Name: Jessie Robert Lazo	Date Performed: 12/04/2024
Course/Section:CPE 212-CPE31S2	Date Submitted:12/04/2024
Instructor: Engr. Robin Valenzuela	Semester and SY:

Hands-on Final Exam

1. Procedure

1. 1. Create a repository and label it as "Final_Exam_Surname"

2. 2. Clone your new repository in your VM

3. 3. Create an Ansible playbook that does the following with an input of a config.yaml file and structure inventory file.

4. 3.1 Install and configure one enterprise service that can be installed in Debian and Centos servers

5. 3.2 Install and configure one monitoring tool that can be installed in Debian and Centos servers (if it is a stack there should be option of different host)

6. 4.4 Change Motd as "Ansible Managed by <username>"

7. 4. Push and commit your files in GitHub

8. 5. Make sure to show evidence of input (codes) process (codes successfully running) and output (evidence of installation)

9. 5. For your final exam to be counted, please paste your repository link as an answer in this exam.

10. Note: Extra points if you will implement the said services via containerization.

2. Output

```
jessielazo@Desktop:~$ git clone git@github.com:Jessie-Lazo/Final_Exam_Lazo.git
Cloning into 'Final_Exam_Lazo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
jessielazo@Desktop:~$
```

Installing and configuring enterprise service

Enterprise service used: Prometheus

File Edit View Search Terminal Help

  GNU nano 2.9.3   ansible.cfg

[defaults]

inventory = inventory
host_key_checking = False

deprecation_warning = False

remote_user = jessielazo
private_key_file = ~/.ssh/

File Edit View Search Terminal Help

  GNU nano 2.9.3   inventory

[ubuntu_servers]
192.168.56.102 ansible_user=jessieserver2 ansible_python_interpreter=/usr/bin/$

[centos_servers]
192.168.56.108 ansible_user=lazocentos ansible_python_interpreter=/usr/bin/pyt$

Create a playbook for enterprise service named config.yaml and implement basic update and upgrade commands for Ubuntu and CentOS machines.

```
  GNU nano 2.9.3                          config.yaml

---
- hosts: all
  become: true
  pre_tasks:

  - name: install update and repositories (CentOS)
    tags: always
    yum:
      name: "*"
      update_cache: yes
      state: latest
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: install update and repositories (Ubuntu)
    tags: always
    apt:
      upgrade: yes
      update_cache: yes
      cache_valid_time: 86400
    changed_when: false
    when: ansible_distribution == "Ubuntu"
```

Install the necessary packages for the installation of Prometheus

For Ubuntu:

```
jessielazo@Desktop: ~/Final_Exam_Lazo/roles/remote_servers_ubuntu

File  Edit  View  Search  Terminal  Help

GNU nano 2.9.3                    main.yml                    Modified

 - name: install necessary packages for Prometheus (Ubuntu)
   apt:
     name:
       - prometheus
     state: latest

 - name: Copying the Prometheus Configuration (Ubuntu)
   copy:
     src: prometheus.service
     dest: /etc/systemd/system/prometheus.service
     owner: root
     group: root
     mode: 777

 - name: Begin/Restart the Prometheus service (Ubuntu)
   service:
     name: prometheus
     state: restarted
     enabled: yes
```

For Centos:

```
jessielazo@Desktop: ~/Final_Exam_Lazo/roles/remote_servers_centos

File  Edit  View  Search  Terminal  Help

GNU nano 2.9.3                    main.yml

 - name: prothemetus download directory (CentOS)
   file:
     path: ~/prometheus
     state: directory

 - name: Downloading and extracting Prometheus (CentOS)
   unarchive:
     src: https://github.com/prometheus/prometheus/releases/download/v2.8.1/p$
     dest: ~/prometheus
     remote_src: yes
     mode: 0777
     owner: root
     group: root
```

Make the Prometheus Configuration file:

```
                    jessielazo@Desktop: ~/Final_Exam_Lazo
File  Edit  View  Search  Terminal  Help
  GNU nano 2.9.3                    prometheus.service

[Unit]
Description=Prometheus
After=network.target

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

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

Make a code for moving the pre-made prometheus configuration file to its designated path directory on the remote machine's prometheus file

For Ubuntu:

```
      state: latest

  - name: Copying the Prometheus Configuration (Ubuntu)
    copy:
      src: prometheus.service
      dest: /etc/systemd/system/prometheus.service
      owner: root
      group: root
      mode: 777
```

For Centos:

```
    - name: Copying the Prometheus Configuration (CentOS)
      copy:
        src: prometheus.service
        dest: /etc/systemd/system/prometheus.service
        owner: root
        group: root
        mode: 777
```

Lastly, begin or restart the Prometheus service

For Ubuntu:

```
      mode: 777

  - name: Begin/Restart the Prometheus service (Ubuntu)
    service:
      name: prometheus
      state: restarted
      enabled: yes
```
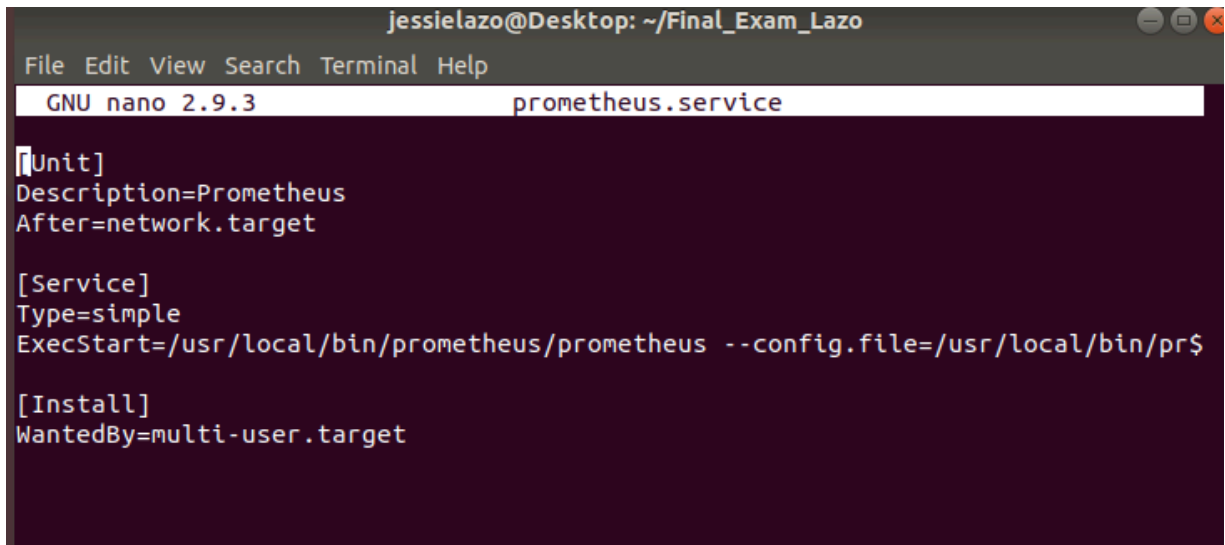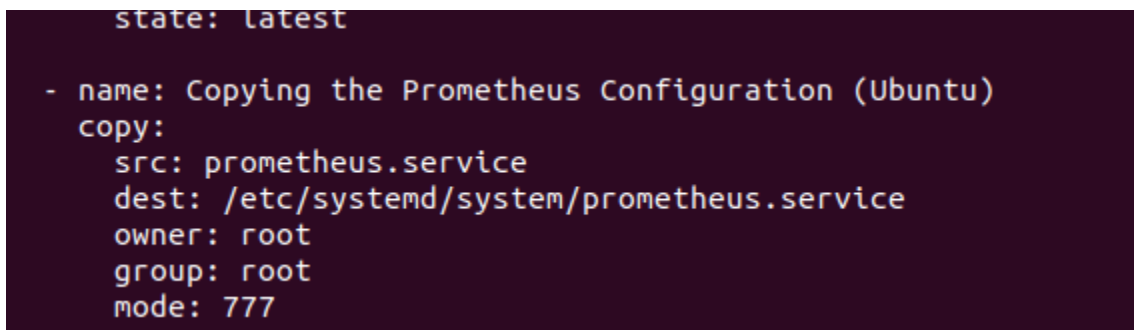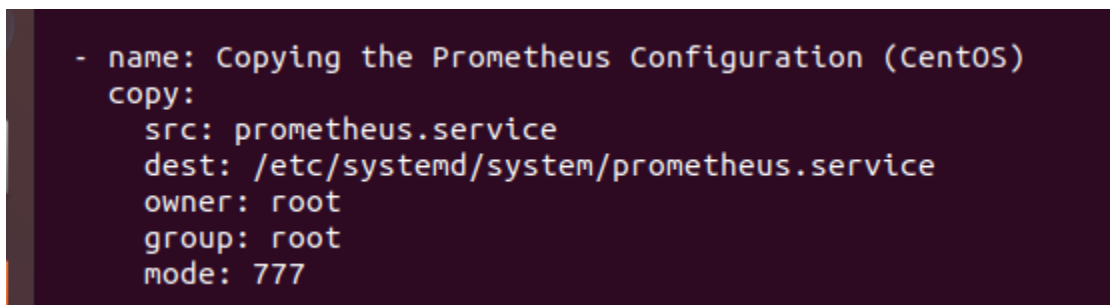
For Centos:

```
  - name: Begin/Restart the Prometheus service (CentOS)
    service:
      name: prometheus
      state: restarted
      enabled: yes
```

Installing and configuring monitoring tool:

Monitoring tool used: Nagios

I just use the same playbook and include the installation of Nagios in the remote server main.yml for ubuntu and centos.

jessielazo@Desktop: ~/Final_Exam_Lazo

File Edit View Search Terminal Help

```
  GNU nano 2.9.3                        config.yaml

---
- hosts: all
  become: true
  pre_tasks:

  - name: install update and repositories (CentOS)
    tags: always
    yum:
      name: "*"
      update_cache: yes
      state: latest
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: install update and repositories (Ubuntu)
    tags: always
    apt:
      upgrade: yes
      update_cache: yes
      cache_valid_time: 86400
    changed_when: false
    when: ansible_distribution == "Ubuntu"
```
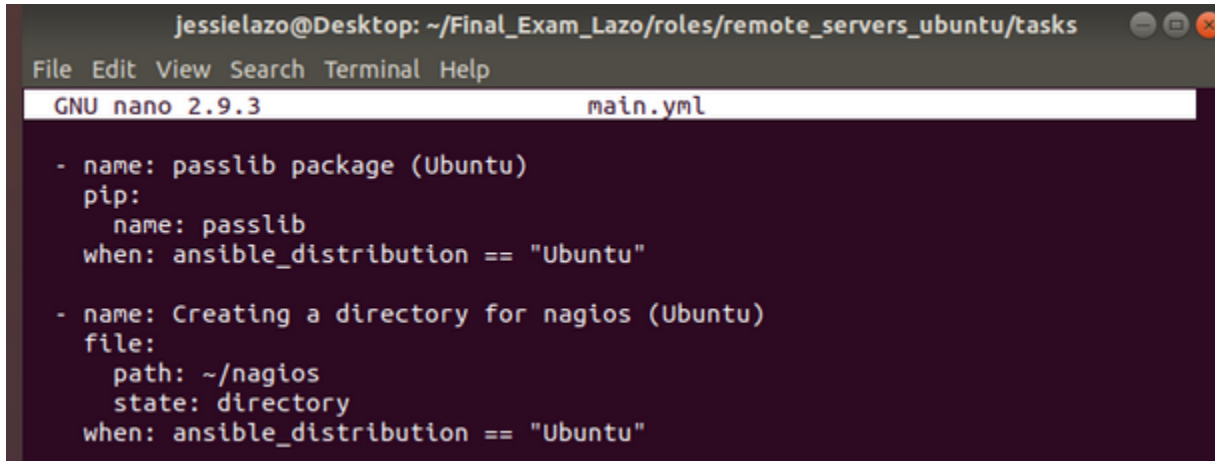
Setup Nagios including the list of dependencies which consists of libraries, services, and stuff needed for the setup of Nagios.

```
  - name: nagios libraries and dependencies (Ubuntu)
    tags: ubuntu, dependencies, libraries
    apt:
      name:
        - autoconf
        - libc6
        - gcc
        - make
        - wget
        - unzip
        - apache2
        - php
        - libapache2-mod-php
        - libgd-dev
        - openssl
```

```
      - libc6
      - gcc
      - make
      - wget
      - unzip
      - apache2
      - php
      - libapache2-mod-php
      - libgd-dev
      - openssl
      - libssl-dev
      - bc
      - gawk
      - dc
      - build-essential
      - snmp
      - libnet-snmp-perl
      - gettext
      - python3
      - python3-pip
    state: latest
  when: ansible_distribution == "Ubuntu"
```

Create a directory for storing the Nagios files and plugins which would later be downloaded
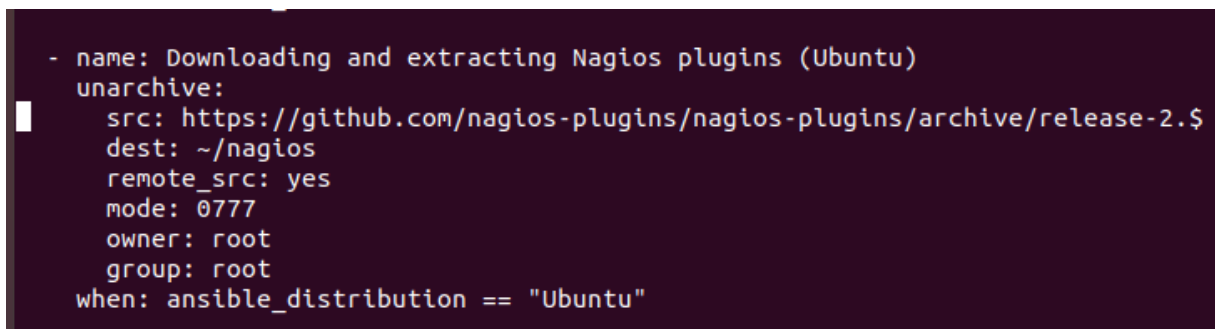
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                          main.yml

  - name: passlib package (Ubuntu)
    pip:
      name: passlib
    when: ansible_distribution == "Ubuntu"

  - name: Creating a directory for nagios (Ubuntu)
    file:
      path: ~/nagios
      state: directory
    when: ansible_distribution == "Ubuntu"
```

After creating a directory, we can start downloading and extracting Nagios files and Plugins.

```
  - name: Downloading and extracting Nagios plugins (Ubuntu)
    unarchive:
      src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.$
      dest: ~/nagios
      remote_src: yes
      mode: 0777
      owner: root
      group: root
    when: ansible_distribution == "Ubuntu"
```

When Nagios files and Plugins are already extracted, we can begin the installation and additional procedures.

```
 - name: install and compile in Nagios (Ubuntu)
   shell: |
     cd ~/nagios/nagioscore-*
     sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
     sudo make all
     sudo make install-groups-users
     sudo usermod -a -G nagios www-data
     sudo make install
     sudo make install-daemoninit
     sudo make install-commandmode
     sudo make install-config
     sudo make install-webconf
     sudo a2enmod rewrite
     sudo a2enmod cgi
   when: ansible_distribution == "Ubuntu"


^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify
^X Exit          ^R Read File    ^\ Replace     ^U Uncut Text  ^T To Spell
```

After running the necessary installation and configurations of Nagios files and plugins, we have to make sure the Nagios and apache/httpd services are in UP state.

```
File  Edit  View  Search  Terminal  Help
  GNU nano 2.9.3                          main.yml

  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install
  when: ansible_distribution == "Ubuntu"

- name: starting/enabling nagios (Ubuntu)
  service:
    name: nagios
    state: restarted
    enabled: true
  when: ansible_distribution == "Ubuntu"

- name: starting/enabling apache2 (Ubuntu)
  service:
    name: apache2
    state: restarted
    enabled: true
  when: ansible_distribution == "Ubuntu"

^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify
```

For Centos:

```yaml
  - name: Installing nagios dependecies and libraries (CentOS)
    tags: dependecies, libraries
    yum:
      name:
        - gcc
        - glibc
        - glibc-common
        - perl
        - httpd
        - php
        - wget
        - gd
        - gd-devel
        - openssl-devel
        - gcc
        - glibc
        - glibc-common
```

File Edit View Search Terminal Help

GNU nano 2.9.3   main.yml

```yaml
        - glibc-common
        - perl
        - httpd
        - php
        - wget
        - gd
        - gd-devel
        - openssl-devel
        - gcc
        - glibc
        - glibc-common
        - make
        - gettext
        - automake
        - autoconf
        - wget
        - openssl-devel
        - net-snmp
        - net-snmp-utils
        - python3-pip
      state: present
    when: ansible_distribution == "CentOS"
```

```yaml
  - name: Install passlib python package (CentOS)
    pip:
      executable: /usr/bin/pip3
      name: passlib
    when: ansible_distribution == "CentOS"

  - name: Creating a directory for nagios (CentOS)
    file:
      path: ~/nagios
      state: directory
    when: ansible_distribution == "CentOS"
```

```yaml
  - name: Downloading and extracting Nagios (CentOS)
    unarchive:
      src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.$
      dest: ~/nagios
      remote_src: yes
      mode: 0777
      owner: root
      group: root
    when: ansible_distribution == "CentOS"
```

```yaml
  - name: Downloading and extracting Nagios plugins (CentOS)
    unarchive:
      src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.$
      dest: ~/nagios
      remote_src: yes
      mode: 0777
      owner: root
      group: root
    when: ansible_distribution == "CentOS"
```

```yaml
  - name: Compiling, installing, and adding users and groups in nagios (CentOS)
    shell: |
      cd ~/nagios/nagioscore-**
      ./configure
      make all
      make install-groups-users
      usermod -a -G nagios apache
      make install
      make install-daemoninit
      make install-commandmode
      make install-config
      make install-webconf
    when: ansible_distribution == "CentOS"
```

```
  GNU nano 2.9.3                          main.yml


      ./tools/setup
      ./configure
      make
      make install
    when: ansible_distribution == "CentOS"

  - name: starting/enabling nagios (CentOS)
    service:
      name: nagios
      state: restarted
      enabled: true
    when: ansible_distribution == "CentOS"

  - name: starting/enabling apache2 (CentOS)
    service:
      name: httpd
      state: restarted
      enabled: true
    when: ansible_distribution == "CentOS"
```

Running Playbook:

Prometheus:

```
jessielazo@Desktop:~/Final_Exam_Lazo$ ansible-playbook --ask-become-pass config
.yaml
SUDO password:

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]
ok: [192.168.56.108]

TASK [install update and repositories (CentOS)] *******************************
*
skipping: [192.168.56.102]
ok: [192.168.56.108]

TASK [install update and repositories (Ubuntu)] *******************************
*
skipping: [192.168.56.108]
 [WARNING]: Could not find aptitude. Using apt-get instead.

ok: [192.168.56.102]

PLAY [ubuntu_servers] *********************************************************
*
```

```
TASK [Gathering Facts] *****************************************************
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : install necessary packages for Prometheus (Ubuntu
)] ***
changed: [192.168.56.102]

TASK [remote_servers_ubuntu : Copying the Prometheus Configuration (Ubuntu)] **
*
changed: [192.168.56.102]

TASK [remote_servers_ubuntu : Begin/Restart the Prometheus service (Ubuntu)] **
*
changed: [192.168.56.102]

PLAY [centos_servers] *****************************************************
*

TASK [Gathering Facts] *****************************************************
*
ok: [192.168.56.108]

TASK [remote_servers_centos : prothemetus download directory (CentOS)] ********
*
```

Nagios:
Ubuntu:

```
TASK [remote_servers_ubuntu : nagios libraries and dependencies (Ubuntu)] *****
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : passlib package (Ubuntu)] ***********************
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : Creating a directory for nagios (Ubuntu)] *******
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios (Ubuntu)] *****
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios plugins (Ubuntu
)] ***
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : install and compile in Nagios (Ubuntu)] *********
*
changed: [192.168.56.102]
```

jessielazo@Desktop: ~/Final_Exam_Lazo

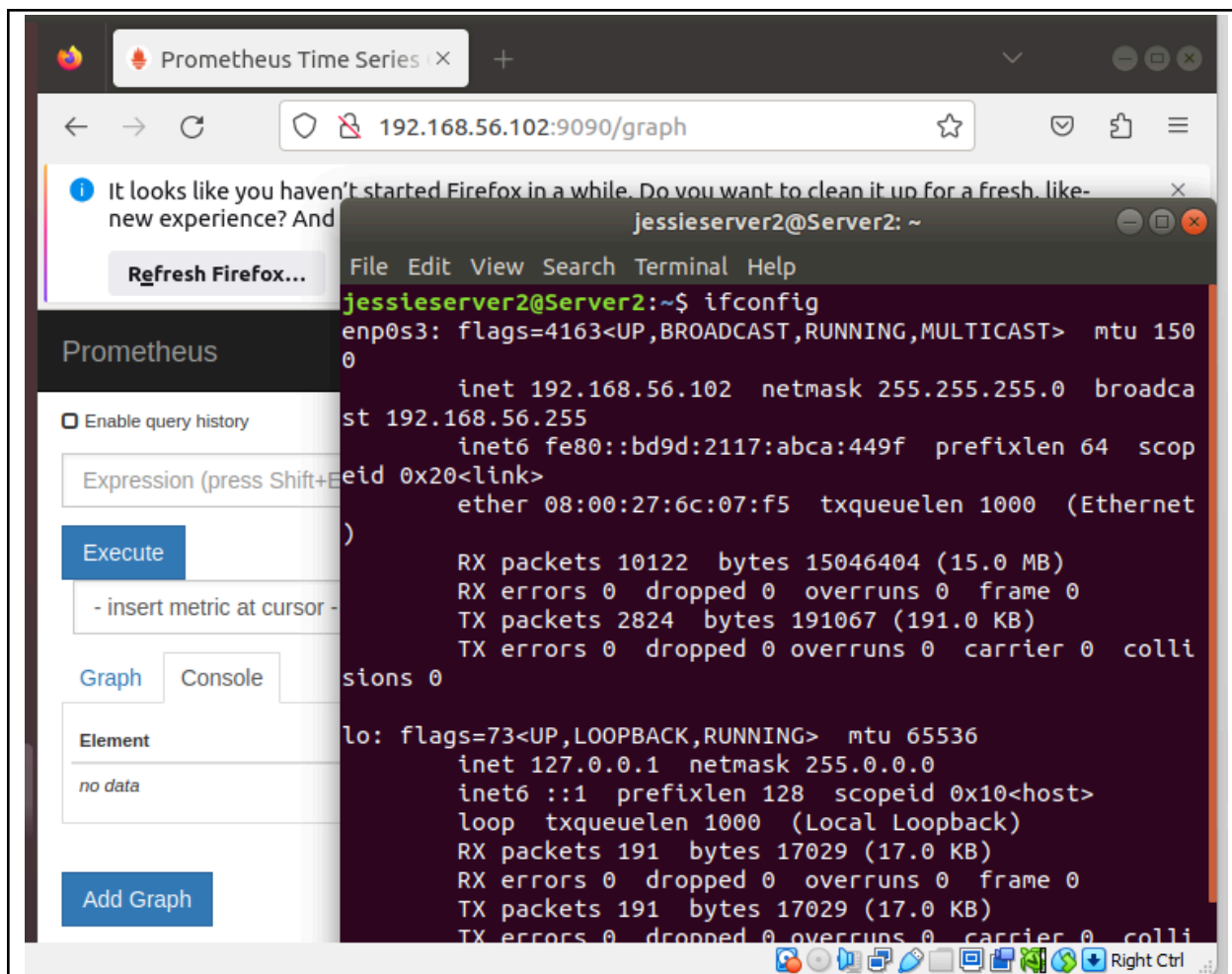File Edit View Search Terminal Help

```
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios (Ubuntu)] *****
*
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios plugins (Ubuntu
)] ***
ok: [192.168.56.102]

TASK [remote_servers_ubuntu : install and compile in Nagios (Ubuntu)] *********
*
changed: [192.168.56.102]

TASK [remote_servers_ubuntu : compile and install plugins (Ubuntu)] ***********
*
changed: [192.168.56.102]

TASK [remote_servers_ubuntu : starting/enabling nagios (Ubuntu)] **************
*
changed: [192.168.56.102]

TASK [remote_servers_ubuntu : starting/enabling apache2 (Ubuntu)] *************
*
changed: [192.168.56.102]
```

Centos:

```
TASK [remote_servers_centos : Installing nagios dependecies and libraries (Cent
OS)] ***
ok: [192.168.56.108]

TASK [remote_servers_centos : Install passlib python package (CentOS)] ********
*
ok: [192.168.56.108]

TASK [remote_servers_centos : Creating a directory for nagios (CentOS)] *******
*
ok: [192.168.56.108]

TASK [remote_servers_centos : Downloading and extracting Nagios (CentOS)] *****
*
ok: [192.168.56.108]

TASK [remote_servers_centos : Downloading and extracting Nagios plugins (CentOS
)] ***
ok: [192.168.56.108]

TASK [remote_servers_centos : Compiling, installing, and adding users and group
s in nagios (CentOS)] ***
changed: [192.168.56.108]
```
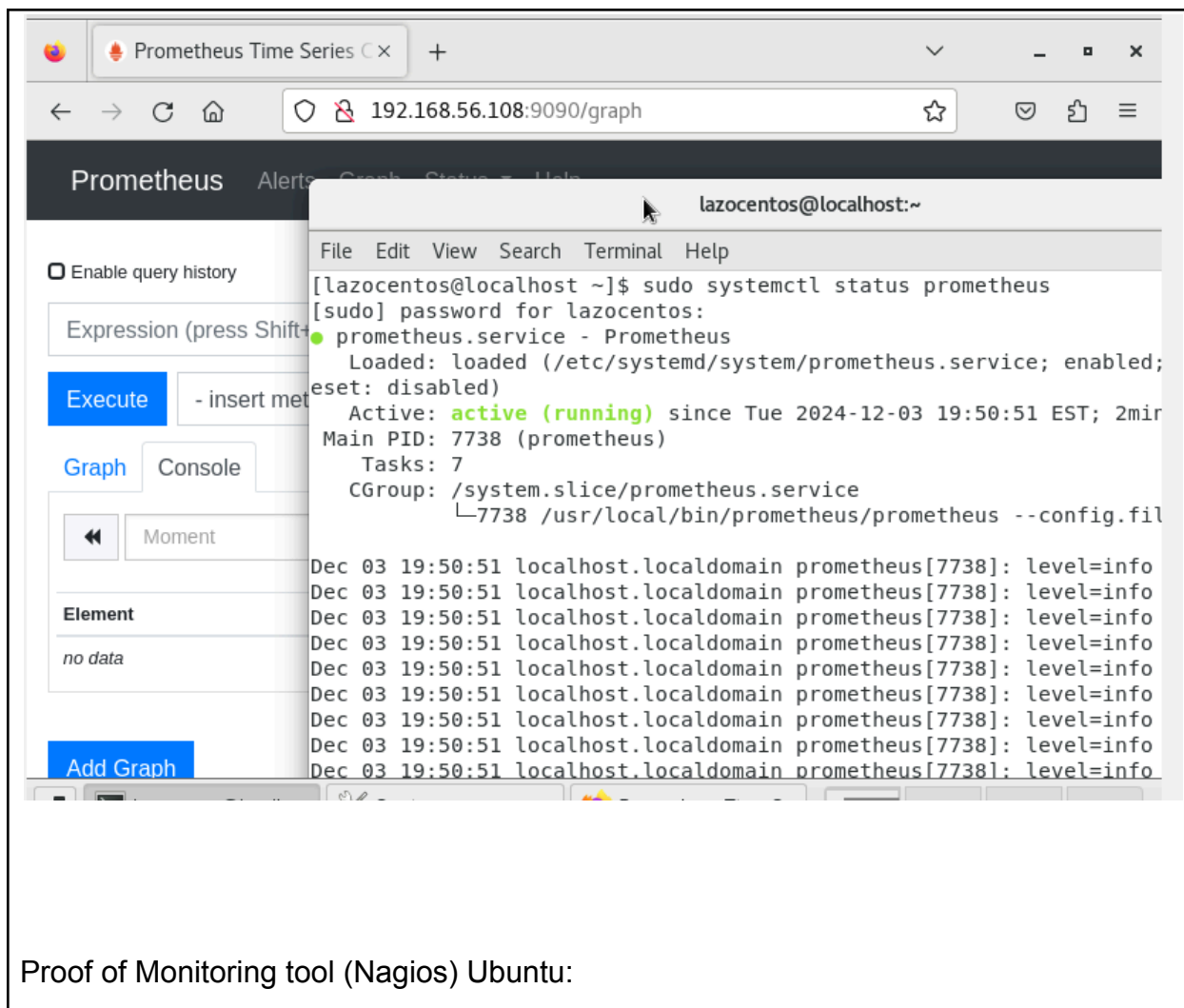
jessielazo@Desktop: ~/Final_Exam_Lazo

File Edit View Search Terminal Help

```
ok: [192.168.56.108]

TASK [remote_servers_centos : Downloading and extracting Nagios plugins (CentOS
)] ***
ok: [192.168.56.108]

TASK [remote_servers_centos : Compiling, installing, and adding users and group
s in nagios (CentOS)] ***
changed: [192.168.56.108]

TASK [remote_servers_centos : Compiling and installing plugins (CentOS)] ******
*
changed: [192.168.56.108]

TASK [remote_servers_centos : starting/enabling nagios (CentOS)] **************
*
changed: [192.168.56.108]

TASK [remote_servers_centos : starting/enabling apache2 (CentOS)] *************
*
changed: [192.168.56.108]

PLAY RECAP *********************************************************************
*
192.168.56.102             : ok=15    changed=5    unreachable=0    failed=0
192.168.56.108             : ok=18    changed=7    unreachable=0    failed=0
```

Proof of enterprise service(Prometheus) for Ubuntu:

For of enterprise service Centos:

Proof of Monitoring tool (Nagios) Ubuntu:

For Centos:

```
                    lazocentos@localhost:~              _  ▢  ✕

File  Edit  View  Search  Terminal  Help
Dec 03 19:50:51 localhost.localdomain prometheus[7738]: level=info ts=2024...
Dec 03 19:50:51 localhost.localdomain prometheus[7738]: level=info ts=2024...
Dec 03 19:50:51 localhost.localdomain prometheus[7738]: level=info ts=2024...
Dec 03 19:50:51 localhost.localdomain prometheus[7738]: level=info ts=2024...
Dec 03 19:50:51 localhost.localdomain prometheus[7738]: level=info ts=2024...
Hint: Some lines were ellipsized, use -l to show in full.
[lazocentos@localhost ~]$ SSsudo
bash: SSsudo: command not found...
[lazocentos@localhost ~]$ sudo systemctl status nagios
[sudo] password for lazocentos:
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; vendor pr
eset: disabled)
   Active: active (running) since Tue 2024-12-03 20:19:13 EST; 16min ago
     Docs: https://www.nagios.org/documentation
  Process: 8543 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/e
tc/nagios.cfg (code=exited, status=0/SUCCESS)
  Process: 8542 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagio
s/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 8545 (nagios)
    Tasks: 6
   CGroup: /system.slice/nagios.service
           └─8545 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/na...
```

GithubLink: https://github.com/Jessie-Lazo/Final_Exam_Lazo

Reflections:

Monitoring tool is very important to use, it is like being a networking administrator on top of all your remote machines where you can easily see which specific areas have a problem such as what services. This is what the Nagios do in an enterprise, it eases the monitoring work between remote machines especially in a large enterprise.On the other hand, using Prometheus to monitor performance on Ubuntu and CentOS machines provides a versatile way to collect, save, and study system and app data. This helps system administrators take early action to oversee and solve issues so that the system works well and stays dependable.