Name: Torrecampo, Juan Piolo S.	**Date Performed:** Oct 22, 2022
Course/Section: CPE 232 / CPE31S22	**Date Submitted:** Oct 22, 2022
Instructor: Dr. Jonathan Taylar	**Semester and SY:** 1st Sem, 2022 - 2023

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

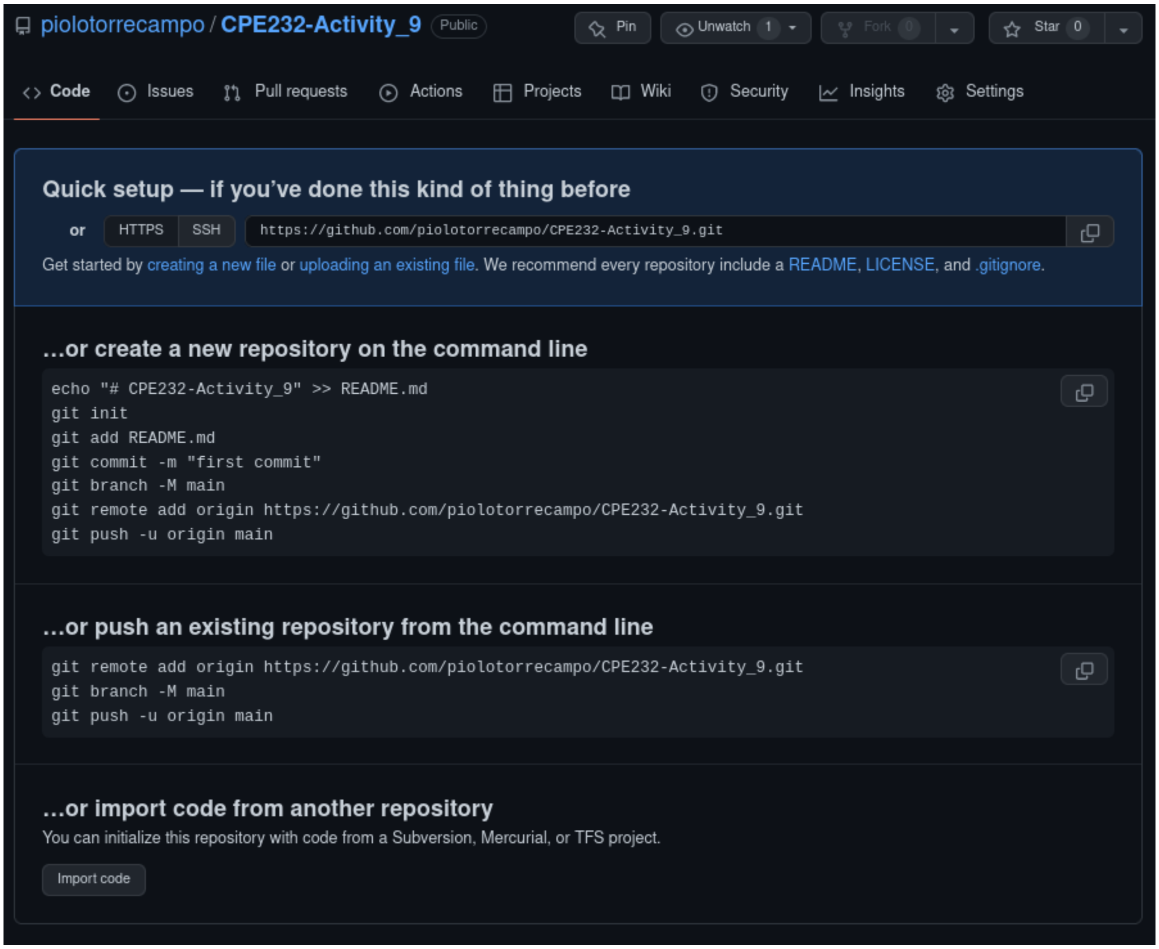
Cacti is a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality. Cacti provides a fast poller, advanced graph templating, multiple data acquisition methods, and user management features out of the box. All of this is wrapped in an intuitive, 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 and clone it into your machine.



Quick setup — if you've done this kind of thing before

or HTTPS SSH `https://github.com/piolotorrecampo/CPE232-Activity_9.git`

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

…or create a new repository on the command line

```
echo "# CPE232-Activity_9" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/piolotorrecampo/CPE232-Activity_9.git
git push -u origin main
```

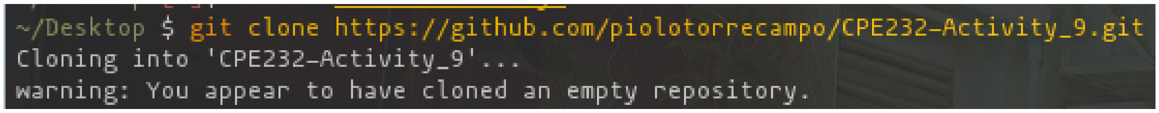
…or push an existing repository from the command line

```
git remote add origin https://github.com/piolotorrecampo/CPE232-Activity_9.git
git branch -M main
git push -u origin main
```

…or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

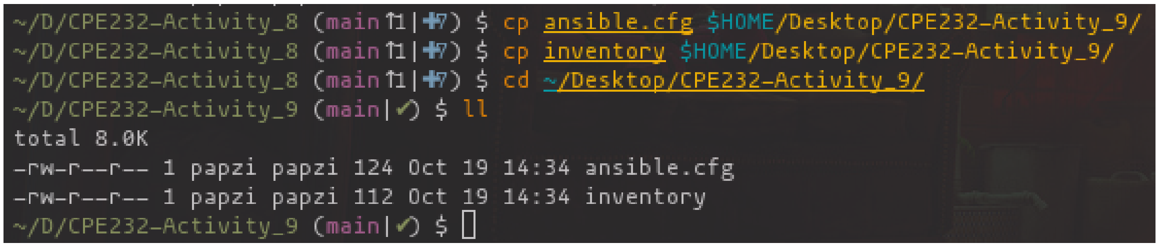
Figure 1.1. Github page of the new repository.



```
~/Desktop $ git clone https://github.com/piolotorrecampo/CPE232-Activity_9.git
Cloning into 'CPE232-Activity_9'...
warning: You appear to have cloned an empty repository.
```

Figure 1.2. Cloning the newly created repository in the host machine.

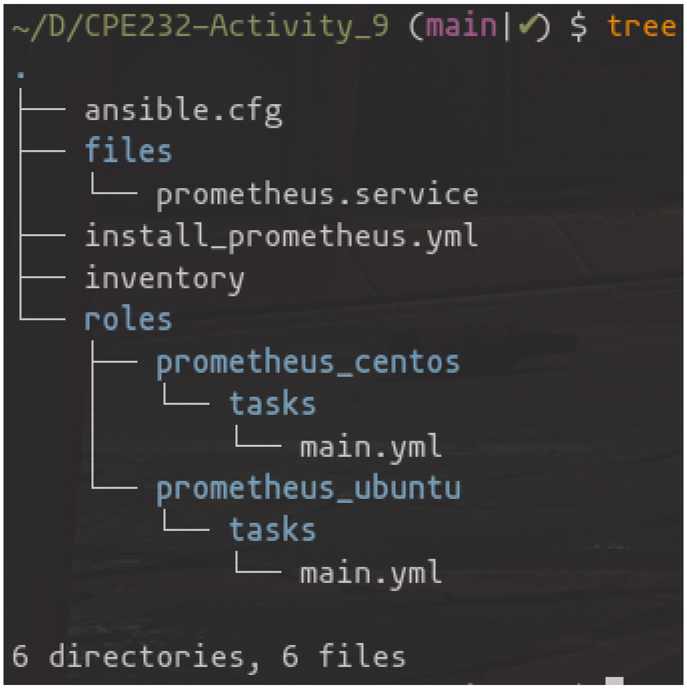
STEP 2. Copy the inventory and ansible.cfg from the previous repository in Activity 8.



```
~/D/CPE232-Activity_8 (main⇅1|✚7) $ cp ansible.cfg $HOME/Desktop/CPE232-Activity_9/
~/D/CPE232-Activity_8 (main⇅1|✚7) $ cp inventory $HOME/Desktop/CPE232-Activity_9/
~/D/CPE232-Activity_8 (main⇅1|✚7) $ cd ~/Desktop/CPE232-Activity_9/
~/D/CPE232-Activity_9 (main|✓) $ ll
total 8.0K
-rw-r--r-- 1 papzi papzi 124 Oct 19 14:34 ansible.cfg
-rw-r--r-- 1 papzi papzi 112 Oct 19 14:34 inventory
~/D/CPE232-Activity_9 (main|✓) $ 
```

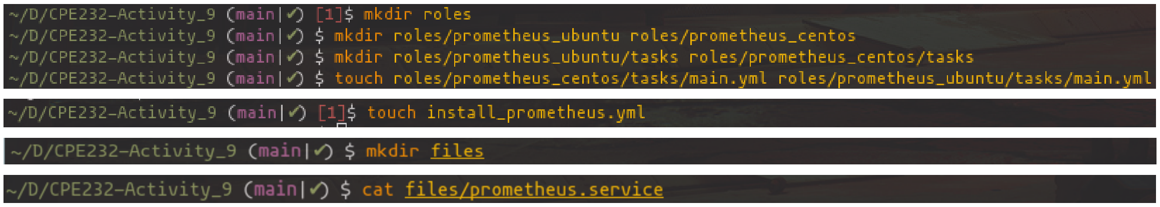
Figure 1.3. Copying the inventory and ansible.cfg from the previous repository.

STEP 3. Create the needed files and directories, as shown in *Figure 1.4.*



```
~/D/CPE232-Activity_9 (main|✓) $ tree
.
├── ansible.cfg
├── files
│   └── prometheus.service
├── install_prometheus.yml
├── inventory
└── roles
    ├── prometheus_centos
    │   └── tasks
    │       └── main.yml
    └── prometheus_ubuntu
        └── tasks
            └── main.yml

6 directories, 6 files
```

Figure 1.4. Tree structure of the directory.



```
~/D/CPE232-Activity_9 (main|✓) [1]$ mkdir roles
~/D/CPE232-Activity_9 (main|✓) $ mkdir roles/prometheus_ubuntu roles/prometheus_centos
~/D/CPE232-Activity_9 (main|✓) $ mkdir roles/prometheus_ubuntu/tasks roles/prometheus_centos/tasks
~/D/CPE232-Activity_9 (main|✓) $ touch roles/prometheus_centos/tasks/main.yml roles/prometheus_ubuntu/tasks/main.yml
```

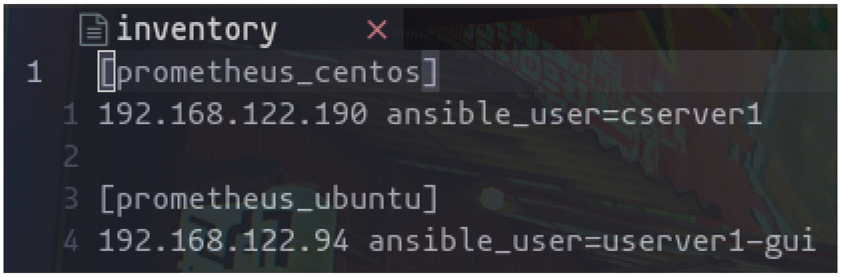
```
~/D/CPE232-Activity_9 (main|✓) [1]$ touch install_prometheus.yml
```

```
~/D/CPE232-Activity_9 (main|✓) $ mkdir files
```

```
~/D/CPE232-Activity_9 (main|✓) $ cat files/prometheus.service
```

Figure 1.5. Creating the files and directories inside of the repository

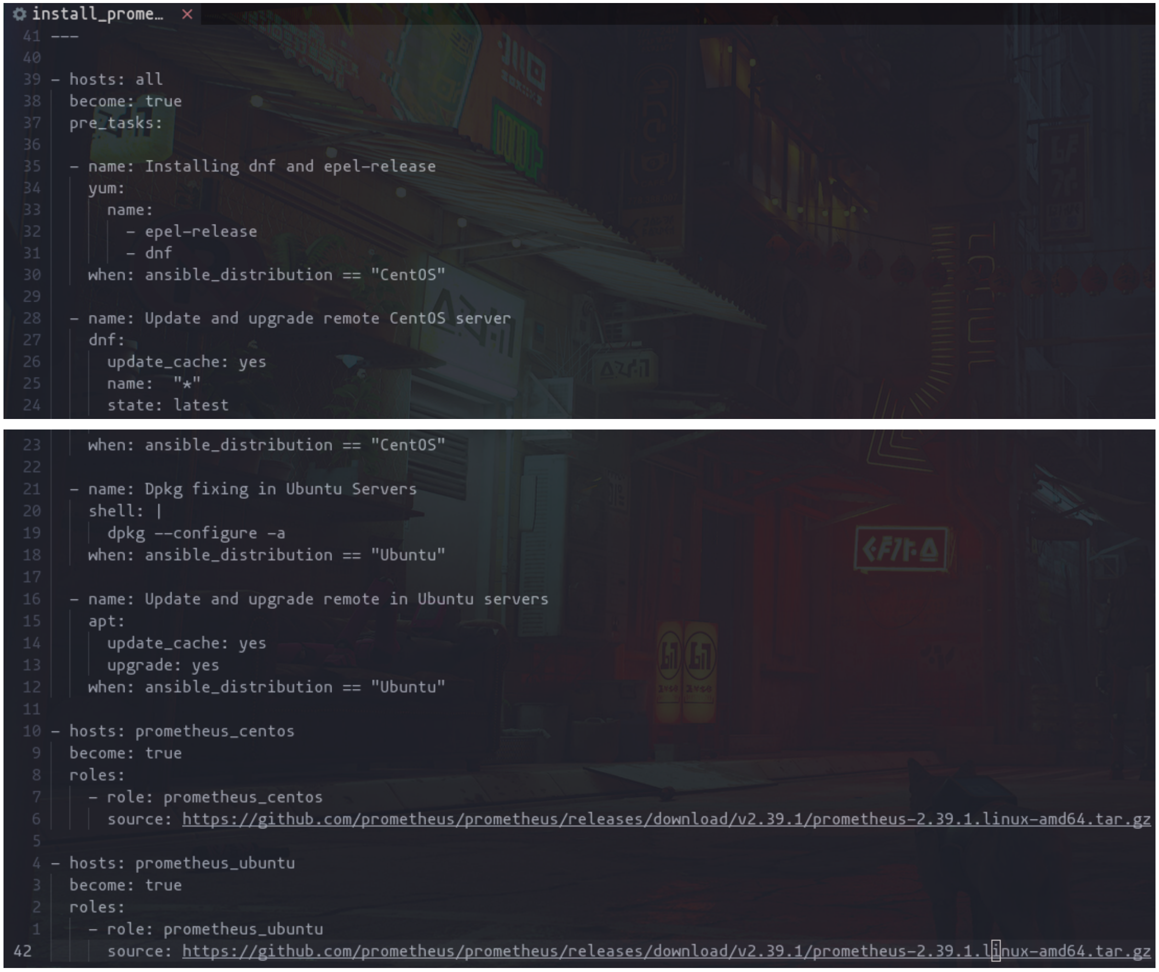
STEP 4. Copy the format below for the inventory file. Take note that the IP address and username varies depending on your machine.



```
[prometheus_centos]
192.168.122.190 ansible_user=cserver1

[prometheus_ubuntu]
192.168.122.94 ansible_user=userver1-gui
```

Figure 1.6. Content of inventory file which supports the 2 servers (Ubuntu and CentOS).

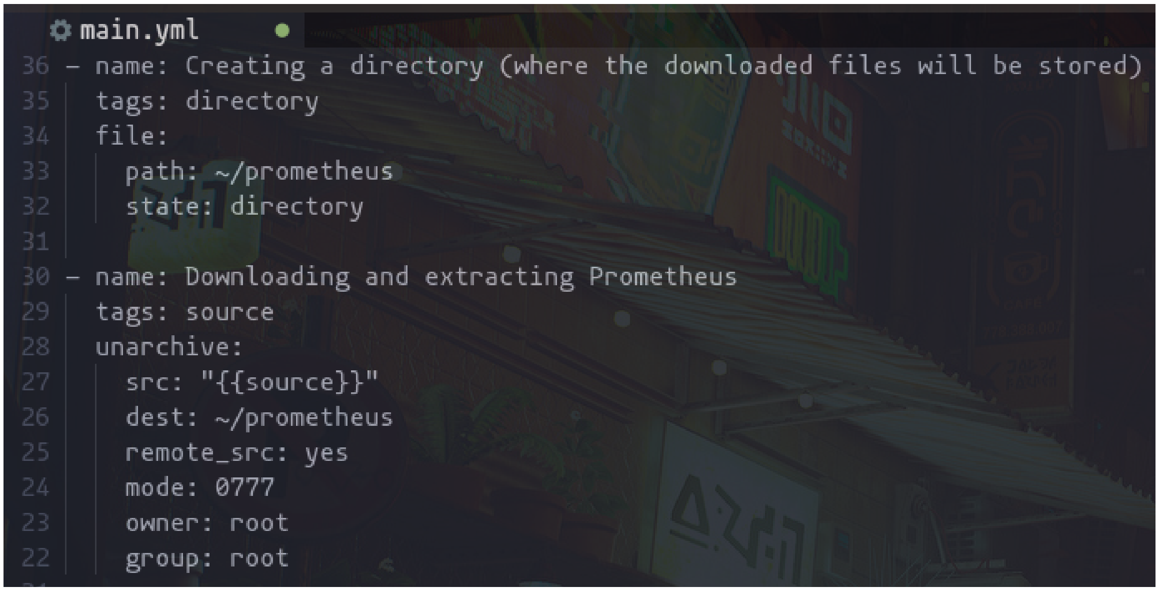
STEP 5. Copy the contents of the install_prometheus.yml below.



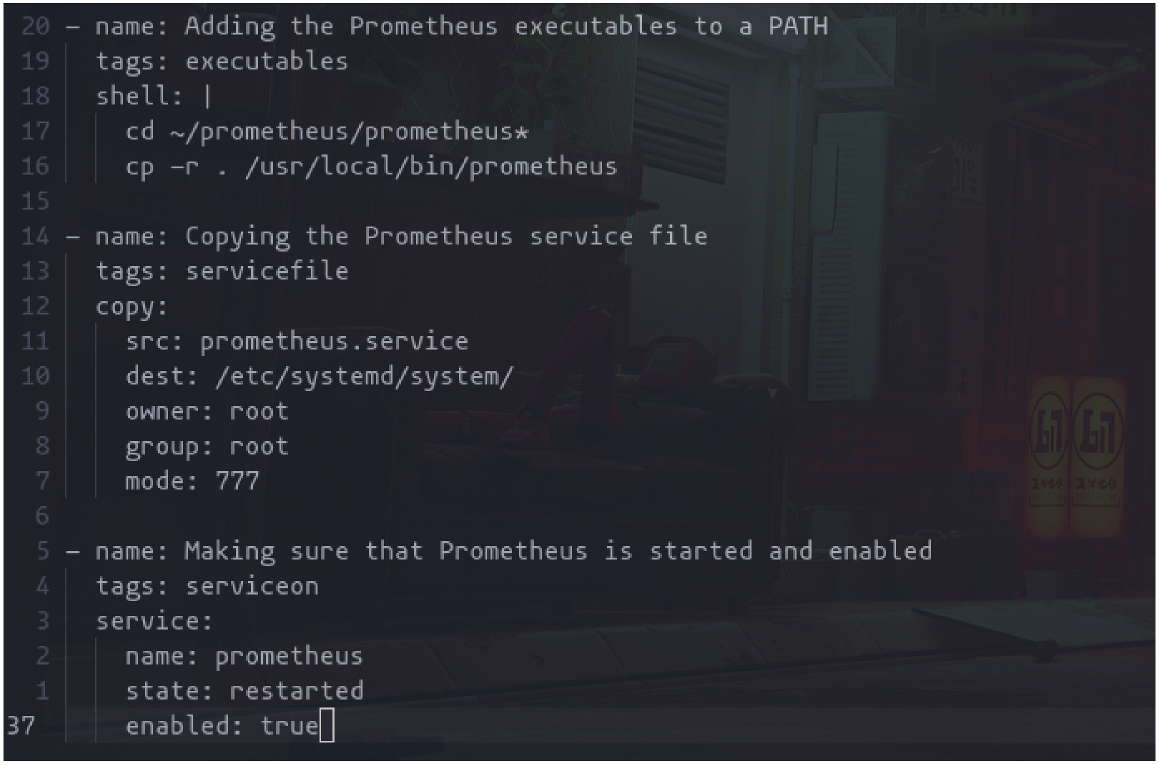
```
---

- hosts: all
  become: true
  pre_tasks:

  - name: Installing dnf and epel-release
    yum:
      name:
        - epel-release
        - dnf
    when: ansible_distribution == "CentOS"

  - name: Update and upgrade remote CentOS server
    dnf:
      update_cache: yes
      name:  "*"
      state: latest
    when: ansible_distribution == "CentOS"

  - name: Dpkg fixing in Ubuntu Servers
    shell: |
      dpkg --configure -a
    when: ansible_distribution == "Ubuntu"

  - name: Update and upgrade remote in Ubuntu servers
    apt:
      update_cache: yes
      upgrade: yes
    when: ansible_distribution == "Ubuntu"

- hosts: prometheus_centos
  become: true
  roles:
    - role: prometheus_centos
      source: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-amd64.tar.gz

- hosts: prometheus_ubuntu
  become: true
  roles:
    - role: prometheus_ubuntu
      source: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-amd64.tar.gz
```

Figure 1.7. Editing the install_nagios.yml.

STEP 6. Copy the contents of the main.yml for Centos and Ubuntu.



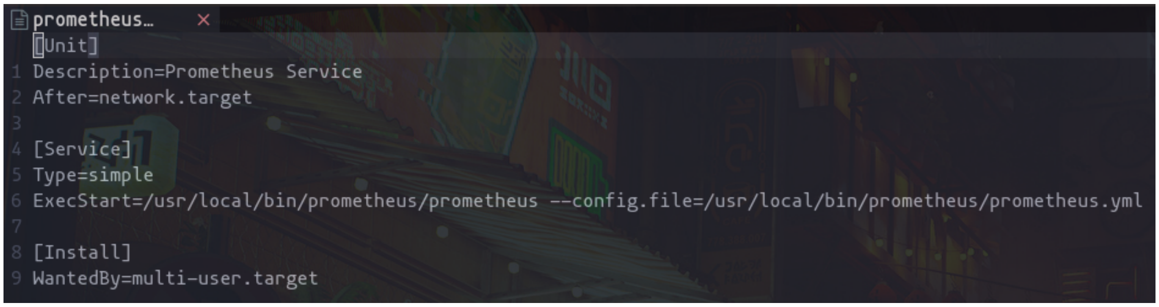
```
- name: Creating a directory (where the downloaded files will be stored)
  tags: directory
  file:
    path: ~/prometheus
    state: directory

- name: Downloading and extracting Prometheus
  tags: source
  unarchive:
    src: "{{source}}"
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root
```



```
- name: Adding the Prometheus executables to a PATH
  tags: executables
  shell: |
    cd ~/prometheus/prometheus*
    cp -r . /usr/local/bin/prometheus

- name: Copying the Prometheus service file
  tags: servicefile
  copy:
    src: prometheus.service
    dest: /etc/systemd/system/
    owner: root
    group: root
    mode: 777

- name: Making sure that Prometheus is started and enabled
  tags: serviceon
  service:
    name: prometheus
    state: restarted
    enabled: true
```

Figure 1.8. The screenshot above shows the contents of main.yml under centos.

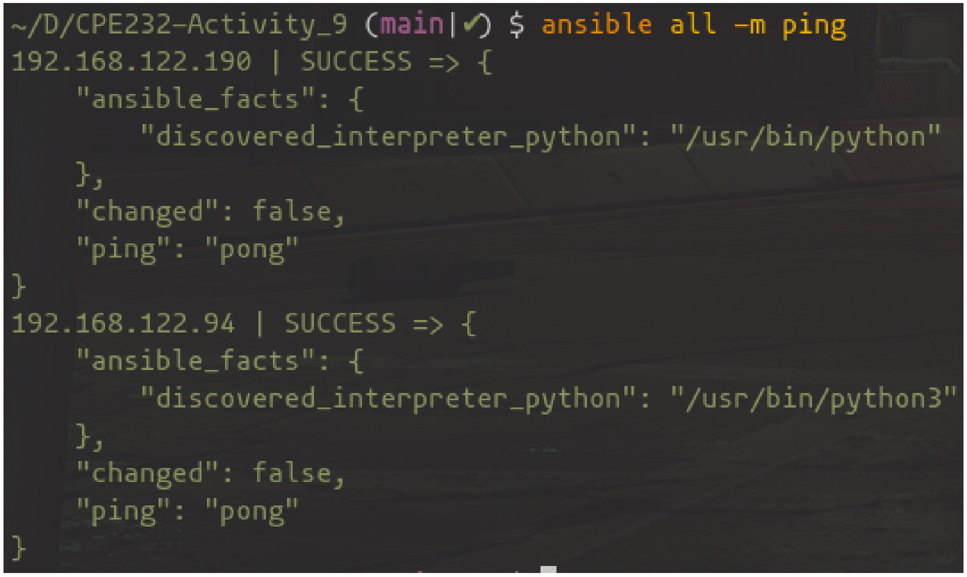
STEP 7. In the file prometheus.service, copy the contents below and paste to the file. This file is a service file that enables the prometheus service to run after booting up.



```
[Unit]
Description=Prometheus Service
After=network.target

[Service]
Type=simple
ExecStart=/usr/local/bin/prometheus/prometheus --config.file=/usr/local/bin/prometheus/prometheus.yml

[Install]
WantedBy=multi-user.target
```

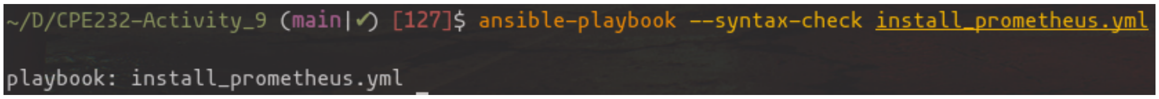
Figure 1.9. The screenshot above shows the contents of prometheus.service under files directory.

Step 8. Check the connection between the computer and the servers by running “ansible all -m ping” (Note: ssh without asking a password is required). Also, you can check you syntax by running “ansible-playbook –syntax-check playbook.yml”



```
~/D/CPE232-Activity_9 (main|✓) $ ansible all -m ping
192.168.122.190 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.122.94 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

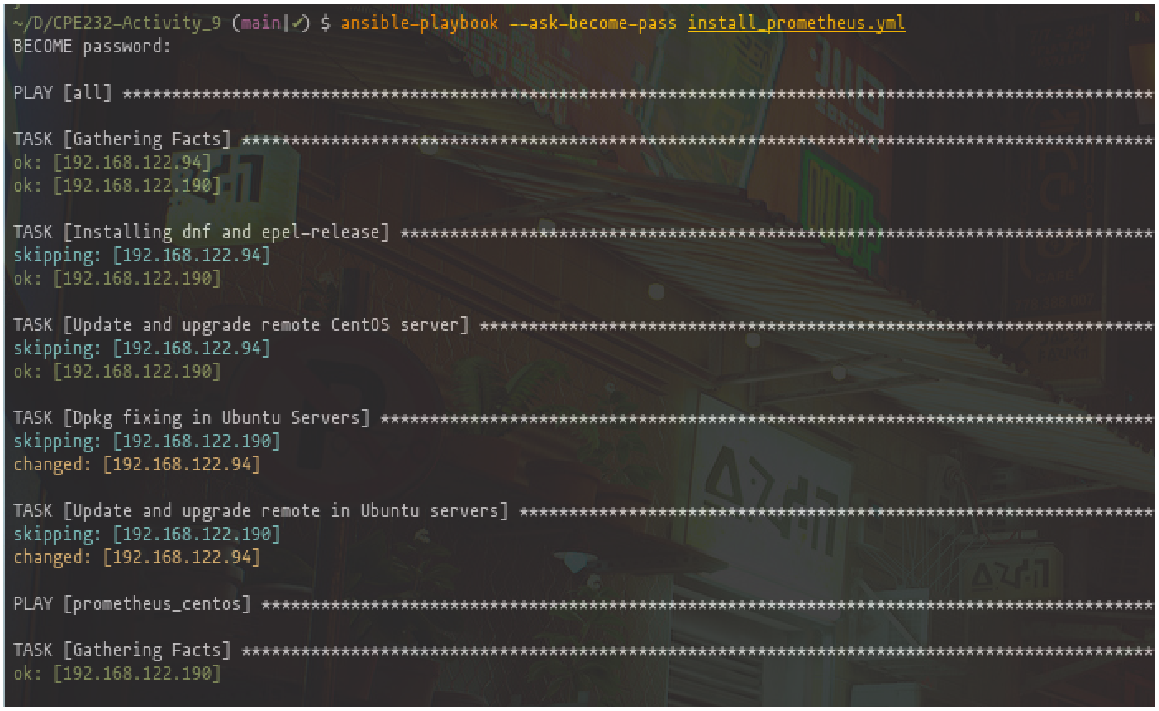
Figure 1.10. Running a test to make sure the two servers are reachable.



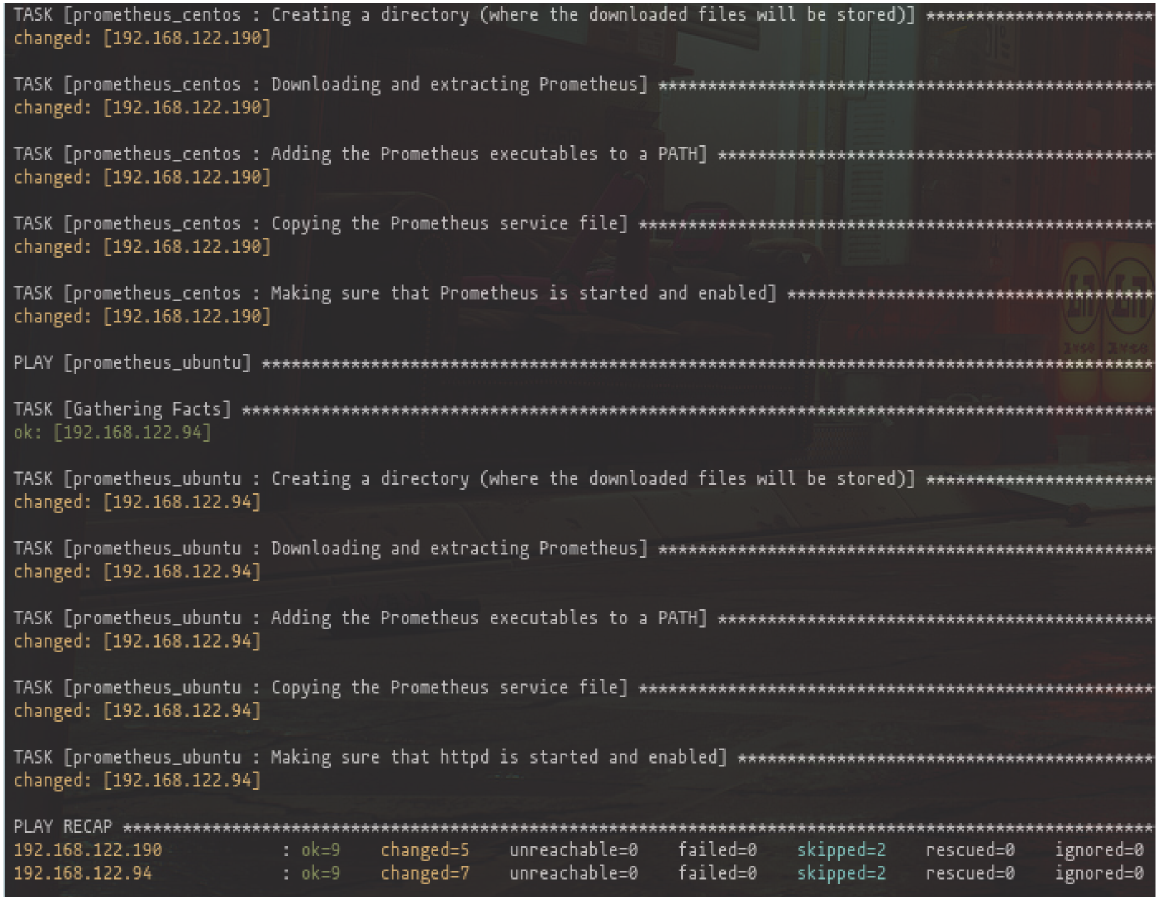
```
~/D/CPE232-Activity_9 (main|✓) [127]$ ansible-playbook --syntax-check install_prometheus.yml

playbook: install_prometheus.yml
```

Figure 1.11. Running a test to make sure the scripts does not have any syntax errors.

Step 9. Run "ansible-playbook –ask-become-pass playbook.yml" to execute the scripts to the servers.



```
~/D/CPE232-Activity_9 (main|✓) $ ansible-playbook --ask-become-pass install_prometheus.yml
BECOME password:

PLAY [all] *******************************************************************************************************

TASK [Gathering Facts] *******************************************************************************************
ok: [192.168.122.94]
ok: [192.168.122.190]

TASK [Installing dnf and epel-release] ***************************************************************************
skipping: [192.168.122.94]
ok: [192.168.122.190]

TASK [Update and upgrade remote CentOS server] *******************************************************************
skipping: [192.168.122.94]
ok: [192.168.122.190]

TASK [Dpkg fixing in Ubuntu Servers] *****************************************************************************
skipping: [192.168.122.190]
changed: [192.168.122.94]

TASK [Update and upgrade remote in Ubuntu servers] ***************************************************************
skipping: [192.168.122.190]
changed: [192.168.122.94]

PLAY [prometheus_centos] *****************************************************************************************

TASK [Gathering Facts] *******************************************************************************************
ok: [192.168.122.190]
```



```
TASK [prometheus_centos : Creating a directory (where the downloaded files will be stored)] ************************
changed: [192.168.122.190]

TASK [prometheus_centos : Downloading and extracting Prometheus] *************************************************
changed: [192.168.122.190]

TASK [prometheus_centos : Adding the Prometheus executables to a PATH] *******************************************
changed: [192.168.122.190]

TASK [prometheus_centos : Copying the Prometheus service file] ***************************************************
changed: [192.168.122.190]

TASK [prometheus_centos : Making sure that Prometheus is started and enabled] ************************************
changed: [192.168.122.190]

PLAY [prometheus_ubuntu] *****************************************************************************************

TASK [Gathering Facts] *******************************************************************************************
ok: [192.168.122.94]

TASK [prometheus_ubuntu : Creating a directory (where the downloaded files will be stored)] ************************
changed: [192.168.122.94]

TASK [prometheus_ubuntu : Downloading and extracting Prometheus] *************************************************
changed: [192.168.122.94]

TASK [prometheus_ubuntu : Adding the Prometheus executables to a PATH] *******************************************
changed: [192.168.122.94]

TASK [prometheus_ubuntu : Copying the Prometheus service file] ***************************************************
changed: [192.168.122.94]

TASK [prometheus_ubuntu : Making sure that httpd is started and enabled] *****************************************
changed: [192.168.122.94]

PLAY RECAP *******************************************************************************************************
192.168.122.190            : ok=9    changed=5    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
192.168.122.94             : ok=9    changed=7    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
```

Figure 1.12. Output of the command “ansible-playbook –ask-become-pass install_nagios.yml”.

Step 10. Verify the installation by checking both servers. In firefox or other web browser, type the IP address of the machine and the port 9090 (e.g, 192.168.122.94:9090). The port 9090 is where the Prometheus used in connecting the website.

VERIFYING PROMETHEUS IN BOTH SERVER	
Server	**Output in Firefox**

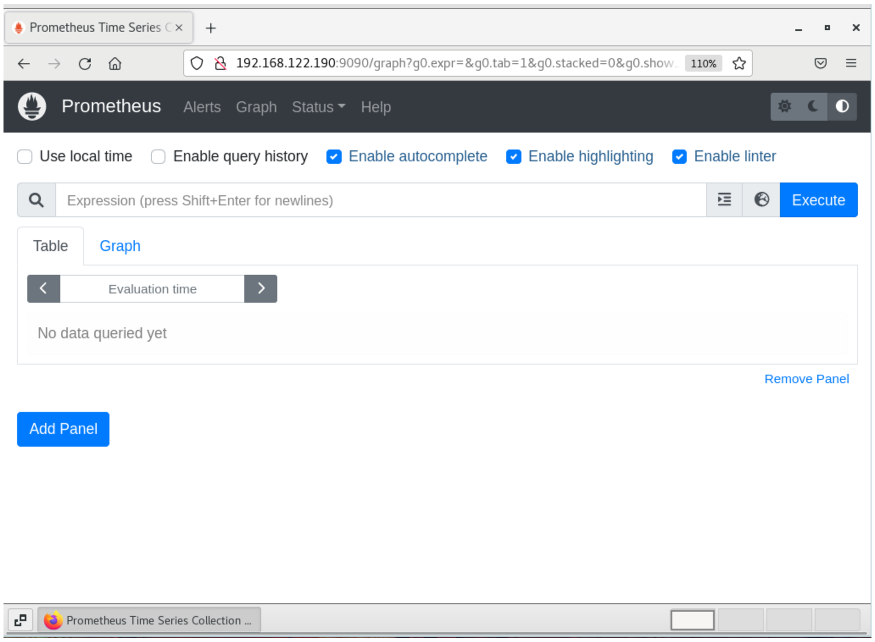
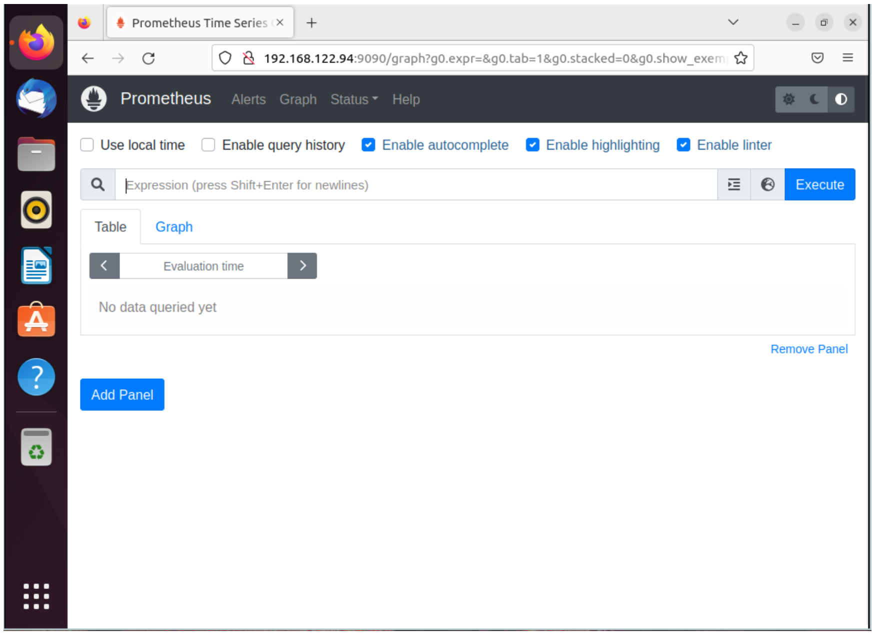
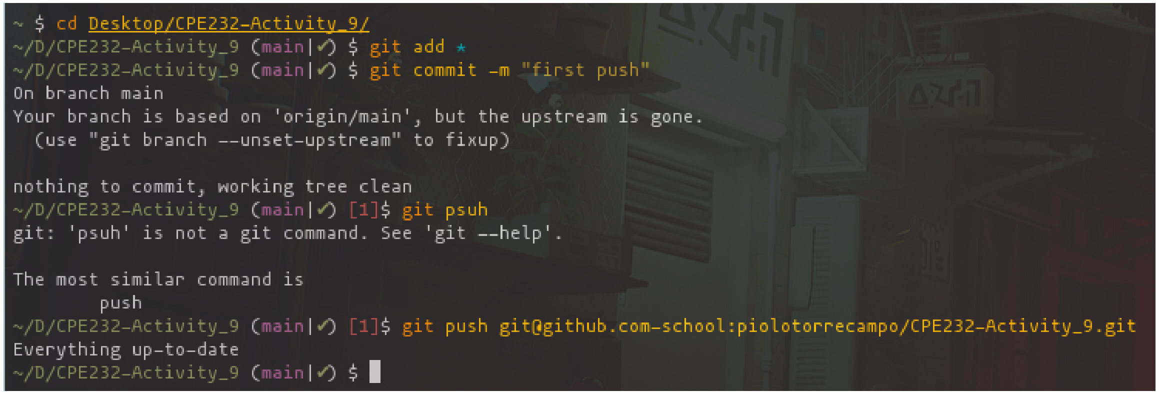
CentOS *(192.168.122.190:9090)*	 192.168.122.190:9090/graph?g0.expr=&g0.tab=1&g0.stacked=0&g0.show ☐ Use local time ☐ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter Expression (press Shift+Enter for newlines) No data queried yet Prometheus Time Series C 110% Prometheus Alerts Graph Status Help Execute Table Graph Evaluation time Remove Panel Add Panel Prometheus Time Series Collection ...
Ubuntu *(192.168.122.94:9090)*	 192.168.122.94:9090/graph?g0.expr=&g0.tab=1&g0.stacked=0&g0.show_exem ☐ Use local time ☐ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter Expression (press Shift+Enter for newlines) No data queried yet Prometheus Time Series Prometheus Alerts Graph Status Help Execute Table Graph Evaluation time Remove Panel Add Panel

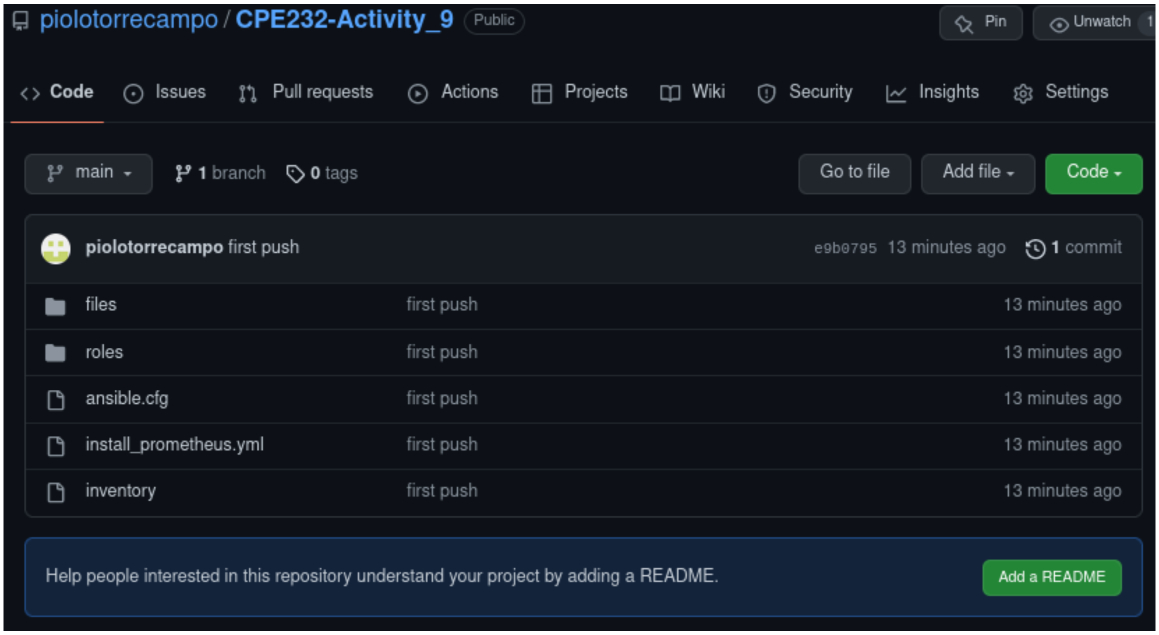
Table 1.1. The table above shows the proofs in each server.

Step 11. Update the repository.



```
~ $ cd Desktop/CPE232-Activity_9/
~/D/CPE232-Activity_9 (main|✓) $ git add *
~/D/CPE232-Activity_9 (main|✓) $ git commit -m "first push"
On branch main
Your branch is based on 'origin/main', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean
~/D/CPE232-Activity_9 (main|✓) [1]$ git psuh
git: 'psuh' is not a git command. See 'git --help'.

The most similar command is
        push
~/D/CPE232-Activity_9 (main|✓) [1]$ git push git@github.com-school:piolotorrecampo/CPE232-Activity_9.git
Everything up-to-date
~/D/CPE232-Activity_9 (main|✓) $ 
```

Figure 1.13. Pushing the updated repository into Github.



piolotorrecampo first push — e9b0795 13 minutes ago — 1 commit

files	first push	13 minutes ago
roles	first push	13 minutes ago
ansible.cfg	first push	13 minutes ago
install_prometheus.yml	first push	13 minutes ago
inventory	first push	13 minutes ago

Help people interested in this repository understand your project by adding a README.

Figure 1.14. The screenshot above shows the Github updated repository page.

Github Link: https://github.com/piolotorrecampo/CPE232-Activity_9.git
Prometheus Download Page: https://prometheus.io/download/

Reflections:

Answer the following:

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

- Having a performance monitoring tool is a life saver for the system administrations. It gives access to the system administrators to monitor the current state of the servers. This monitors whether the system has errors, enough hardware resources, processes, and many more.

One example that supports this massive feature is the Prometheus. Prometheus can detect the source of errors and alerts the system administrators and constantly monitor services per device but is not responsible for fixing bugs or errors. The Cacti is not included in the activity but upon researching, the Cacti is also considered as a monitoring tools but has a feature of creating performance management Graphs.

Conclusions:

This activity achieves its goal to introduce other enterprise performance tools that apply the use of Ansible as a medium in automating the processes in a server. In the procedure part of the activity introduces the monitoring tool called Prometheus. The first step in completing this activity is to research for a possible installation guide for a Ubuntu and CentOS server. Then convert this set of instructions into a ansible playbook that installs and runs services in both servers. Upon observation throughout this activity, I realized that the method of installing and starting the services of Prometheus in both servers are the same. Overall, this activity gives me confidence in creating ansible playbooks and to implement roles to it.

Honor Pledge:

“I affirm that I will not give or receive unauthorized help on this activity and that all will be my own.”