

Name: Kenn Cherwin C. Yu	Date Performed: 31/03/2024
Course/Section: CPE31S1	Date Submitted: 01/04/2024
Instructor: Dr. Jonathan Taylar	Semester and SY: 2nd Sem
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	
<p>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.</p> <p>Prometheus</p> <p>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</p> <p>Cacti</p> <p>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</p>	
3. Tasks	
<ol style="list-style-type: none"> 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)	
<ul style="list-style-type: none"> - whole playbook 	

```
cherwin@cherwinyu:~/HOA9_YU$ tree
.
├── ansible.cfg
├── install_prometheus.yml
├── inventory
├── README.md
└── roles
    ├── centos_prometheus
    │   └── tasks
    │       └── main.yml
    └── ubuntu_prometheus
        └── tasks
            └── main.yml

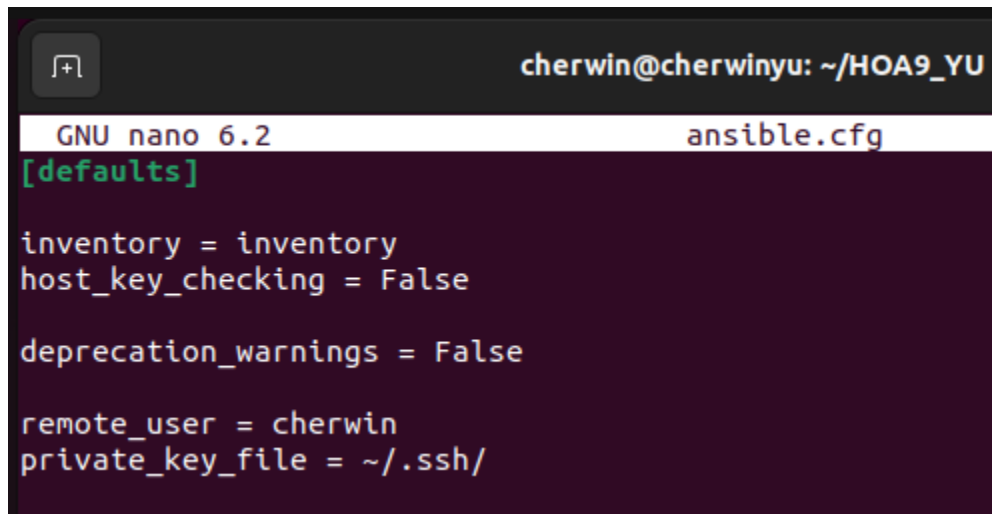
5 directories, 6 files
cherwin@cherwinyu:~/HOA9_YU$
```

install_prometheus.yml

```
cherwin@cherwinyu: ~/HOA9_YU
GNU nano 6.2      install_prometheus.yml
---
- hosts: all
  become: true
  pre_tasks:
    - name: install updates (CentOS)
      package:
        update_only: yes
        update_cache: yes
      when: ansible_distribution == "CentOS"
    - name: install wget (CentOS)
      package:
        name: wget
        state: latest
      when: ansible_distribution == "CentOS"
    - name: install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
      when: ansible_distribution == "Ubuntu"
- hosts: ubuntu_prometheus
  become: true
  roles:
    - ubuntu_prometheus
- hosts: centos_prometheus
  become: true
  roles:
    - centos_prometheus
```

- Here we created a command run named as install_prometheus.yml to help us run the code later.

ansible.cfg



```
cherwin@cherwinyu: ~/HOA9_YU
GNU nano 6.2 ansible.cfg
[defaults]

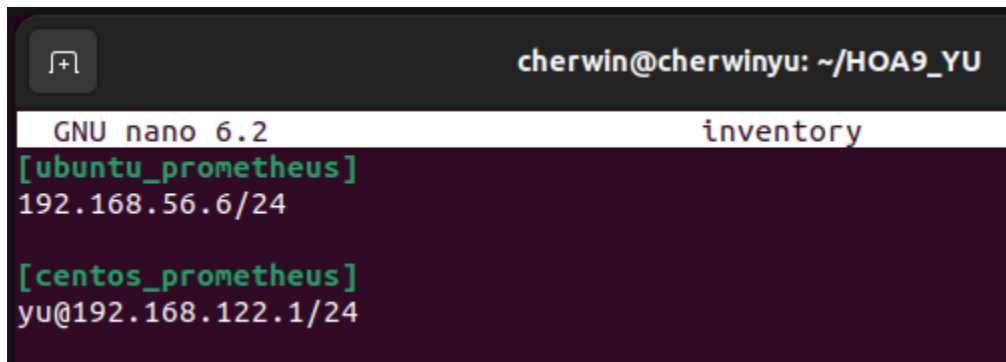
inventory = inventory
host_key_checking = False

deprecation_warnings = False

remote_user = cherwin
private_key_file = ~/.ssh/
```

- Here is the ansible.cfg of my workstation

inventory



```
cherwin@cherwinyu: ~/HOA9_YU
GNU nano 6.2 inventory
[ubuntu_prometheus]
192.168.56.6/24

[centos_prometheus]
yu@192.168.122.1/24
```

- This is the inventory used in the given tasks

Centos Prometheus

```
cherwin@cherwinyu: ~/HOA9_YU/roles/centos_prometheus/tasks
GNU nano 6.2 main.yml *
- name: Prometheus PATH directory
  file:
    path: ~/prometheus
    state: directory

- name: Creating directory for Prometheus files
  file:
    path:
      - /etc/prometheus
      - /var/lib/prometheus
    mode: 0777
    state: directory

- name: Install Prometheus (CentOS)
  unarchive:
    src: https://github.com/prometheus/prometheus/releases/download/v2.8.1/prometheus-2.8.1.linux-amd64.tar.gz
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Configuring Prometheus
  shell: |
    cd ~/prometheus/prometheus*
    cp -r . /usr/local/bin/prometheus

- name: Prometheus config file duplicate
  copy:
    src: prometheus.service
    dest: /etc/systemd/system
    mode: 777
    owner: root
    group: root
```

- This is the code that was used in running and creating Prometheus under Centos.

Ubuntu Prometheus

GNU nano 6.2

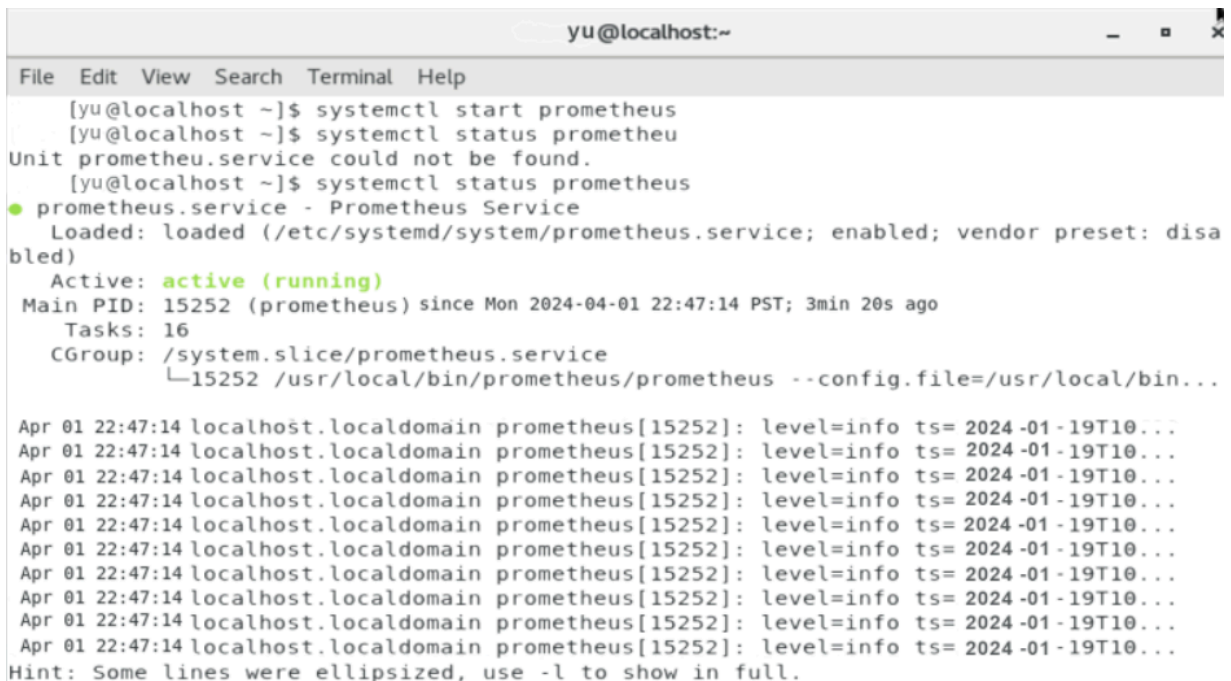
```
- name: install prometheus (Ubuntu)
  apt:
    name: prometheus
    state: latest

- name: Prometheus Start/Enable Check
  service:
    name: prometheus
    state: restarted
    enabled: true

- name: Apache Start/Enable Check
  service:
    name: prometheus
    state: restarted
    enabled: true
```

- This is the code that was used in running and creating Prometheus under Ubuntu s.

Prometheus output

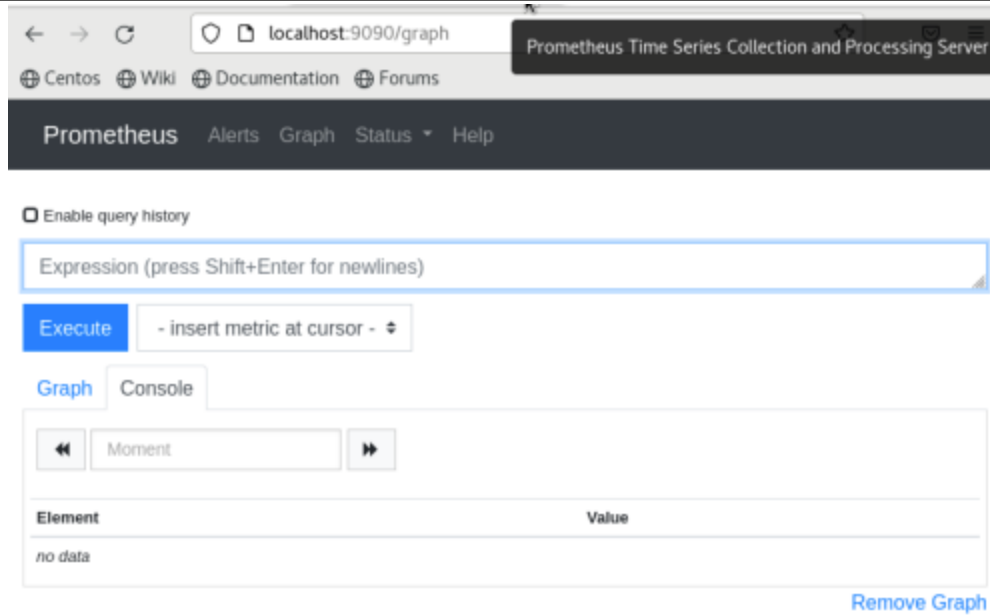


The screenshot shows a terminal window titled 'yu@localhost:~'. The terminal output is as follows:

```
File Edit View Search Terminal Help
[yu@localhost ~]$ systemctl start prometheus
[yu@localhost ~]$ systemctl status prometheus
Unit prometheus.service could not be found.
[yu@localhost ~]$ systemctl status prometheus
● prometheus.service - Prometheus Service
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
   Active: active (running)
 Main PID: 15252 (prometheus) since Mon 2024-04-01 22:47:14 PST; 3min 20s ago
    Tasks: 16
   CGroup: /system.slice/prometheus.service
           └─15252 /usr/local/bin/prometheus/prometheus --config.file=/usr/local/bin...
```

Below the service status, there are several log lines from Prometheus:

```
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Apr 01 22:47:14 localhost.localdomain prometheus[15252]: level=info ts= 2024-01-19T10...
Hint: Some lines were ellipsized, use -l to show in full.
```



- After Running the playbook we input `systemctl start prometheus` and `systemctl status` to show if prometheus is running in the CentOS server.

Reflections:

Answer the following:

1. What are the benefits of having a performance monitoring tool?
 - Performance monitoring tools provide real-time information about the health and efficiency of systems and applications. They enable the proactive detection of potential performance issues before they affect users or operations. By examining performance measurements, these tools aid in identifying bottlenecks and areas for optimization, resulting in increased system reliability and response times. Continuous monitoring enables organizations to make data-driven decisions about resource allocation and task prioritization based on actual performance data. Finally, performance monitoring tools help to increase productivity, user satisfaction, and overall business success.

Conclusions:

- This hands-on exercise taught us how to install Prometheus and the value of a system performance monitoring tool while managing several machines in a datacenter or an office. With the new skills and information we gained throughout this hands-on exercise, we can now monitor the performance and status of remote systems using Nagios and Prometheus. We install it by creating a main.yml code for ubuntu and also one specifically for centos and run it using ansible-playbook --ask-become-pass install_prometheus.yml.