

Name: Andreu John L. Salvador	Date Performed: 10/12/2023
Course/Section: CPE31S5	Date Submitted: 10/12/2023
Instructor: Engr. Roman Richard	Semester and SY: 1st 2023-2024
Activity 15: OpenStack Installation (Neutron, Horizon, Cinder)	
1. Objectives	
Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).	
2. Intended Learning Outcomes	
<ol style="list-style-type: none"> 1. Analyze the advantages and disadvantages of cloud services 2. Evaluate different Cloud deployment and service models 3. Create a workflow to install and configure OpenStack base services using Ansible as documentation and execution. 	
3. Resources	
<p>Oracle VirtualBox (Hypervisor)</p> <p>1x Ubuntu VM or Centos VM</p>	
4. Tasks	
<ol style="list-style-type: none"> 1. Create a new repository for this activity. 2. Create a playbook that converts the steps in the following items in https://docs.openstack.org/install-guide/ <ol style="list-style-type: none"> a. Neutron b. Horizon c. Cinder d. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in the Inventory file. e. Add, commit and push it to your GitHub repo. 	
5. Output (screenshots and explanations)	

1. Creating The repository in github

The screenshot shows the GitHub repository page for 'Hands-on-Activity-15-' by user 'qajlsalvador'. The repository is public and has 1 branch and 0 tags. The README.md file is visible, showing the title 'Hands-on-Activity-15-'. A 'DirectX Diagnostic Tool' window is overlaid on the right side of the page, displaying system information.

qajlsalvador / Hands-on-Activity-15-

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Hands-on-Activity-15- Public

Pin Unwatch 1 Fork 0 Star 0

main 1 branch 0 tags

Go to file Add file <> Code About

qajlsalvador Initial commit

README.md Initial commit

README.md

Hands-on-Activity-15-

DirectX Diagnostic Tool

System Display Sound 1 Sound 2 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: Sunday, 10 December 2023, 8:44:20 pm
Computer Name: DESKTOP-BVLHQ70
Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)
Language: English (Regional Setting: English)
System Manufacturer: EMAXX TECHNOLOGY INC
System Model: EMX-A70FM2-HCafe
BIOS: 4.6.5
Processor: AMD A8-7600 Radeon R7, 10 Compute Cores 4C+6G (4 CPUs), ~3.1GHz
Memory: 8192MB RAM
Page file: 10059MB used, 5478MB available
DirectX Version: DirectX 12

2. Cloning the github repository to the workstation

The screenshot shows a terminal window with the command 'git clone git@github.com:qajlsalvador/Hands-on-Activity-15-.git' being executed. The output shows the repository being cloned into 'Hands-on-Activity-15-'. A 'DirectX Diagnostic Tool' window is overlaid on the right side of the terminal, displaying system information.

salvador@Workstation: ~/Hands-on-Activity-15-

```
salvador@Workstation:~$ git clone git@github.com:qajlsalvador/Hands-on-Activity-15-.git
Cloning into 'Hands-on-Activity-15-'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
salvador@Workstation:~$ cd Hands-on-Activity-15-
salvador@Workstation:~/Hands-on-Activity-15-$ ls
README.md
salvador@Workstation:~/Hands-on-Activity-15-$
```

DirectX Diagnostic Tool

System Display Sound 1 Sound 2 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

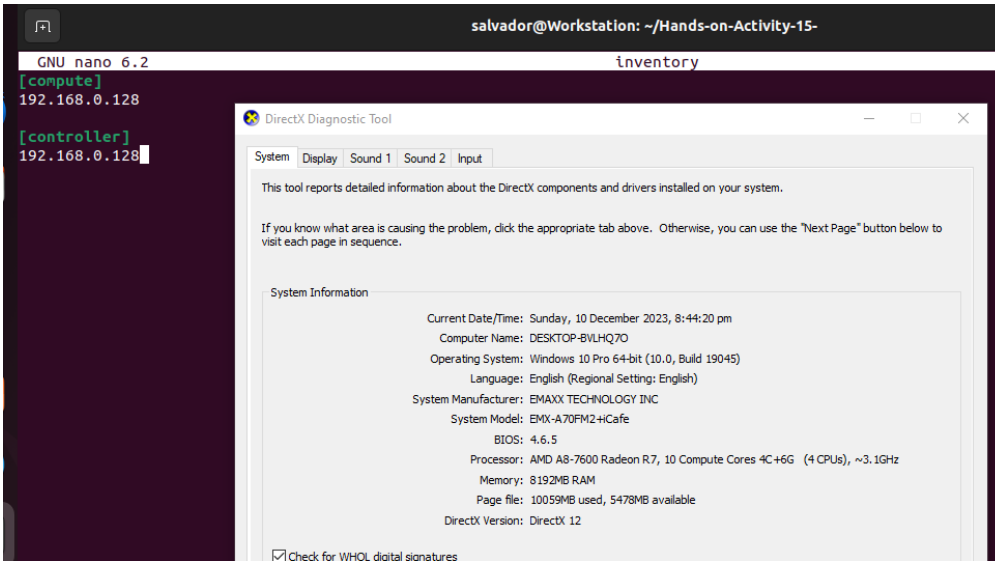
Current Date/Time: Sunday, 10 December 2023, 8:44:20 pm
Computer Name: DESKTOP-BVLHQ70
Operating System: Windows 10 Pro 64-bit (10.0, Build 19045)
Language: English (Regional Setting: English)
System Manufacturer: EMAXX TECHNOLOGY INC
System Model: EMX-A70FM2-HCafe
BIOS: 4.6.5
Processor: AMD A8-7600 Radeon R7, 10 Compute Cores 4C+6G (4 CPUs), ~3.1GHz
Memory: 8192MB RAM
Page file: 10059MB used, 5478MB available
DirectX Version: DirectX 12

☒ Check for WHQL digital signatures

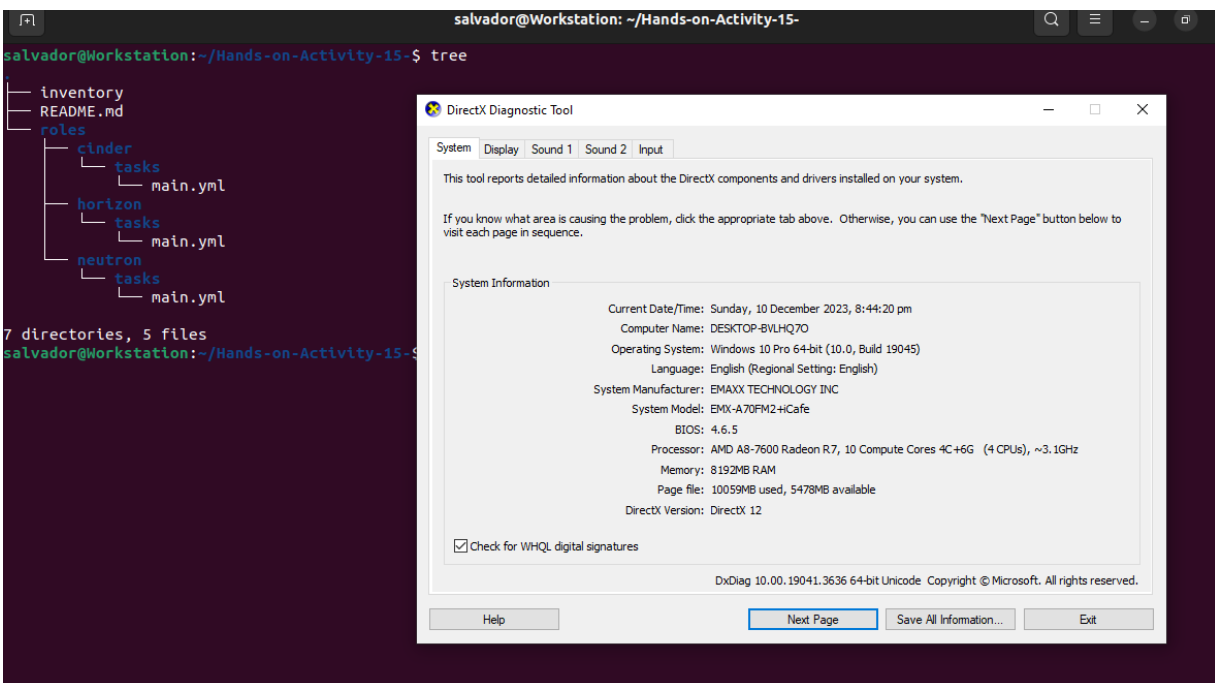
DxDiag 10.00.19041.3636 64-bit Unicode Copyright © Microsoft. All rights reserved.

Help Next Page Save All Information... Exit

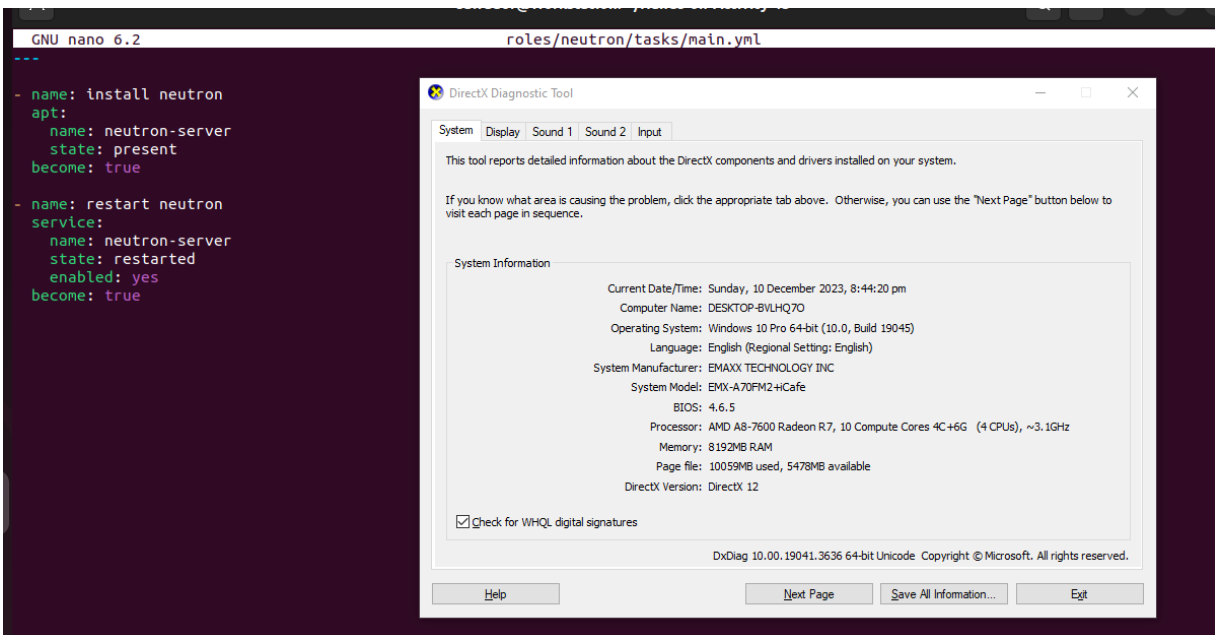
3. Create the inventory to store the ip address of the Virtual Machines you want to install the applications to



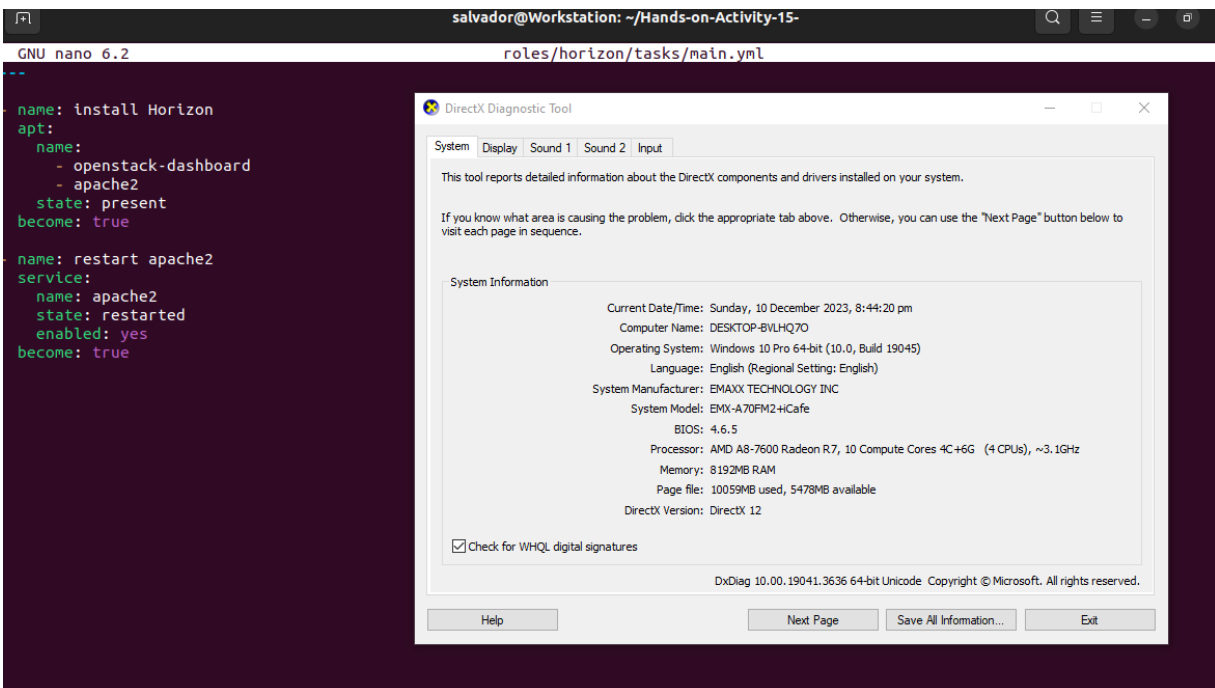
4. Creating the roles directory and creating the appropriate directory for each application inside the roles directory.



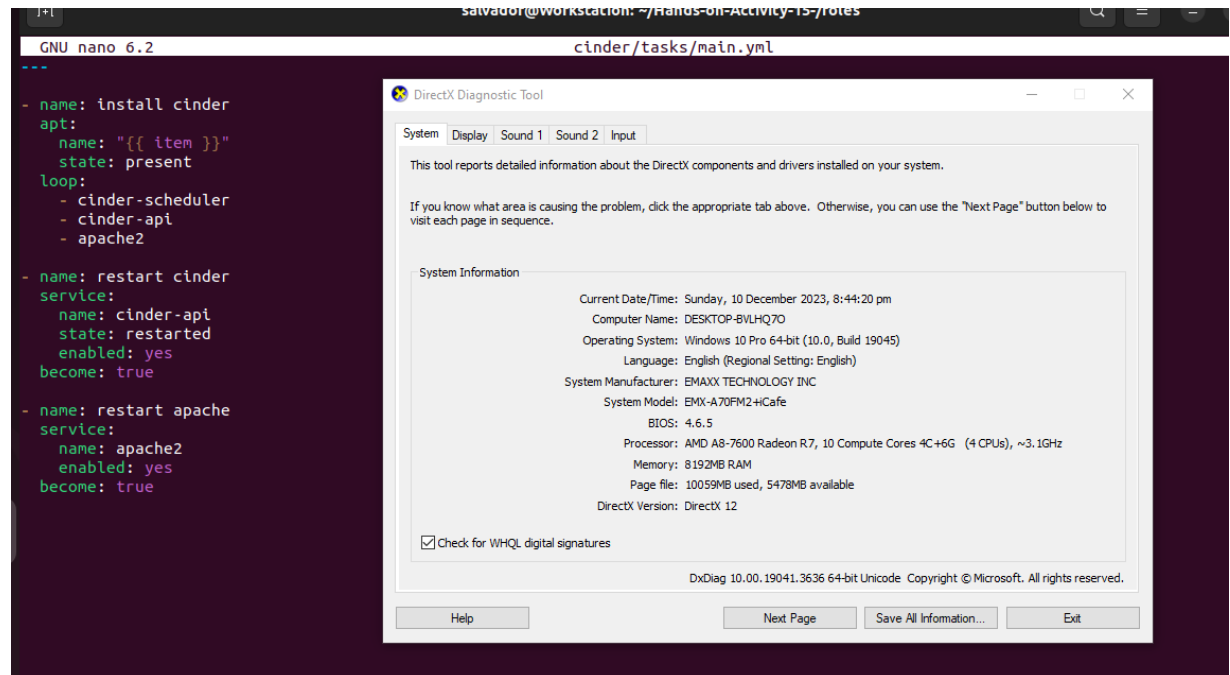
5. Naming main.yml the yaml file that will contain the codes that is needed in automating the installation of the said items
Neutron:



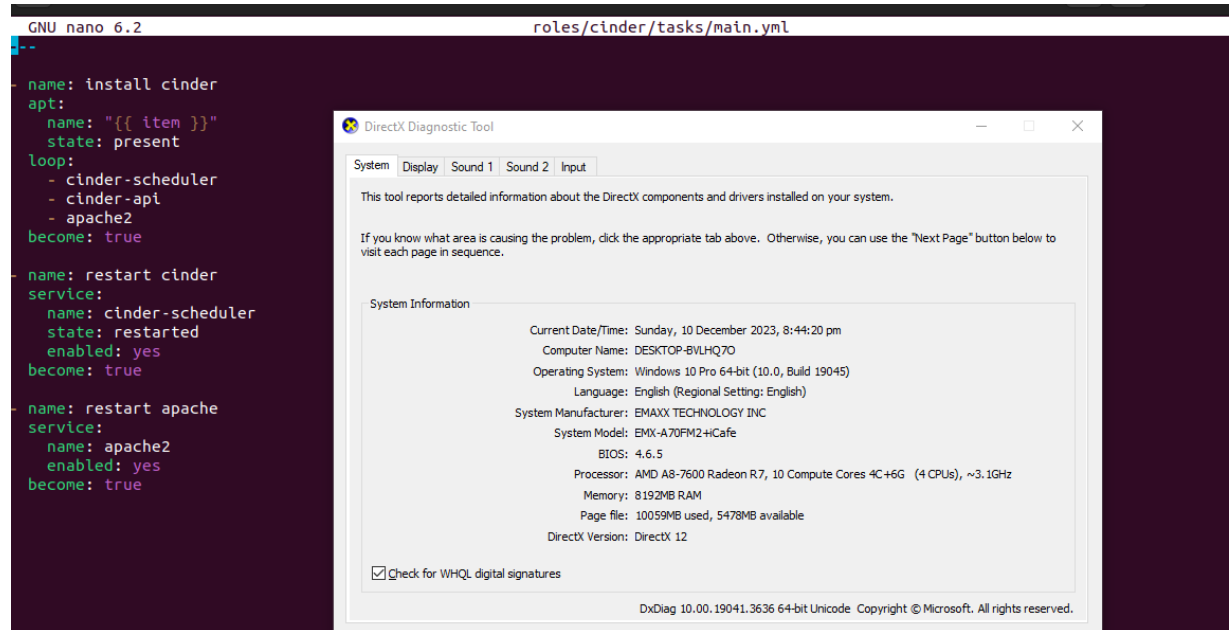
Horizon:



Cinder:



6. Creating the playbook to play the roles



7. Running the playbook

```
TASK [cinder : restart cinder] *****
changed: [192.168.0.128]

TASK [cinder : restart apache] *****
ok: [192.168.0.128]

PLAY [compute] *****

TASK [Gathering Facts] *****
ok: [192.168.0.128]

TASK [neutron : install neutron] *****
ok: [192.168.0.128]

TASK [neutron : restart neutron] *****
changed: [192.168.0.128]

TASK [horizon : install Horizon] *****
ok: [192.168.0.128]

TASK [horizon : restart apache2] *****
changed: [192.168.0.128]

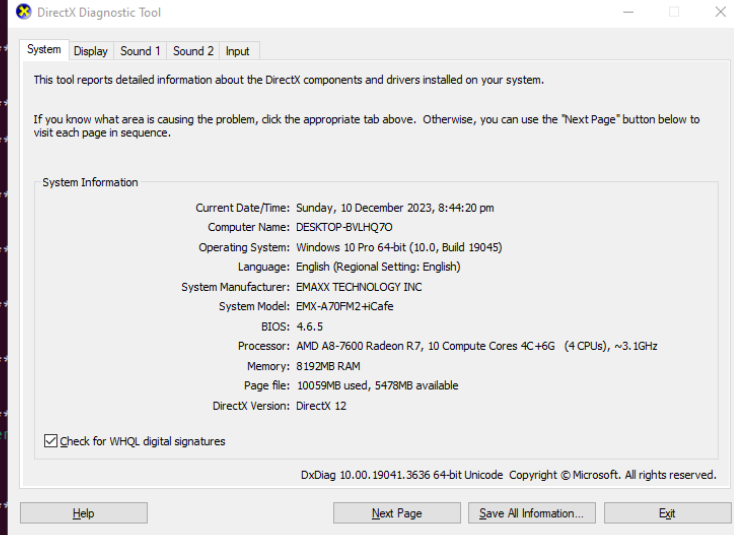
TASK [cinder : install cinder] *****
ok: [192.168.0.128] => (item=cinder-scheduler)
ok: [192.168.0.128] => (item=cinder-api)
ok: [192.168.0.128] => (item=apache2)

TASK [cinder : restart cinder] *****
changed: [192.168.0.128]

TASK [cinder : restart apache] *****
ok: [192.168.0.128]

PLAY RECAP *****
192.168.0.128      : ok=18   changed=7    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

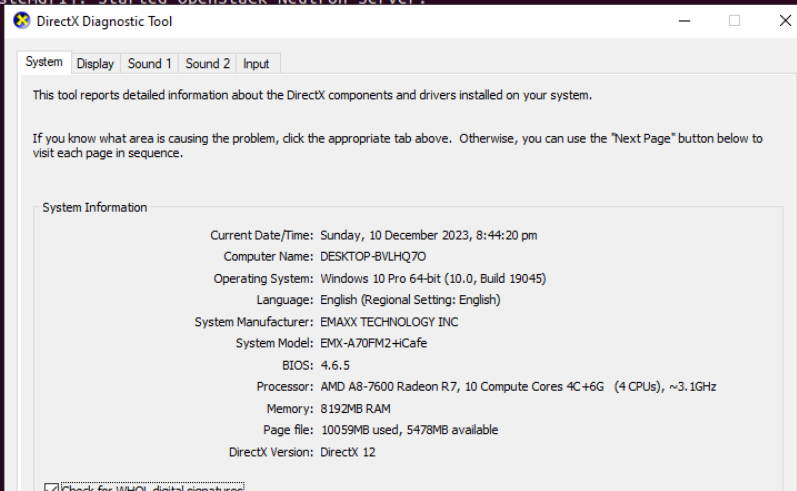
salvador@Workstation: ~/Hands-on-Activity-15 - $
```



8. Checking Neutron:

```
salvador@Server2:~$ systemctl status neutron-server
● neutron-server.service - OpenStack Neutron Server
   Loaded: loaded (/lib/systemd/system/neutron-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-12-10 13:50:19 UTC; 33s ago
     Docs: man:neutron-server(1)
  Main PID: 18296 (neutron-server)
    Tasks: 1 (limit: 2966)
   Memory: 125.1M
      CPU: 6.354s
   CGroup: /system.slice/neutron-server.service
           └─18296 /usr/bin/python3 /usr/bin/neutron-server --config-file=/etc/neutron/neutron.conf --config-file=/etc/

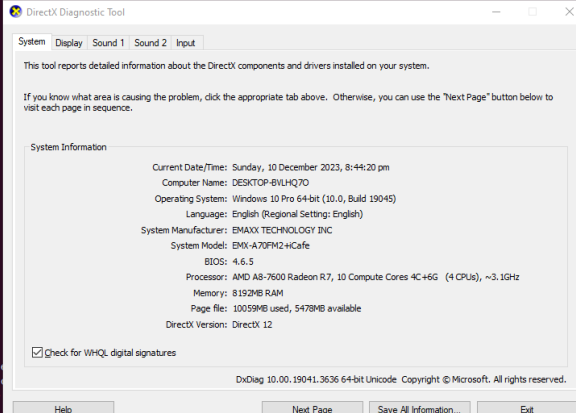
Dec 10 13:50:19 Server2 systemd[1]: Started OpenStack Neutron Server.
lines 1-12/12 (END)
```



Horizon:

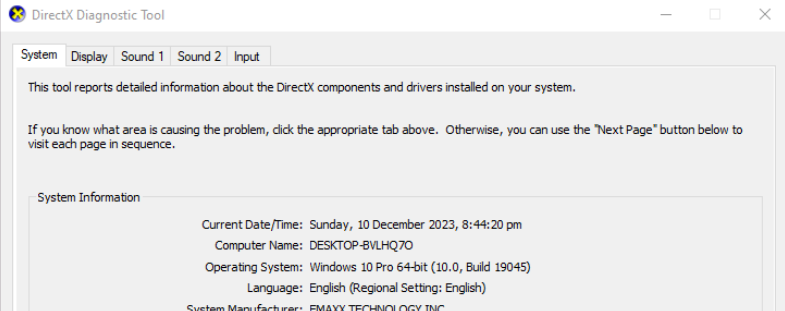
```
salvador@Server2:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-12-10 13:49:06 UTC; 2min 31s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 17915 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 17919 (apache2)
     Tasks: 85 (limit: 2966)
    Memory: 82.9M
       CPU: 1.064s
    CGroup: /system.slice/apache2.service
            └─17919 /usr/sbin/apache2 -k start
              └─17925 " (wsgl:cinder-wsgl" -k start
                └─17926 " (wsgl:cinder-wsgl" -k start
                  └─17927 " (wsgl:cinder-wsgl" -k start
                    └─17928 " (wsgl:cinder-wsgl" -k start
                      └─17929 " (wsgl:cinder-wsgl" -k start
                        └─17938 " (wsgl:horizon) " -k start
                          └─17942 " (wsgl:horizon) " -k start
                            └─17943 " (wsgl:horizon) " -k start
                              └─17952 " (wsgl:keystone-pu" -k start
                                └─17953 " (wsgl:keystone-pu" -k start
                                  └─17954 " (wsgl:keystone-pu" -k start
                                    └─17956 " (wsgl:keystone-pu" -k start
                                      └─17959 " (wsgl:keystone-pu" -k start
                                        └─17960 /usr/sbin/apache2 -k start
                                          └─17961 /usr/sbin/apache2 -k start
                                            └─17962 /usr/sbin/apache2 -k start
                                              └─17963 /usr/sbin/apache2 -k start
                                                └─17964 /usr/sbin/apache2 -k start

Dec 10 13:49:06 Server2 systemd[1]: Starting The Apache
Dec 10 13:49:06 Server2 apachectl[17918]: AH00558: apa
Dec 10 13:49:06 Server2 systemd[1]: Started The Apache
lines 1-33/33 (END)
```



The screenshot shows the DirectX Diagnostic Tool window. It has tabs for System, Display, Sound 1, Sound 2, and Input. The System tab is selected. The text inside says: "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." Below this is a section titled "System Information" with the following details: Current Date/Time: Sunday, 10 December 2023, 8:44:20 pm; Computer Name: DESKTOP-BVLHQ70; Operating System: Windows 10 Pro 64-bit (10.0, Build 19045); Language: English (Regional Setting: English); System Manufacturer: EMAXX TECHNOLOGY INC; System Model: EMX-A70FM2-iCafe; BIOS: 4.6.5; Processor: AMD A8-7600 Radeon R7, 10 Compute Cores 4C+6G (4 CPUs), ~3.1GHz; Memory: 8192MB RAM; Page File: 10059MB used, 5478MB available; DirectX Version: DirectX 12. At the bottom, there is a checkbox for "Check for WHQL digital signatures" which is checked. The footer says "DxDiag 10.00.19041.3636 64-bit Unicode Copyright © Microsoft. All rights reserved." and there are buttons for Help, Next Page, Save All Information, and Exit.

```
salvador@Server2:~$ dpkg -l openstack-dashboard
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/half-f-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name Version Architecture Description
+++-----+-----+-----+-----+
ii openstack-dashboard 4:22.1.1-0ubuntu1 all Django web interface for OpenStack
salvador@Server2:~$
```

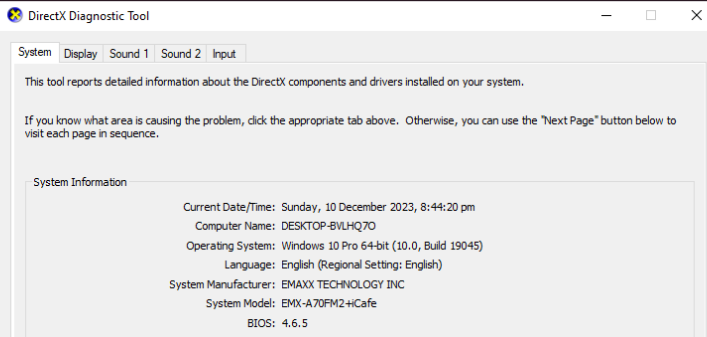


The screenshot shows the DirectX Diagnostic Tool window. It has tabs for System, Display, Sound 1, Sound 2, and Input. The System tab is selected. The text inside says: "This tool reports detailed information about the DirectX components and drivers installed on your system. If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the 'Next Page' button below to visit each page in sequence." Below this is a section titled "System Information" with the following details: Current Date/Time: Sunday, 10 December 2023, 8:44:20 pm; Computer Name: DESKTOP-BVLHQ70; Operating System: Windows 10 Pro 64-bit (10.0, Build 19045); Language: English (Regional Setting: English); System Manufacturer: EMAXX TECHNOLOGY INC. At the bottom, there is a checkbox for "Check for WHQL digital signatures" which is checked. The footer says "DxDiag 10.00.19041.3636 64-bit Unicode Copyright © Microsoft. All rights reserved." and there are buttons for Help, Next Page, Save All Information, and Exit.

Cinder:

```
salvador@Server2:~$ systemctl status cinder-scheduler
● cinder-scheduler.service - OpenStack Cinder Scheduler
   Loaded: loaded (/lib/systemd/system/cinder-scheduler.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-12-10 13:52:39 UTC; 3s ago
     Docs: man:cinder-scheduler(1)
   Main PID: 18689 (cinder-schedule)
     Tasks: 1 (limit: 2966)
    Memory: 24.2M
       CPU: 750ms
    CGroup: /system.slice/cinder-scheduler.service
            └─18689 /usr/bin/python3 /usr/bin/cinder-scheduler --config-file=/etc/cinder/cinder.conf --log-file=/var/log/cinder/ci

Dec 10 13:52:39 Server2 systemd[1]: Started OpenStack Cinder Scheduler.
lines 1-12/12 (END)
```

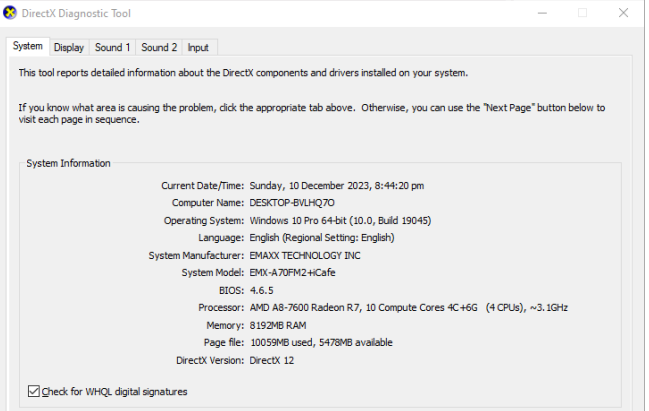


The screenshot shows the DirectX Diagnostic Tool window. It has tabs for System, Display, Sound 1, Sound 2, and Input. The System tab is selected. The text inside says: "This tool reports detailed information about the DirectX components and drivers installed on your system. If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the 'Next Page' button below to visit each page in sequence." Below this is a section titled "System Information" with the following details: Current Date/Time: Sunday, 10 December 2023, 8:44:20 pm; Computer Name: DESKTOP-BVLHQ70; Operating System: Windows 10 Pro 64-bit (10.0, Build 19045); Language: English (Regional Setting: English); System Manufacturer: EMAXX TECHNOLOGY INC; System Model: EMX-A70FM2-iCafe; BIOS: 4.6.5; Processor: AMD A8-7600 Radeon R7, 10 Compute Cores 4C+6G (4 CPUs), ~3.1GHz. At the bottom, there is a checkbox for "Check for WHQL digital signatures" which is checked. The footer says "DxDiag 10.00.19041.3636 64-bit Unicode Copyright © Microsoft. All rights reserved." and there are buttons for Help, Next Page, Save All Information, and Exit.

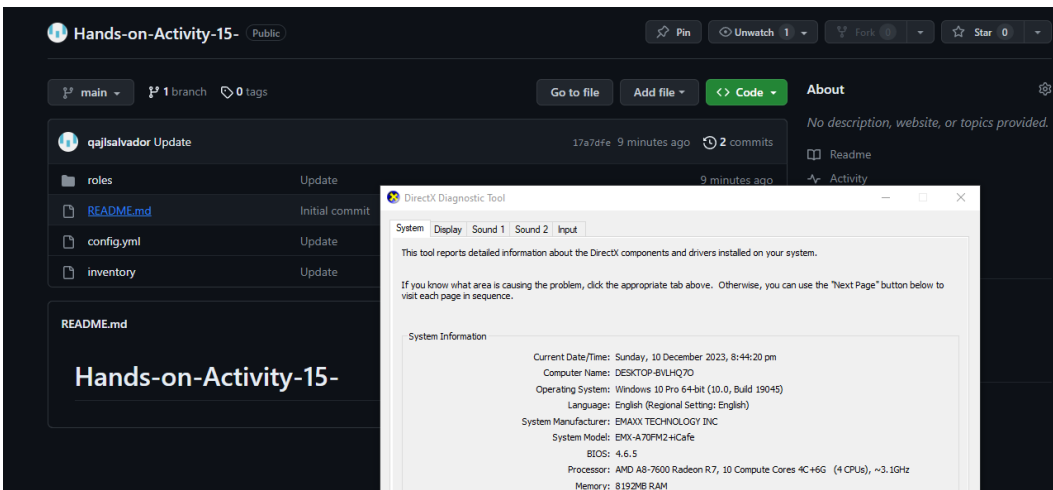
9. Pushing everything in the github repository

```
connecton to 172.168.0.128 closed.
salvador@Workstation: ~/Hands-on-Activity-15 $ sudo nano roles/cinder/tasks/main.yml
salvador@Workstation: ~/Hands-on-Activity-15 $ sudo nano roles/neutron/tasks/main.yml
salvador@Workstation: ~/Hands-on-Activity-15 $ sudo nano roles/horizon/tasks/main.yml
salvador@Workstation: ~/Hands-on-Activity-15 $ git add *
salvador@Workstation: ~/Hands-on-Activity-15 $ git commit -m "Update"
[main 17a7dfe] Update
5 files changed, 81 insertions(+)
create mode 100644 config.yml
create mode 100644 inventory
create mode 100644 roles/cinder/tasks/main.yml
create mode 100644 roles/horizon/tasks/main.yml
create mode 100644 roles/neutron/tasks/main.yml
salvador@Workstation: ~/Hands-on-Activity-15 $ git push origin main
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (14/14), 1.28 KiB | 262.00 KiB/s, done.
Total 14 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qajlsalvador/Hands-on-Activity-15-.git
941e16d..17a7dfe  main -> main
salvador@Workstation: ~/Hands-on-Activity-15 $ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
salvador@Workstation: ~/Hands-on-Activity-15 $
```



The DirectX Diagnostic Tool window displays system information. The 'System' tab is selected. The tool reports detailed information about the DirectX components and drivers installed on the system. The system information includes: Current Date/Time: Sunday, 10 December 2023, 8:44:20 pm; Computer Name: DESKTOP-BVLHQ70; Operating System: Windows 10 Pro 64-bit (10.0, Build 19045); Language: English (Regional Setting: English); System Manufacturer: EMAXX TECHNOLOGY INC; System Model: EMX-A70FM2-HCafe; BIOS: 4.6.5; Processor: AMD A8-7600 Radeon R7, 10 Compute Cores 4C+6G (4 CPUs), ~3.1GHz; Memory: 8192MB RAM; Page file: 10059MB used, 5478MB available; DirectX Version: DirectX 12. There is a checkbox for 'Check for WHQL digital signatures' which is checked.



Github link: <https://github.com/qajlsalvador/Hands-on-Activity-15-.git>

Reflections:

Answer the following:

1. Describe Neutron, Horizon and Cinder services

These are openstack services that can be used to support a user when working with networks. An example is Neutron which provides the network connectivity in virtual environment in the cloud. Horizon provides a web-based interface for the user just like how nova and keystone does, and lastly Cinder is also an open-source software that is to help a user to create and manage services using data storage to the cloud or cloud computing applications. It is also known as OpenStack Block Storage project.

Conclusions:

Throughout making this activity, we are able to successfully install the necessary applications for our openstack application. Neutron, Horizon, and Cinder are the services that is needed to be installed using the ansible playbook. With the use

of the ansible playbook the installation was made simple and quick, with just a single run of a play, the installation of the 3 services was achieved. In the end the services was checked to confirm the installation. The state of the services upon checking was active therefore the goal finished and the services can now be use by the user.