Name: Frias, Abegail L.	**Date Performed: Dec. 13, 2024**
Course/Section: CPE212 - CPE31S21	**Date Submitted: Dec. 13, 2024**
Instructor: Engr. Robin Valenzuela	**Semester and SY: 1st Sem/2024-2025**

Final Exam

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

2. Clone your new repository in your VM

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

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

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)

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

4. Push and commit your files in GitHub

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

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

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

1. **Screenshots**

```
abegailfrias@workstation:~/Final_Exam_frias$ cat ansible.cfg
[defaults]
inventory = inventory
remote_user = abegailfrias
host_key_checking = True
```

- This is what is inside my ansible.cfg.

```
abegailfrias@workstation:~/Final_Exam_frias$ cat inventory
[Ubuntu]
192.168.56.107
[CentOS]
192.168.56.105
```

- And this is what is inside my inventory.

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

  - name: Update the repository Index (CentOS)
    tags: always
    yum:
      name: "*"
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: Update the repository index (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

  - name: Banner MOTD
    copy:
      content: "Ansible managed by Frias \n"
      dest: /etc/motd

- hosts: all
  become: true
  roles:
      - prometheus
```

```
  - name: Banner MOTD
    copy:
      content: "Ansible managed by Frias \n"
      dest: /etc/motd

- hosts: all
  become: true
  roles:
      - promentheus
      - apache2
```

- This is my main installer. This is what is inside my installer.yml.

```
abegailfrias@workstation:~/Final_Exam_frias/roles/apache2/tasks$ cat main.yml
---
  - name: Install apache2 for Ubuntu
    apt:
      name: apache2
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: Install PHP for Ubuntu
    apt:
      name: libapache2-mod-php
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: Install apache2 for CentOS
    yum:
      name: httpd
      state: latest
    when: ansible_distribution == "CentOS"

  - name: Install PHP packages for CentOS
    yum:
      name: php
      state: latest
    when: ansible_distribution == "CentOS"
```

- I created a roles in my repository and then I created a tasks inside the apache2. This is the playbook for the apache2.

```
abegailfrias@workstation:~/Final_Exam_frias/roles/promentheus/tasks$ cat main.yml
---

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

- 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
      de: 0755
      ote_src: yes
  when: ansible_distribution == "CentOS"

- name: Copy Promtool binaries
  copy:
```

Terminal

```
- 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: 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:
```

Terminal

```
- 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
```

```
- name: Start Prometheus Service (Ubuntu)
  systemd:
       name: prometheus
       enabled: yes
       state: started
       ansible_distribution == "Ubuntu"
Trash
frias@workstation:~/Final_Exam_frias/roles/prometheus/tasks$
```

- Inside the roles I made the prometheus and created tasks. Theis what is inside my playbook for the prometheus.

```
abegailfrias@workstation:~/Final_Exam_frias$ ansible-playbook --ask-become-pass install.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.107]
[DEPRECATION WARNING]: Distribution centos 9 on host 192.168.56.105 should use
/usr/libexec/platform-python, but is using /usr/bin/python for backward compatibility with
prior Ansible releases. A future Ansible release will default to using the discovered platform
python for this host. See
https://docs.ansible.com/ansible/2.10/reference_appendices/interpreter_discovery.html for more
information. This feature will be removed in version 2.12. Deprecation warnings can be disabled
 by setting deprecation_warnings=False in ansible.cfg.
ok: [192.168.56.105]

TASK [Update repository Index (CentOS)] ****************************************************
skipping: [192.168.56.107]
fatal: [192.168.56.105]: FAILED! => {"changed": false, "failures": [], "msg": "Depsolve Error oc
cured: \n Problem 1: conflicting requests\n  - nothing provides perl(Switch) needed by awstats-7
.8-9.el9.noarch from epel\n Problem 2: conflicting requests\n  - nothing provides perl(DateTime)
 needed by munin-2.0.76-3.el9.noarch from epel\n  - nothing provides perl(Params::Validate) need
ed by munin-2.0.76-3.el9.noarch from epel\n  - nothing provides perl(File::Copy::Recursive) need
ed by munin-2.0.76-3.el9.noarch from epel\n  - nothing provides perl(Taint::Runtime) needed by m
unin-2.0.76-3.el9.noarch from epel\n Problem 3: conflicting requests\n  - nothing provides perl(
Path::Tiny) needed by smokeping-2.8.2-10.el9.noarch from epel\n Problem 4: conflicting requests\
n  - nothing provides pkgconfig(libxklavier) needed by caribou-devel-0.4.21-33.el9.x86_64 from e
pel\n  - nothing provides pkgconfig(gee-0.8) needed by caribou-devel-0.4.21-33.el9.x86_64 from e
-stream-release-9.0-26.el9.noarch from @System", "rc": 1, "results": []}

TASK [Update repository index (Ubuntu)] ****************************************************
ok: [192.168.56.107]

TASK [Banner MOTD] *************************************************************************
ok: [192.168.56.107]

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.107]

TASK [promentheus : Install Prometheus (Ubuntu)] *******************************************
ok: [192.168.56.107]

TASK [promentheus : Install Prometheus (CentOS)] *******************************************
skipping: [192.168.56.107]

TASK [promentheus : Copy Prometheus binaries] **********************************************
skipping: [192.168.56.107]

TASK [promentheus : Copy Promtool binaries] ************************************************
skipping: [192.168.56.107]

TASK [promentheus : Create Prometheus directories] *****************************************
skipping: [192.168.56.107] => (item=/etc/prometheus)
skipping: [192.168.56.107] => (item=/var/lib/prometheus)
```

```
skipping: [192.168.56.107] => (item=/var/lib/prometheus)

TASK [promentheus : Copy prometheus.yml to /etc/prometheus] ***********************************
skipping: [192.168.56.107]

TASK [promentheus : Copy consoles directory to /etc/prometheus] ******************************
skipping: [192.168.56.107]

TASK [promentheus : Copy console_libraries directory to /etc/prometheus] *********************
skipping: [192.168.56.107]

TASK [promentheus : Create prometheus.service file] ****************************************
skipping: [192.168.56.107]

TASK [promentheus : Reload systemd] ********************************************************
skipping: [192.168.56.107]

TASK [promentheus : Start Prometheus Service] **********************************************
skipping: [192.168.56.107]

TASK [promentheus : Start Prometheus Service (Ubuntu)] *************************************
changed: [192.168.56.107]

TASK [apache2 : Install apache2 for Ubuntu] ************************************************
ok: [192.168.56.107]

TASK [apache2 : Install PHP for Ubuntu] ****************************************************
ok: [192.168.56.107]

TASK [apache2 : Install apache2 for CentOS] ************************************************
```

```
TASK [promentheus : Start Prometheus Service (Ubuntu)] *************************************
changed: [192.168.56.107]

TASK [apache2 : Install apache2 for Ubuntu] ************************************************
ok: [192.168.56.107]

TASK [apache2 : Install PHP for Ubuntu] ****************************************************
ok: [192.168.56.107]

TASK [apache2 : Install apache2 for CentOS] ************************************************
skipping: [192.168.56.107]

TASK [apache2 : Install PHP packages for CentOS] *******************************************
skipping: [192.168.56.107]

PLAY RECAP *********************************************************************************
192.168.56.105             : ok=1    changed=0    unreachable=0    failed=1    skipped=0    resc
ued=0    ignored=0
192.168.56.107             : ok=8    changed=1    unreachable=0    failed=0    skipped=13   resc
ued=0    ignored=0
```

- Running my install.yml. I still have the same problem which is my CentOS but Ubuntu is working.

Validation of the installation:

Ubuntu

```
abegailfrias@server1:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2024-12-12 14:40:07 +08; 10h ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 775 (apache2)
      Tasks: 6 (limit: 2270)
     Memory: 16.2M
        CPU: 1.723s
     CGroup: /system.slice/apache2.service
             ├─  775 /usr/sbin/apache2 -k start
             ├─18466 /usr/sbin/apache2 -k start
             ├─18467 /usr/sbin/apache2 -k start
             ├─18468 /usr/sbin/apache2 -k start
             ├─18469 /usr/sbin/apache2 -k start
             └─18470 /usr/sbin/apache2 -k start

Warning: some journal files were not opened due to insufficient permissions.
abegailfrias@server1:~$
```

2. Conclusion

- In conclusion to this final exam, I'm still having a problem with my centos. I did my best to fix the problem but it seems like i'm doing it wrong but my ubuntu is working. And also by enabling http requests from other users to send them the desired information in the form of files and web pages while maintaining security, an enterprise service's creator can deploy a particular application or web app. Meanwhile, the monitoring tool helps troubleshooters and server administrators stay in touch with the server's workflow to ensure that there are no security flaws, and that the server is well-run and efficient.

Github Link:

https://github.com/wonbe/Final_Exam_frias/tree/mainf