Name: Shaniah Rose Hope M. Sumaoang	Date Performed: October 24, 2023
Course/Section: CPE 232-CPE31S5	Date Submitted: October 25, 2023
Instructor: Engr. Roman Richard	Semester and SY: 1st Sem 23-24

Activity 9: Install, Configure, and Manage Performance Monitoring tools

1. Objectives

Create and design a workflow that installs, configure and manage enterprise performance tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Discussion

Performance monitoring is a type of monitoring tool that identifies current resource consumption of the workload, in this page we will discuss multiple performance monitoring tool.

Prometheus

Prometheus fundamentally stores all data as timeseries: streams of timestamped values belonging to the same metric and the same set of labeled dimensions. Besides stored time series, Prometheus may generate temporary derived time series as the result of queries. Source: Prometheus - Monitoring system & time series database

Cacti

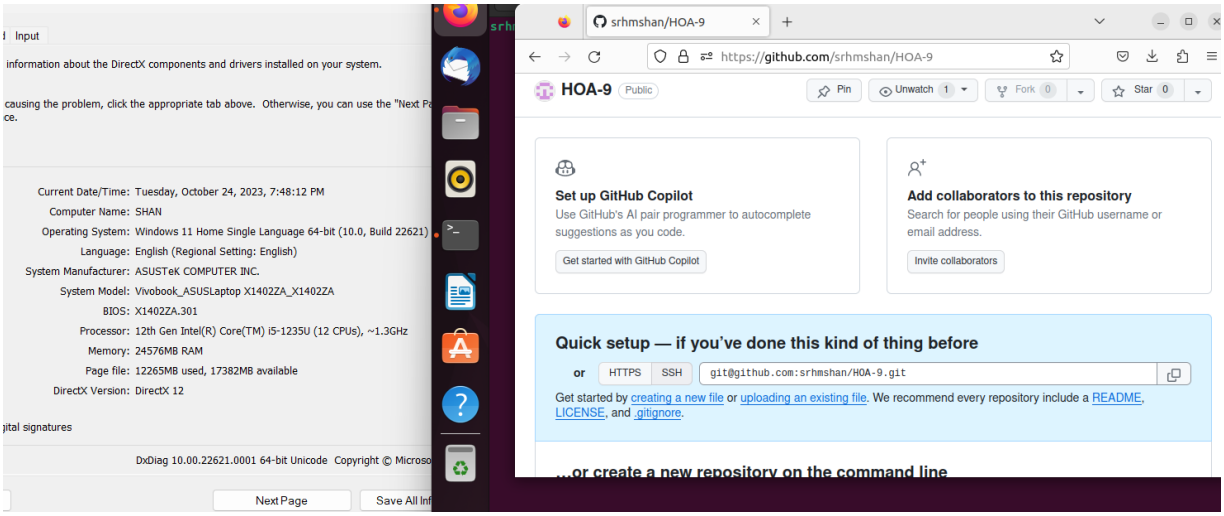
Cacti is a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality. Cacti provides a fast poller, advanced graph templating, multiple data acquisition methods, and user management features out of the box. All of this is wrapped in an intuitive, easy to use interface that makes sense for LAN-sized installations up to complex networks with thousands of devices. Source: Cacti® - The Complete RRDTool-based Graphing Solution

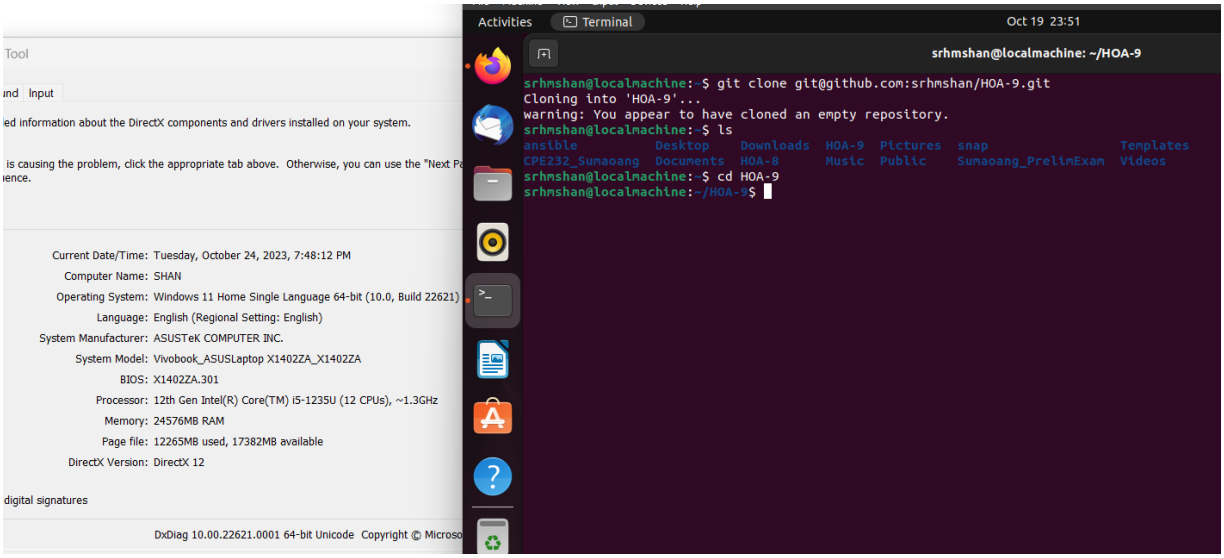
3. Tasks

1. Create a playbook that installs Prometheus in both Ubuntu and CentOS. Apply the concept of creating roles.
2. Describe how you did step 1. (Provide screenshots and explanations in your report. Make your report detailed such that it will look like a manual.)
3. Show an output of the installed Prometheus for both Ubuntu and CentOS.
4. Make sure to create a new repository in GitHub for this activity.

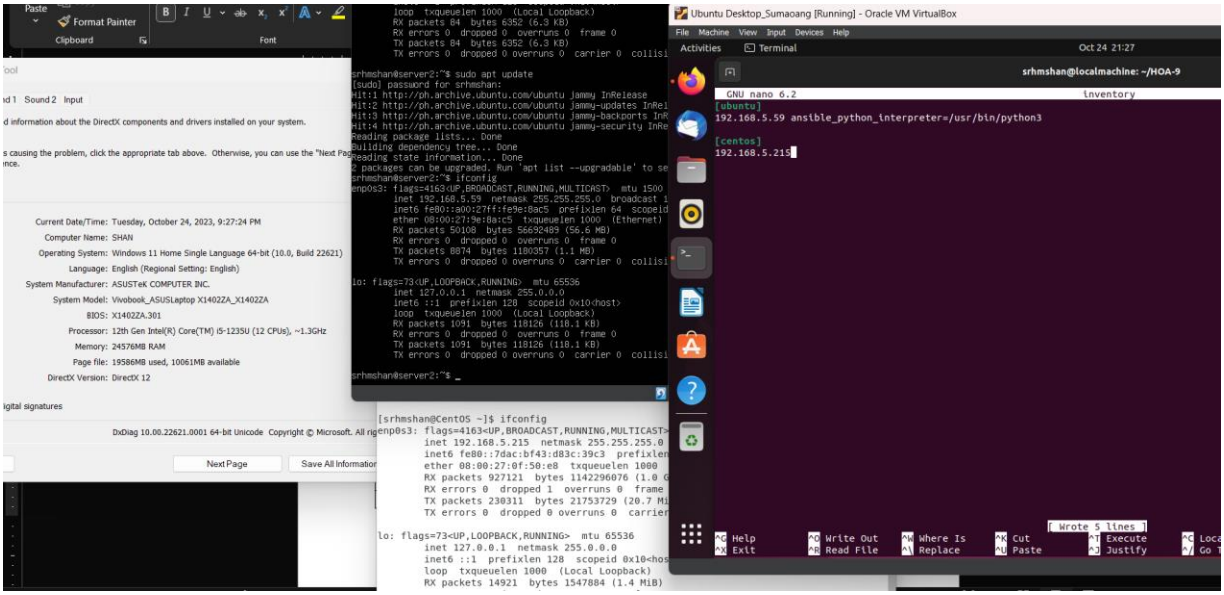
4. **Output** (screenshots and explanations)

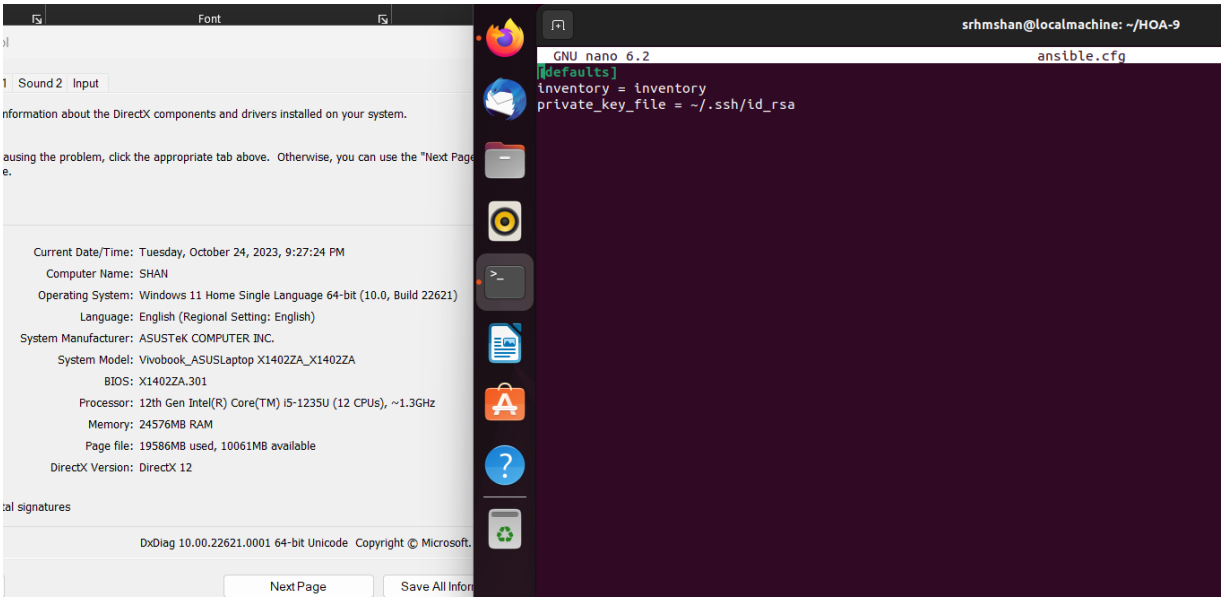
Step 1: Create a new repository in GitHub for this activity. Adding anything to it is optional. Next, copy your new repository in your CN using the code in the 2nd image



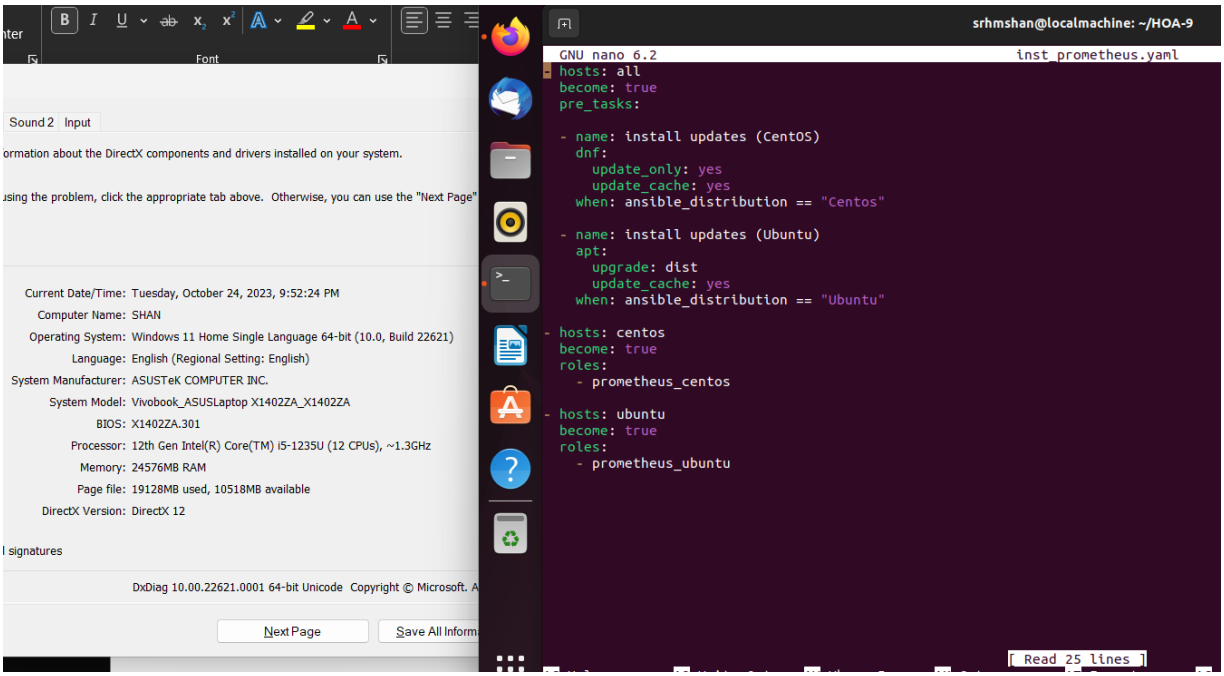


Step 2: Create your ansible.cfg and inventory files using the sudo nano command.



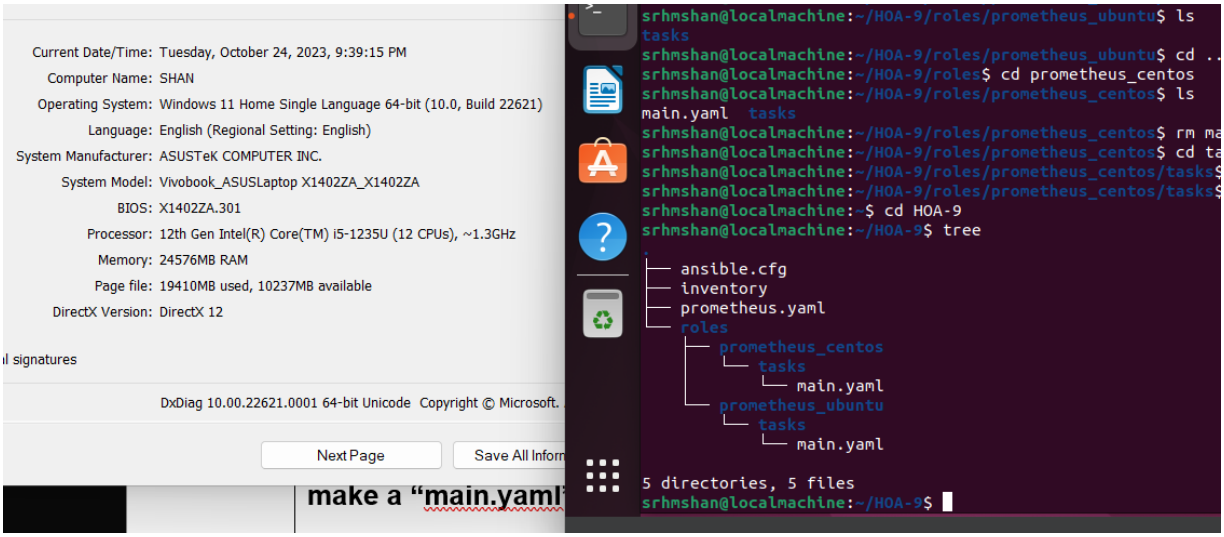


Step 3: Create an inst_prometheus.yaml file with the following command:



This is the main playbook wherein the roles are defined.

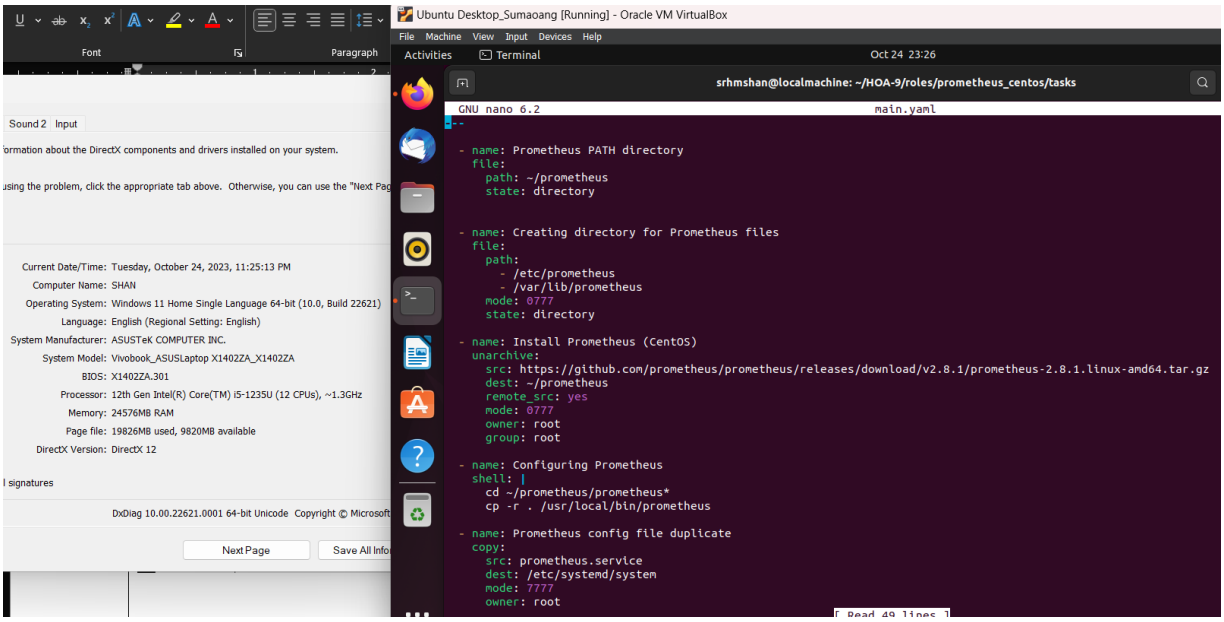
Step 4: Create a “roles” directory under your current directory. Under the “roles” directory, make 2 more directories according to the roles you defined in the inst_prometheus.yaml. Under each directory, create a “tasks” directory and make a “main.yaml” file.

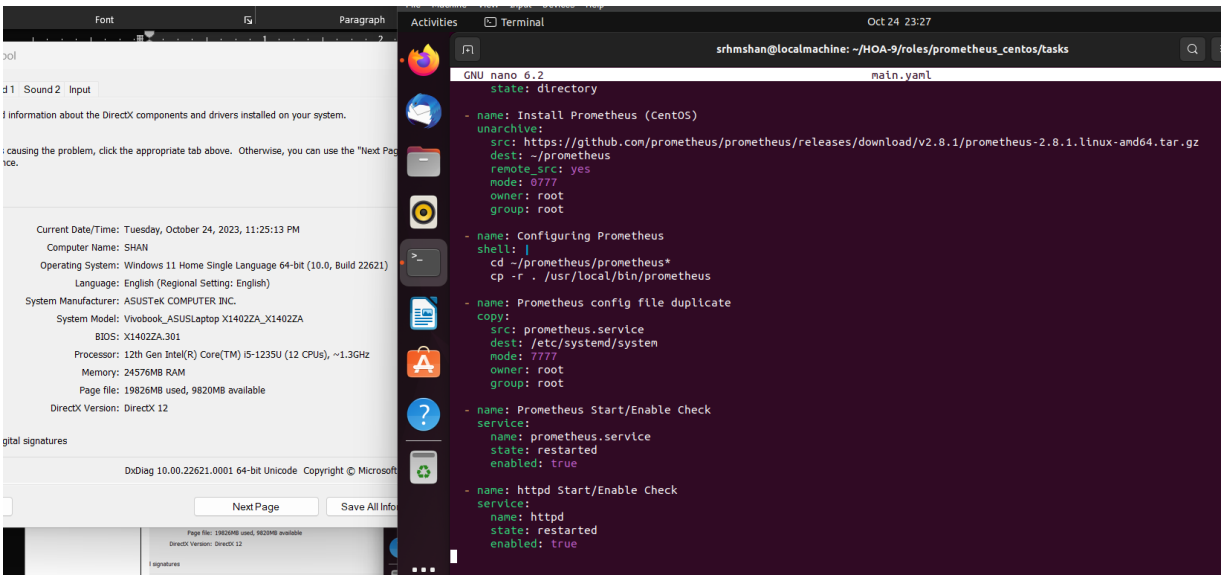


I applied the concept of creating roles.

Step 6: Step 5: Edit your main.yaml files and input all the needed tasks to install Prometheus.

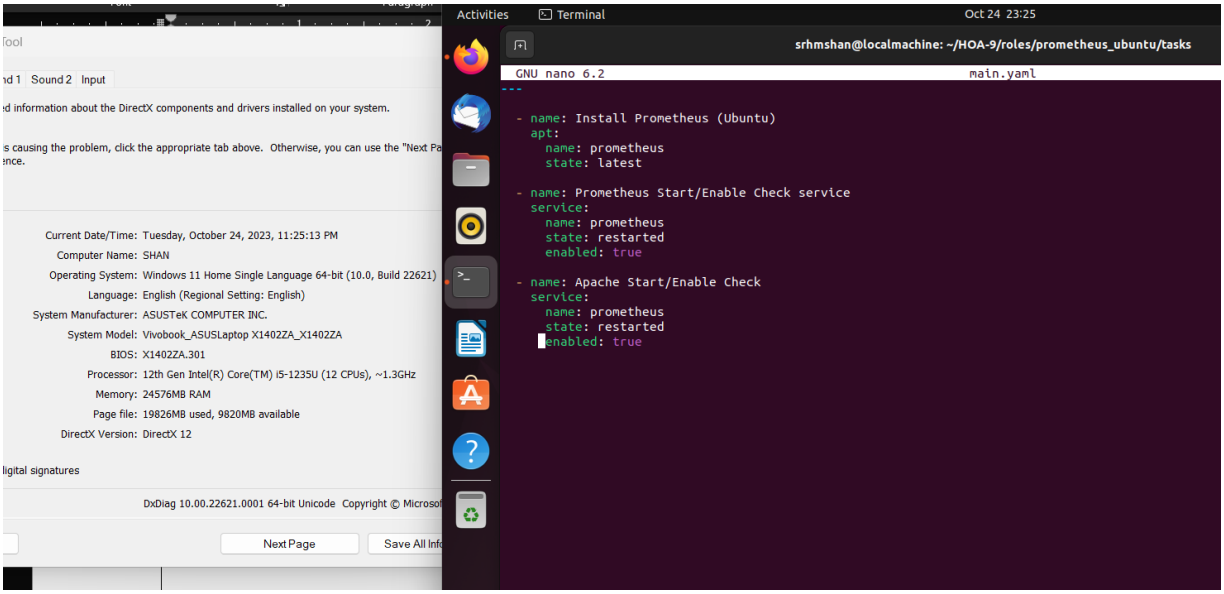
[CentOS]





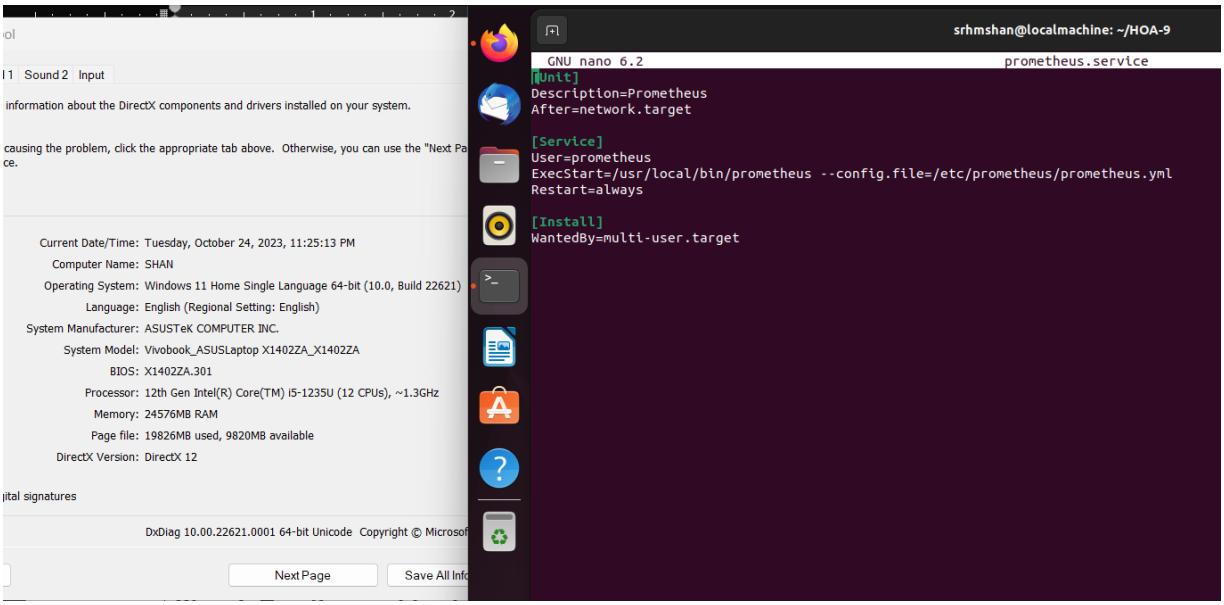
This sets up the necessary directories, installs Prometheus, copies its files, creates a systemd service unit for Prometheus, and ensures both the Prometheus service and the Apache HTTP Server service (httpd) are started and enabled.

[Ubuntu]



This is for Prometheus monitoring on an Ubuntu server. It first installs the Prometheus package, then ensures that the Prometheus service is running and enabled to start at boot.

Step 7: Create a prometheus.service file

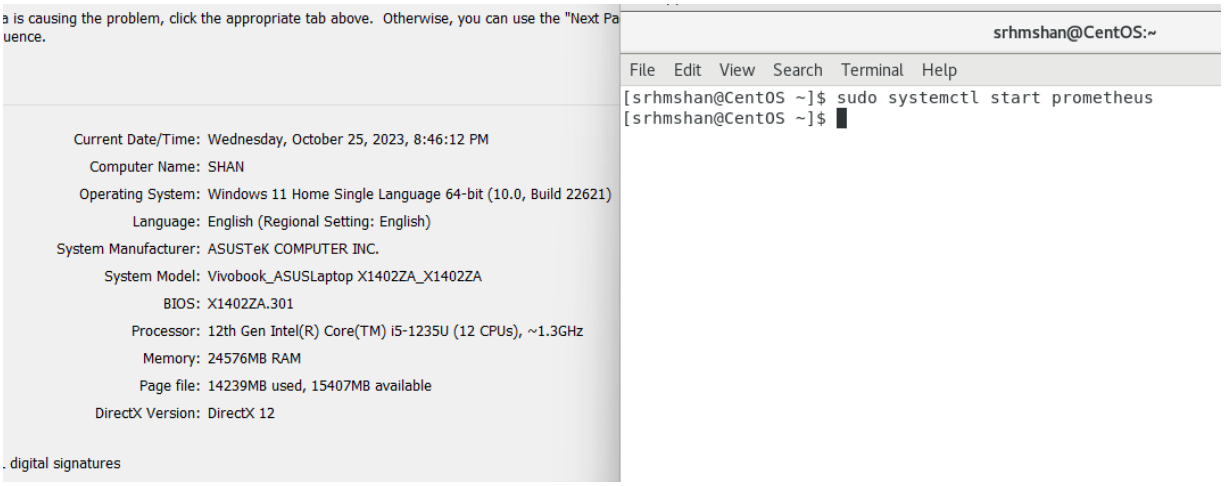


The prometheus.service file plays a role, in managing the Prometheus service on your system. It provides instructions, on how to start Prometheus specifies

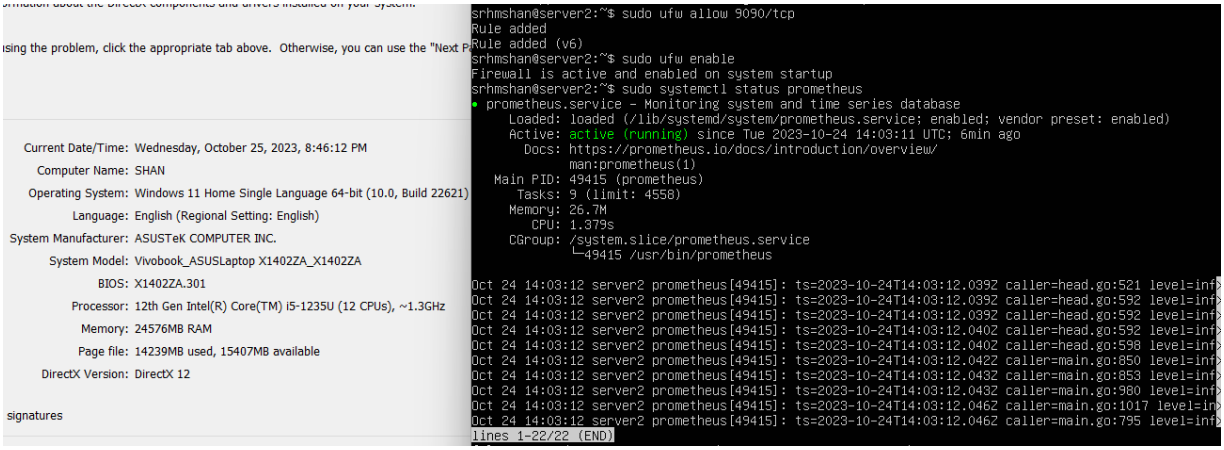
the user for running it and defines its behavior in case of any failures. Once you copy this service file to the /etc/systemd/system directory and enable it systemd will handle the task of starting and managing the Prometheus service based on these settings you have defined.

Step 8: Unblock the firewall for each server using the following commands:

[CentOS]

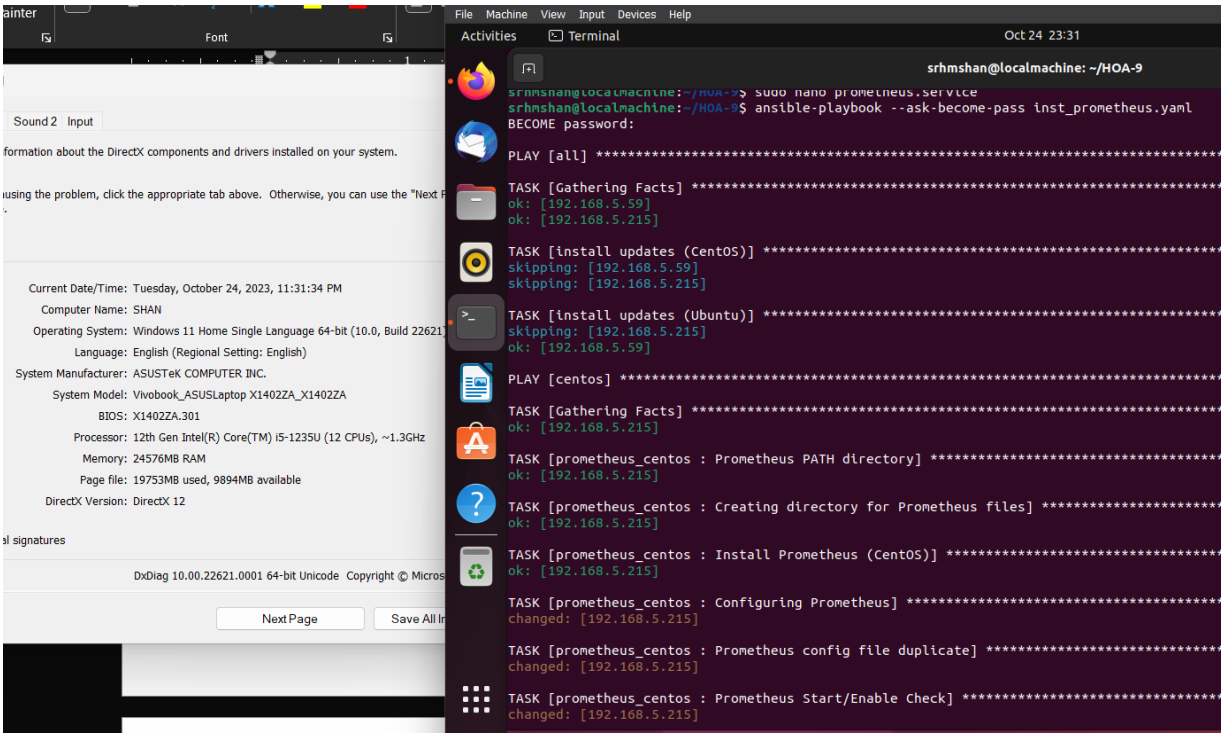


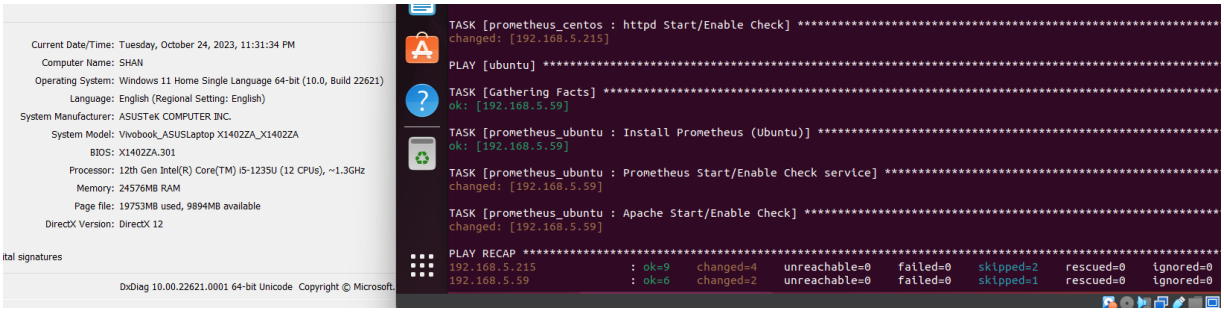
[Ubuntu]



Unblocking the firewall after installing Prometheus on Ubuntu/CentOS is necessary to allow network access to Prometheus, ensure that it can collect metrics, provide web UI, and communicate with other systems.

Step 9: Run the ansible playbook.

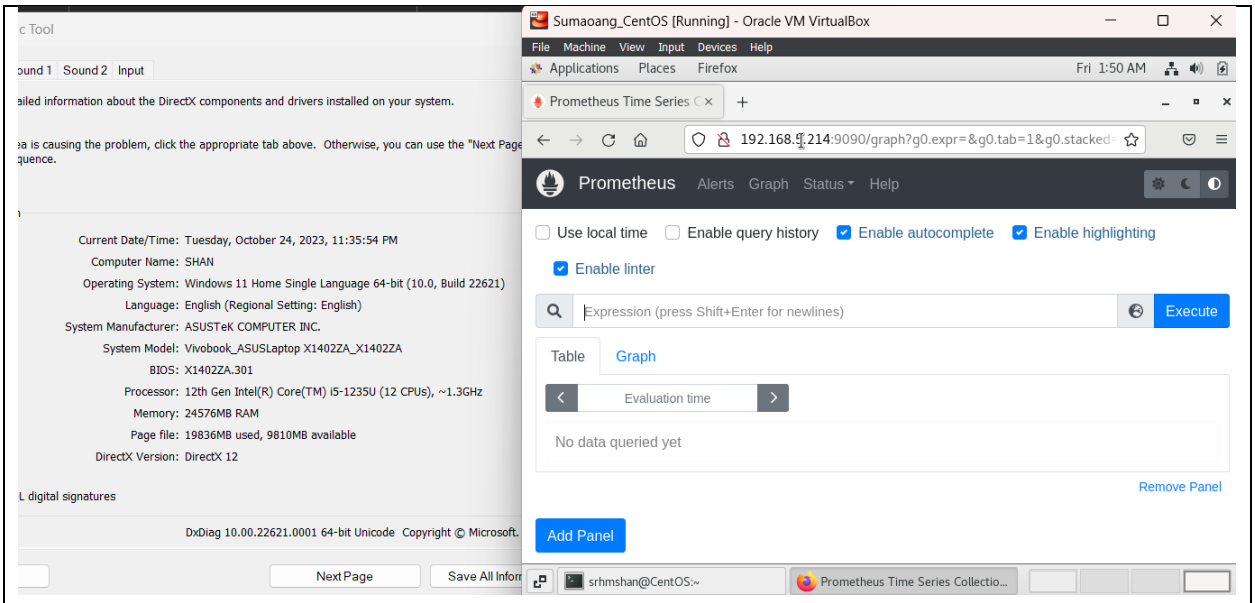




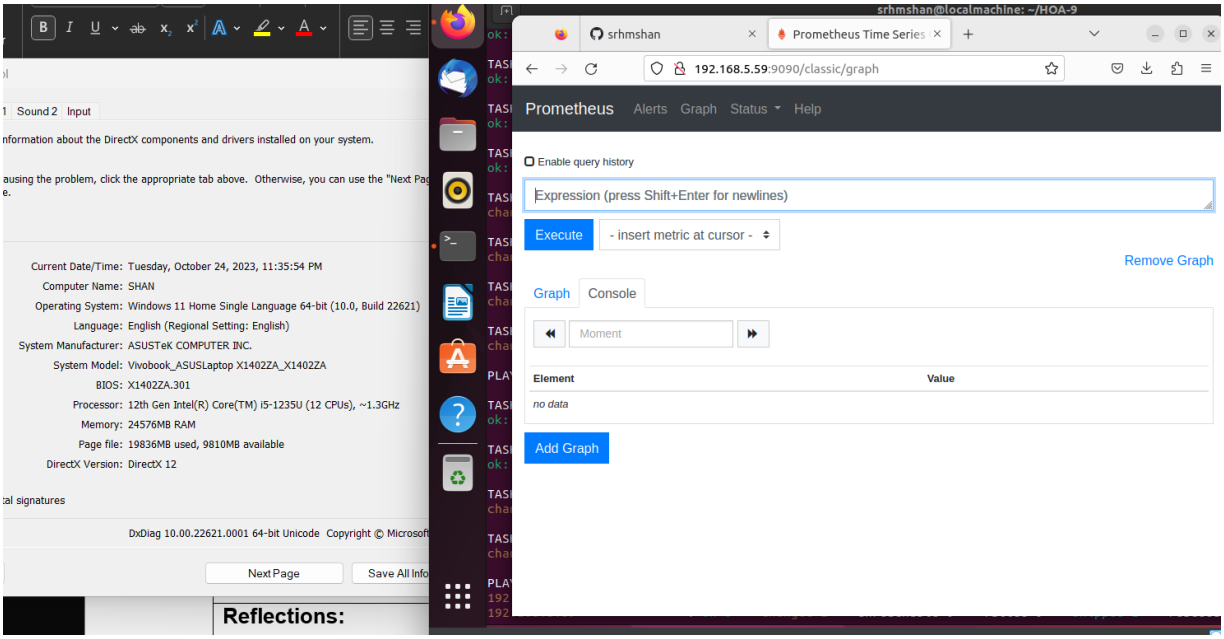
The task was successful.

Step 10: Verify if the installation was successful by putting "http://[ip_address]:9090" in your browser.

[CentOS]



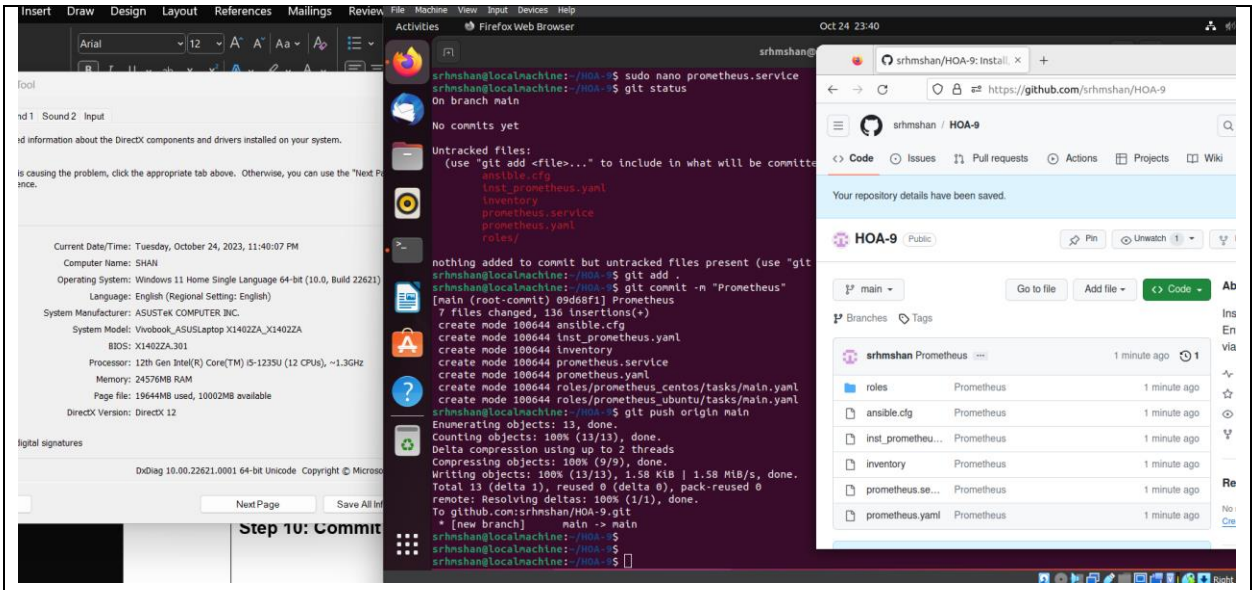
[Ubuntu]



Step 11

: Commit and push to GitHub.

https://github.com/srhmshan/HOA-9



Reflections:

Answer the following:

1. What are the benefits of having a performance monitoring tool?

Performance monitoring tools play a role, in ensuring the stability, security and efficiency of an IT infrastructure. They empower organizations to take measures in managing and enhancing their systems leading to cost savings, enhanced user experiences and improved operational efficiency overall.

Conclusions:

In this activity, I configured Prometheus monitoring on a CentOS and Ubuntu servers using an Ansible playbook. Comparing the tasks of installations, I've encountered more difficulty with CentOS than Ubuntu. To begin with, I made sure that the necessary directories and permissions were set up correctly allowing Prometheus to store its data and configuration files. Next, I downloaded Prometheus from its release. Moved the required files to their appropriate locations. Also, I created a systemd service unit file for Prometheus to define how it should be managed by systemd. To ensure convenience I made sure that the Prometheus service started automatically upon system boot and could be restarted in case of any failures. I also ensured that the Apache HTTP Server service (httpd) was running and enabled.

Overall, this activity enabled me to automate the deployment of Prometheus, along with its associated components, on my CentOS and Ubuntu servers. This made the setup and installation process easier, making it more efficient while reducing the chances of errors occurring.