Name: Aaron Jonathan G. Valencia	**Date Performed: 11/06/2024**
Course/Section: CPE212/CPE31S2	**Date Submitted: 11/06/2024**
Instructor: Robin Valenzuela	**Semester and SY: 1st sem 2023-2024**

Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools

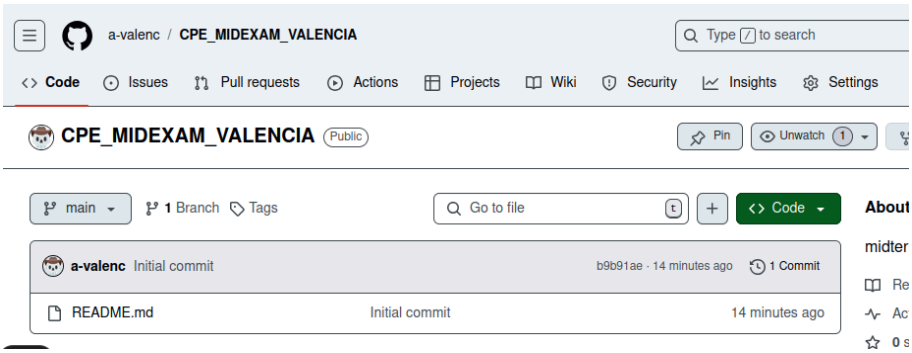
1. Objectives

Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Instructions

1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.
2. Clone the repository and do the following:
   2.1. Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:
   2.2. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash) • Install Nagios in one host
   2.3. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)
   2.4. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)
3. Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.
4. Document the push and commit from the local repository to GitHub.
5. Finally, paste also the link of your GitHub repository in the documentation.

3. Output (screenshots and explanations)

1. 

2.1

2.2

- Elasticsearch installation codes
- Elasticsearch running

- Kibana installation codes
- Kibana running
- Logstash installation codes
- Logstash running
- Nagios installation codes

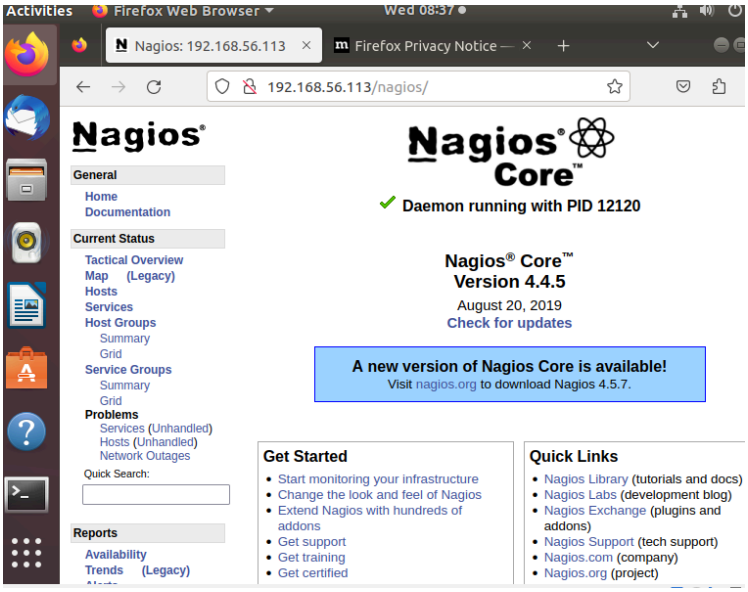
```
---
- hosts: Ubuntu
  become: true
  tasks:
  - name: removing old nagios dir
    file:
      path: /tmp/nagios
      state: absent
    become: true
  - name: removing old nagios dir pt2
    file:
      path: /tmp/nagios-4.4.5
      state: absent
    become: true
  - name: removing old nagios archive
    file:
      path: /tmp/nagios.tar.gz
      state: absent
    become: true
  - name: refresh apt
    apt:
      name: '*'
      update_cache: yes
    become: true
  - name: install nagios prereqs
    apt:
      name:
        - autoconf
        - gcc
        - libc6
        - make
        - wget
        - unzip
        - libssl-dev
        - apache2
```

```
        - php
        - libapache2-mod-php*
        - libgd-dev
        - build-essential
      state: latest
      force: true
      update_cache: true
    become: true
  - name: download and uncompress file
    shell: cd /tmp;rm nagios*.tar.gz;wget -O nagios.tar.gz --no-check-cer
    become: true
  - name: configure nagios
    shell: cd /tmp/nagios;./configure --with-mail=/usr/sbin/sendmail --wi
    become: true
  - name: create users and groups
    command: chdir=/tmp/nagios make install-groups-users
    become: true
  - name: create nagios user homedir
    file:
      path: /home/nagios
      state: directory
      owner: nagios
      group: nagios
      mode: 0775
    become: true
  - name: change nagios user home dir and add group members
    command: "{{ item }}"
    with_items:
      - usermod --home /home/nagios nagios
      - usermod -a -G nagios www-data
    become: true
  - name: make binaries
```

```
  command: chdir=/tmp/nagios {{ item }}
  with_items:
    - make install
    - make install-daemoninit
    - make install-commandmode
    - make install-config
    - make install-webconf
  become: true
- name: Enable apache2 modules
  apache2_module: state=present name={{ item }}
  with_items:
    - cgi
    - rewrite
- name: Restart Apache2 service
  service:
Software ': apache2
    state: restarted
  become: true
- name: Create nagiosadmin User account
  command: htpasswd -cb /usr/local/nagios/etc/htpasswd.users nagiosadmin admin
  become: true
- name: copy event handlers
  synchronize:
    src: /tmp/nagios/contrib/eventhandlers/
    dest: /usr/local/nagios/libexec
    recursive: yes
    mode: push
  delegate_to: Ubuntu
- name: change ownership and permisions
  file:
    path: /usr/local/nagios/libexec/eventhandlers/
    owner: nagios
    group: nagios
    mode: u+rw,g+rw,o+r
```

```
    recurse: true
  become: true
- name: Start Nagios service
  service:
    name: nagios
    state: restarted
  become: true
```

- Nagios running in Ubuntu Host



2.3

- Grafana
- Prometheus

```
avalencia@workstation:~/CPE_MIDEXAM_VALENCIA$ ansible-playbook
BECOME password:

PLAY [Ubuntu] ***********************************************

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

TASK [Create an installation directory] *********************
ok: [Ubuntu]

TASK [Download & Extract Prometheus from site] **************
ok: [Ubuntu]

Help
Create Prometheus script file] ************************
buntu]

TASK [Create a Service file for Prometheus with Copy module] **
ok: [Ubuntu]

TASK [Systemctl daemon-reload after creation] ***************
changed: [Ubuntu]

TASK [Start Prometheus] *************************************
changed: [Ubuntu]
```

-
- Influxdb

2.4

- LAMP

GitHub link:

https://github.com/a-valenc/CPE_MIDEXAM_VALENCIA

Conclusions: (link your conclusion from the objective)

We are tasked to install and configure many server tools while using Ansible playbook so it will be easier to apply to many servers easily.