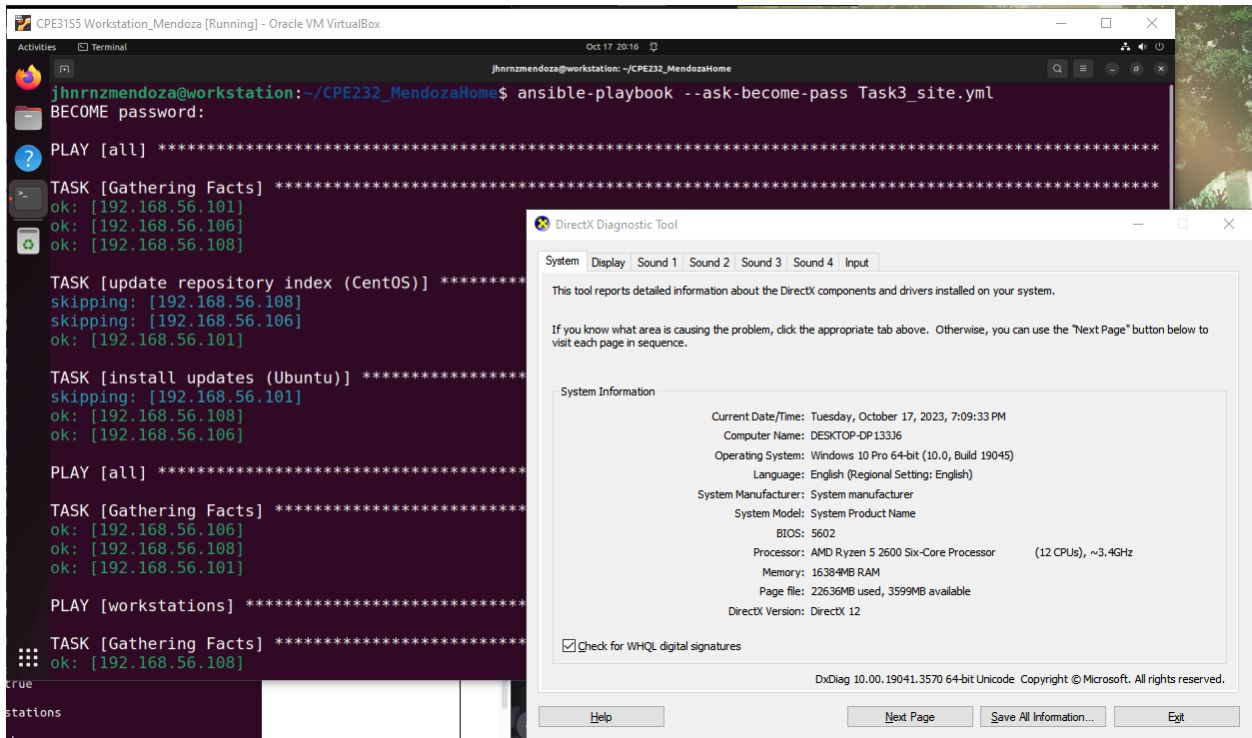
- In this step, I have used the `cp` command in order to copy the old `site.yml` to be the contents of the `main.yml`. In my case, the old `site.yml` is stored on the `Task3_site.yml`. I have concatenated the contents of each `main.yml` under each `roles/directory`.
- As observed on each query, the old `site.yml` was successfully copied to each `main.yml`.

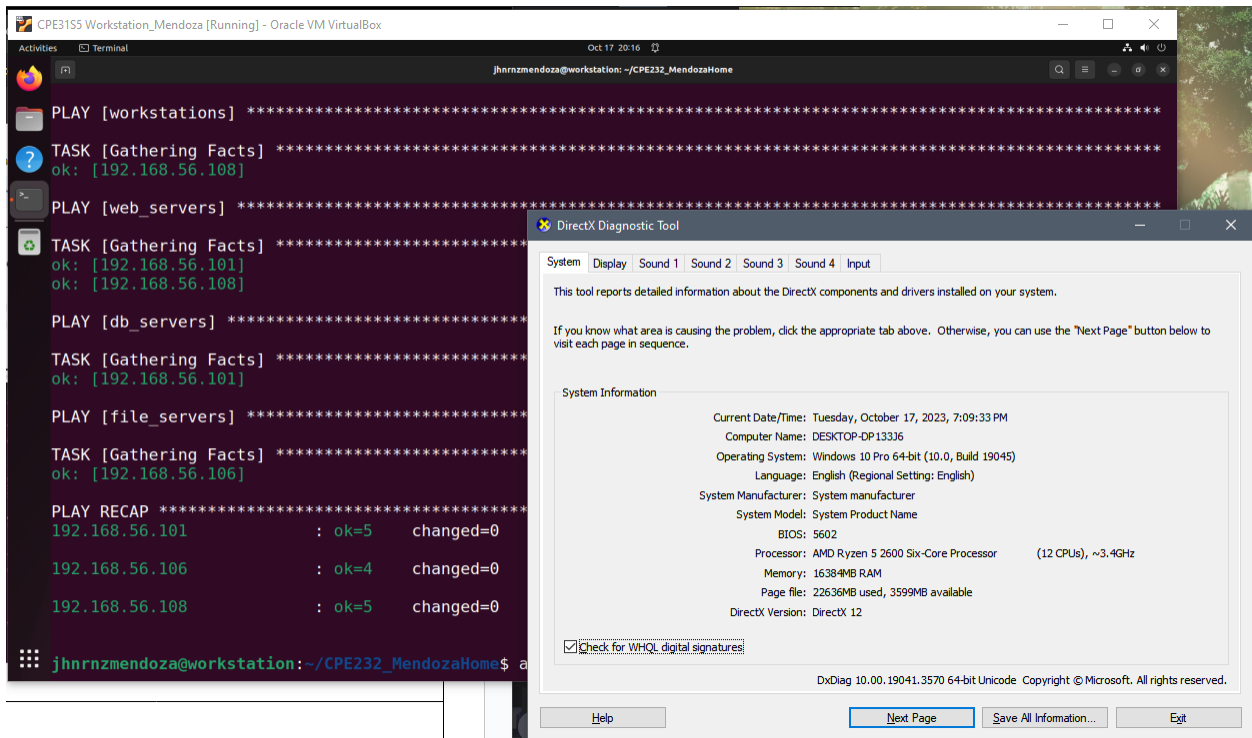
#### 4. Run the site.yml playbook and describe the output.

The original code has errors when executed, therefore I have changed some of the syntaxes such as:

- Changing the package manager from `yum` from `dnf`
- Changing the syntax on defining the roles.

Output:





## Observation:

- After changing some of the syntax, the playbook was successfully executed. We can observe that each of the directory under the roles directory has been executed along the run playbook.
- In my opinion, the roles specifically run the playbook file within its directory. For instance, the main.yml in the directory role/workstations/tasks/, when we run the new site.yml, the workstation directory will now read the playbook main.yml and show which will be the target remote server.
- As observed, the play web\_servers targeted the remote server 1 and CentOS since these 2 servers will be needed in that specific play, which is why we cannot see the file server 3 to be "skipped".

**Reflections:**

Answer the following:

1. What is the importance of creating roles?

- Creating roles in ansible makes playbook run much easier to understand since it will display the target remote servers that will be used in the run. As observed on the previous playbook run, it was able to gather which remote servers will be used specifically without traversing every server and skipping them.
- Essentially, ansible roles also help in organizing the structure of the ansible playbooks and executing them. By having roles, the playbook task only targets specific nodes. In addition, it is a way to group

2. What is the importance of managing files?

- By managing files, we would have an organized structure of our directories. As observed in this activity, each ansible role has a defined directory having the same playbook files under it. This makes the loading of the variables and needed files efficient. We would also be able to debug the errors easily since the directories are properly organized. In addition, the playbook would be able to identify the remote nodes that is needed in each playbook task.