

Name: Christian Aaron C. Buduan	Date Performed: 10/14/2024
Course/Section: CpE31S2	Date Submitted: 10/16/2024
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st sem 2024-2025
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)	

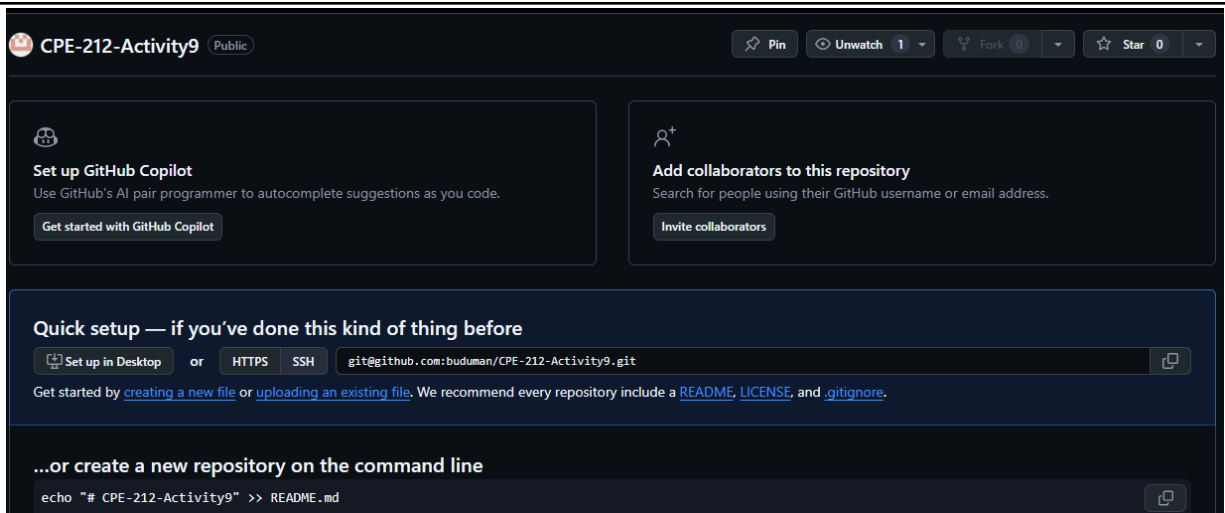


Figure 9.1 Create a working environment where we can install any program. I created a new repository to save all the configurations and playbooks that I have created.

```
qcacbuduan@Workstation:~$ git clone git@github.com:buduman/CPE-212-Activity9.git
Cloning into 'CPE-212-Activity9'...
warning: You appear to have cloned an empty repository.
```

Figure 9.2 Use git clone command to put the new repository in the control node. this is where you will make your configurations and playbooks.

```
qcacbuduan@Workstation:~$ cp -a *8/. *9/.
qcacbuduan@Workstation:~$ cd *9
qcacbuduan@Workstation:~/CPE-212-Activity9$ ls
act8.yml  ansible.cfg  inventory  roles
qcacbuduan@Workstation:~/CPE-212-Activity9$ tree
.
├── act9.yml
├── ansible.cfg
├── inventory
└── roles
    ├── base
    │   └── tasks
    │       └── main.yml
    ├── db_servers
    │   └── tasks
    │       └── main.yml
    └── web_servers
        └── tasks
            └── main.yml
```

Figure 9.3-4 For this activity, I copied all the files that were made from activity 8 into activity9 directory and will make changes when necessary.

```

qcacbuduan@Workstation:~/CPE-212-Activity9$ cat roles/w*/tasks/main.yml
---
- name: Open port 9090
  ufw:
    rule: allow
    port: 9090
    proto: tcp
    state: enabled

- name: Install Prometheus (Ubuntu)
  apt:
    name: prometheus
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: Create prometheus.service file
  copy:
    dest: /etc/systemd/system/prometheus.service
    content: |
      [Unit]
      Description=Prometheus
      Wants=network-online.target
      After=network-online.target

      [Service]
      User=root
      Group=root
      Type=simple
      ExecStart=/usr/local/bin/prometheus \
        --config.file /etc/prometheus/prometheus.yml \
        --storage.tsdb.path /var/lib/prometheus \
        --web.console.templates=/etc/prometheus/consoles \
        --web.console.libraries=/etc/prometheus/console_libraries \

      [Install]
      WantedBy=multi-user.target

- name: Start Prometheus Service (Ubuntu)
  systemd:
    name: prometheus
    enabled: yes
    state: started
  when: ansible_distribution == "Ubuntu"

```

Figure 9.5 I created a new yml file under the tasks of web_servers role which consists of configuring ports, creating service files and installing prometheus in ubuntu to make sure that what we are installing would work.

```
qcacbuduan@Workstation:~/CPE-212-Activity9$ cat roles/d*/tasks/main.yml
```

```
---
```

- name: Allow Prometheus for Firewall
firewalld:
 port: 9090/tcp
 permanent: yes
 state: enabled
- name: Install Prometheus (CentOS)
unarchive:
 src: <https://github.com/prometheus/prometheus/releases/download/v2.30.0/prometheus-2.30.0.linux-amd64.tar.gz>
 dest: /usr/local/bin
 remote_src: yes
 mode: 0755
 owner: root
 group: root
 when: ansible_distribution == "CentOS"
- name: Copy Prometheus binaries
copy:
 src: /usr/local/bin/prometheus-2.30.0.linux-amd64/prometheus
 dest: /usr/local/bin/prometheus
 mode: 0755
 remote_src: yes
 when: ansible_distribution == "CentOS"
- name: Copy Promtool binaries
copy:
 src: /usr/local/bin/prometheus-2.30.0.linux-amd64/prometheus
 dest: /usr/local/bin/promtool
 mode: 0755
 remote_src: yes
 when: ansible_distribution == "CentOS"
- name: Create Prometheus directories
file:
 path: "{{ item }}"
 state: directory
 loop:
 - /etc/prometheus
 - /var/lib/prometheus
 when: ansible_distribution == "CentOS"

```

- name: Create Prometheus directories
  file:
    path: "{{ item }}"
    state: directory
  loop:
    - /etc/prometheus
    - /var/lib/prometheus
  when: ansible_distribution == "CentOS"

- name: Copy prometheus.yml to /etc/prometheus
  command: cp /usr/local/bin/prometheus-2.30.0.linux-amd64/prometheus.yml /etc/prometheus
  when: ansible_distribution == "CentOS"

- name: Copy consoles directory to /etc/prometheus
  command: cp -r /usr/local/bin/prometheus-2.30.0.linux-amd64/consoles /etc/prometheus
  when: ansible_distribution == "CentOS"

- name: Copy console_libraries directory to /etc/prometheus
  command: cp -r /usr/local/bin/prometheus-2.30.0.linux-amd64/console_libraries /etc/prometheus
  when: ansible_distribution == "CentOS"

- name: Create prometheus.service file
  copy:
    dest: /etc/systemd/system/prometheus.service
    content: |
      [Unit]
      Description=Prometheus
      Wants=network-online.target
      After=network-online.target

      [Service]
      User=root
      Group=root
      Type=simple
      ExecStart=/usr/local/bin/prometheus \
        --config.file /etc/prometheus/prometheus.yml \
        --storage.tsdb.path /var/lib/prometheus \
        --web.console.templates=/etc/prometheus/consoles \
        --web.console.libraries=/etc/prometheus/console_libraries \

      [Install]
      WantedBy=multi-user.target
  when: ansible_distribution == "CentOS"

- name: Reload systemd
  command: systemctl daemon-reload
  when: ansible_distribution == "CentOS"

- name: Start Prometheus Service
  systemd:
    name: prometheus
    enabled: yes
    state: started
  when: ansible_distribution == "CentOS"

```

Figure 9.6-8 I created a new file under the tasks of db_servers role which consists of creating a directory to install prometheus in CentOS, allowing it to bypass firewall, and install prometheus along with its binaries and other utilities to make sure that prometheus is installed properly and working

After creating both yml files in their respective roles, run the act9.yml using *ansible-playbook --ask-become-pass command*.

```

PLAY [web_servers] *****

TASK [Gathering Facts] *****
ok: [server1]

TASK [web_servers : Open port 9090] *****
[WARNING]: The value 9090 (type int) in a string field was converted to u'9090'
(type string). If this does not look like what you expect, quote the entire
value to ensure it does not change.
changed: [server1]

TASK [web_servers : Install Prometheus (Ubuntu)] *****
changed: [server1]

TASK [web_servers : Create prometheus.service file] *****
changed: [server1]

TASK [web_servers : Start Prometheus Service (Ubuntu)] *****
ok: [server1]

```

Figure 9.9 running play in web_servers role

```

PLAY [db_servers] *****

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

TASK [db_servers : Allow Prometheus for Firewall] *****
changed: [centosbuduan]

TASK [db_servers : Install Prometheus (CentOS)] *****
changed: [centosbuduan]

TASK [db_servers : Copy Prometheus binaries] *****
changed: [centosbuduan]

TASK [db_servers : Copy Promtool binaries] *****
changed: [centosbuduan]

TASK [db_servers : Create Prometheus directories] *****
changed: [centosbuduan] => (item=/etc/prometheus)
changed: [centosbuduan] => (item=/var/lib/prometheus)

TASK [db_servers : Copy prometheus.yml to /etc/prometheus] *****
changed: [centosbuduan]

TASK [db_servers : Copy consoles directory to /etc/prometheus] *****
changed: [centosbuduan]

TASK [db_servers : Copy console_libraries directory to /etc/prometheus] *****
changed: [centosbuduan]

TASK [db_servers : Create prometheus.service file] *****
changed: [centosbuduan]

TASK [db_servers : Reload systemd] *****
changed: [centosbuduan]

TASK [db_servers : Start Prometheus Service] *****
changed: [centosbuduan]

```

Figure 9.10 running play in db_servers role

```

qcacbuduan@server1:~$ systemctl status prometheus
● prometheus.service - Monitoring system and time series database
   Loaded: loaded (/lib/systemd/system/prometheus.service; enabled; vendor pres
   Active: active (running) since Mon 2024-10-21 07:55:06 +08; 11min ago
     Docs: https://prometheus.io/docs/introduction/overview/
   Main PID: 5288 (prometheus)
      Tasks: 11 (limit: 4656)
    CGroup: /system.slice/prometheus.service
            └─5288 /usr/bin/prometheus
lines 1-8/8 (END)

```

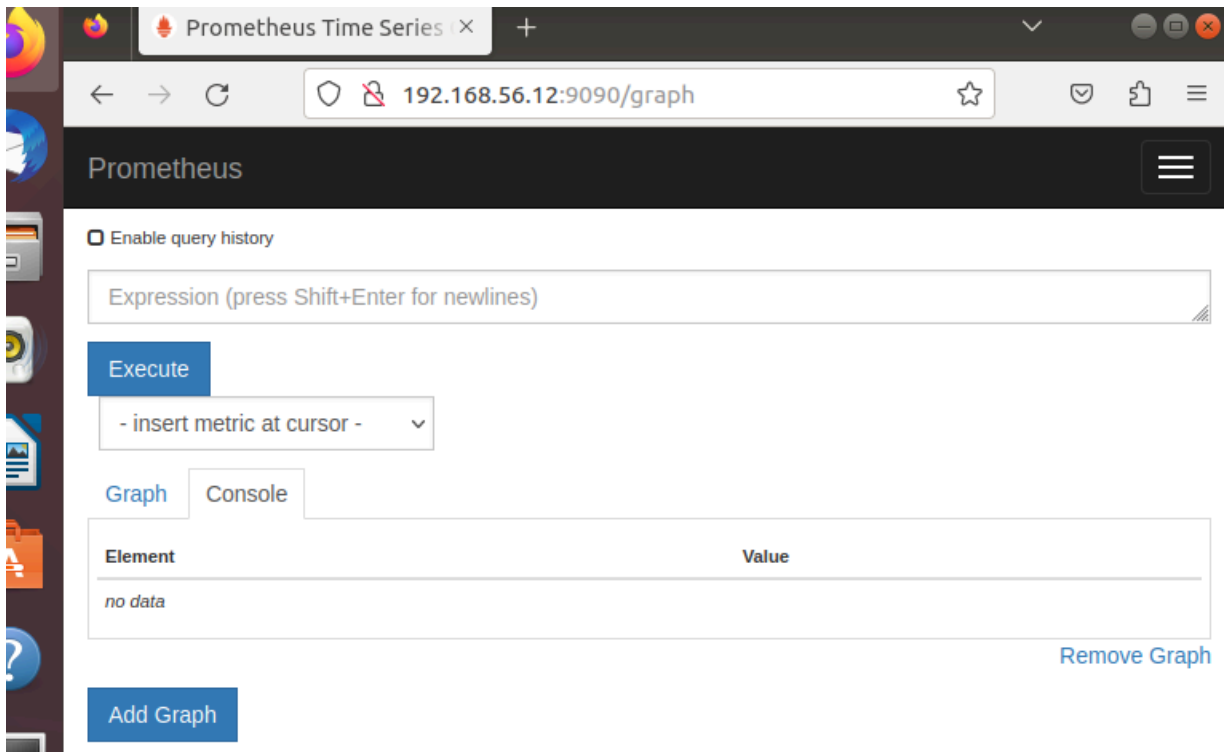


Figure 9.11-12 verify if prometheus is installed in Ubuntu by using `systemctl status` command on the terminal and in the web browser by typing `{server's ip address}:9090`

```

[qcacbuduan@centosbuduan ~]$ systemctl status prometheus
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor prese
t: disabled)
   Active: active (running) since Mon 2024-10-21 07:56:08 PST; 9min ago
     Main PID: 5778 (prometheus)
        Tasks: 9
    CGroup: /system.slice/prometheus.service
            └─5778 /usr/local/bin/prometheus --config.file /etc/prometheus/pro...

```

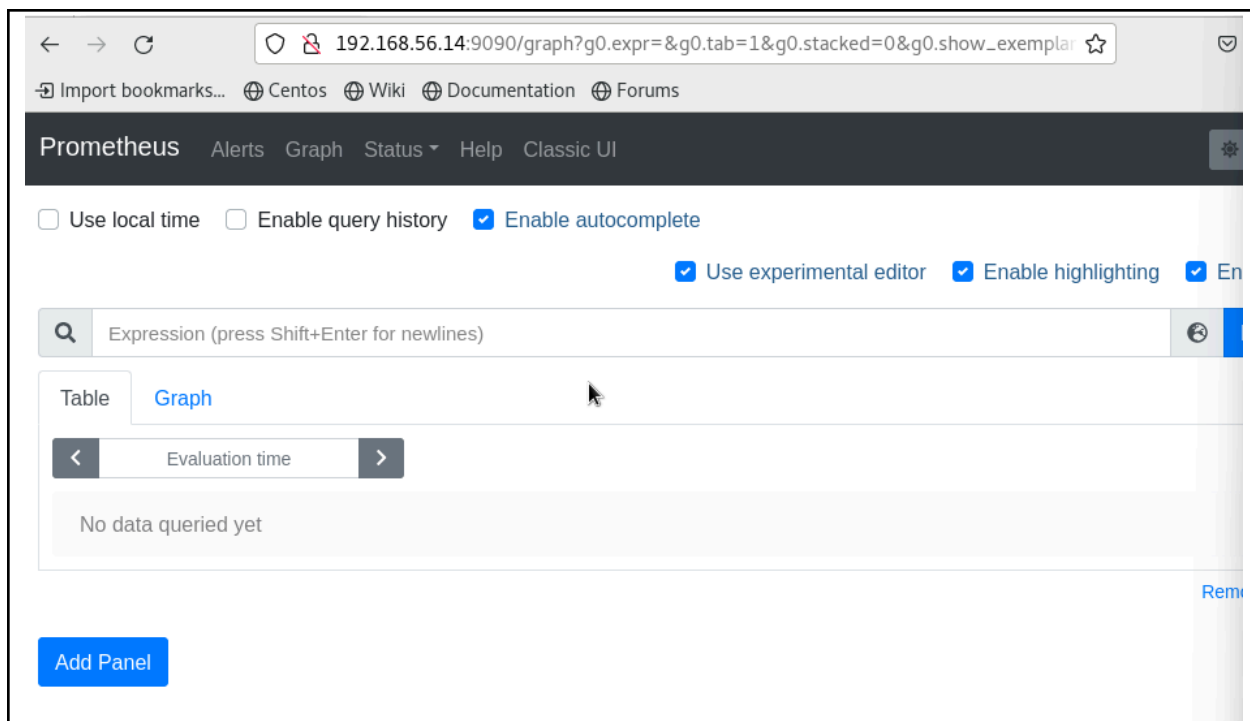


Figure 9.13-14 verify if prometheus is installed in CentOS by using `systemctl status` command on the terminal and in the web browser by typing `{server's ip address}:9090`

```
qcacbuduan@Workstation:~/CPE-212-Activity9$ git add roles
qcacbuduan@Workstation:~/CPE-212-Activity9$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   roles/web_servers/tasks/main.yml

qcacbuduan@Workstation:~/CPE-212-Activity9$ git commit -m "Act9 done"
[master 3d3bcc8] Act9 done
 1 file changed, 5 insertions(+), 5 deletions(-)
qcacbuduan@Workstation:~/CPE-212-Activity9$ git push origin master
Counting objects: 6, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 463 bytes | 463.00 KiB/s, done.
Total 6 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To github.com:buduman/CPE-212-Activity9.git
 c48129a..3d3bcc8 master -> master
```

Figure 9.15 Save the work into the github repository by using the following commands as per image.

Github Repo: <https://github.com/buduman/CPE-212-Activity9.git>

Reflections:

Answer the following:

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

The benefits of having a performance monitoring tool like prometheus is that it improves reliability by detecting issues by collecting detailed data. This helps in reducing server outages, aids you in troubleshooting, and notifies you about these issues, thus offering you a faster response time in solving these issues making the user's experience better. Apart from that, operating and using prometheus is simple, which gives you a user-friendly experience to the server manager, and to the users.

Conclusions:

After doing this activity, I was able to learn the importance of performance monitoring tools such as Prometheus in managing your servers. It fundamentally identifies current resource consumption of the workload. In installing Prometheus in different servers using ansible playbook, I was able to learn about new keywords to use in the playbook such as how to open ports, creating directories and copying files, along with creating service files in which made me able to install Prometheus properly without any issues.