Name: Tamayo, Ray Lan A.	Date Performed: 11/08/2024
Course/Section: CPE212-CPE31S21	Date Submitted: 11/08/2024
Instructor: Engr. Robin Valenzuela	Semester and SY: First 2024-2025

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

1. Objectives

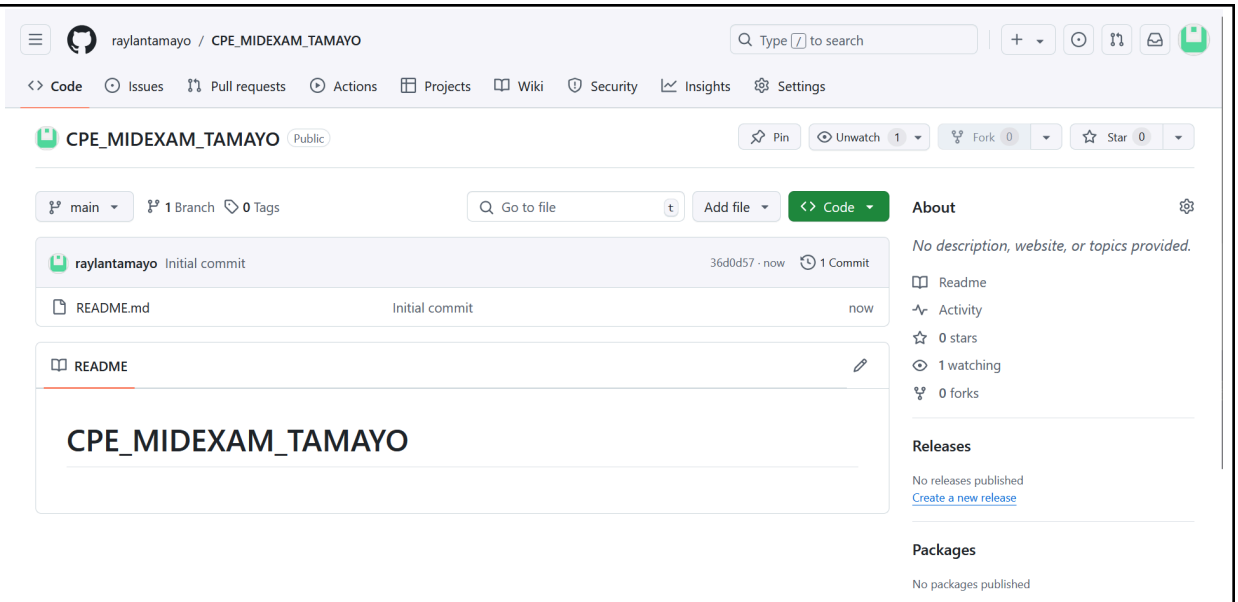
Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

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

1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.



2. Clone the repository and do the following:

```
tamayo@workstation:~$ git clone git@github.com:raylantamayo/CPE_MIDEXAM_TAMAYO.
git
Cloning into 'CPE_MIDEXAM_TAMAYO'...
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.
tamayo@workstation:~$
```

3. Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:

ansible.cfg	GNU nano 2.9.3 ansible.cfg [defaults] inventory = inventory host_key_checking = False deprecation_warnings = False remote_user = tamayo private_key_file = ~/.ssh/

	inventory	tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO File Edit View Search Terminal Help GNU nano 2.9.3 inventory [ubuntu_nagios] 192.168.56.103 [ubuntu_elk] 192.168.56.103 [centos_elk] 192.168.56.105 [igp_centos/ubuntu] 192.168.56.105 192.168.56.103 [ls_centos] 192.168.56.105 [ls_ubuntu] 192.168.56.103
	config.yaml	tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO File Edit View Search Terminal Help GNU nano 2.9.3 config.yaml --- - hosts: all become: true pre_tasks: - name: dnf and epel installation dnf: name: - epel-release - dnf when: ansible_distribution == "CentOS" - name: dpkg in ubuntu shell: | dpkg --configure -a when: ansible_distribution == "Ubuntu" - name: install updates (CentOS) dnf: update_cache: yes update_only: yes when: ansible_distribution == "CentOS"

```
    - name: install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
      when: ansible_distribution == "Ubuntu"


- hosts: centos_elk
  become: true
  roles:
    - centos_elk


- hosts: ubuntu_elk
  become: true
  roles:
    - ubuntu_elk


- hosts: ubuntu_nagios
  become: true
  roles:
    - ubuntu_nagios
```

```
- hosts: igp_centos/ubuntu
  become: true
  roles:
    - igp_centos/ubuntu


- hosts: ls_centos
  tags: ls_centos, ls_both
  become: true
  roles:
    - ls_centos


- hosts: ls_ubuntu
  tags: ls_ubuntu, ls_both
  become: true
  roles:
    - ls_ubuntu
```

4. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash) • Install Nagios in one host

Elastic Stack for CentOS	

tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO/roles/centos_elk/tasks

File Edit View Search Terminal Help

GNU nano 2.9.3 main.yml

```
- name: Install ALL Prerequisites
  dnf:
    name:
      - java-1.8.0-openjdk
      - epel-release
      - wget
      - which
    state: present
  become: yes

- name: Add Elasticsearch RPM Repository
  shell: rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Add Elasticsearch repository
  copy:
    content: |
      [elasticsearch-7.x]
      name=Elasticsearch repository for 7.x packages
      baseurl=https://artifacts.elastic.co/packages/7.x/yum
      gpgcheck=1
      gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
      enabled=1
      autorefresh=1
      type=rpm-md
    dest: /etc/yum.repos.d/elasticsearch.repo
  become: yes

- name: Install Elasticsearch for CentOS
  dnf:
    name: elasticsearch
    state: present
  become: yes

- name: Enable and Start Elasticsearch Service
  systemd:
    name: elasticsearch
    enabled: yes
    state: started
  become: yes

- name: Install Kibana for CentOS
  dnf:
    name: kibana
    state: present
  become: yes

- name: Enable and start Kibana Service
```

```
  systemd:
    name: kibana
    enabled: yes
    state: started
  become: yes

- name: Install Logstash for CentOS
  dnf:
    name: logstash
    state: present
  become: yes
```

```
- name: Enable and start Logstash service
  systemd:
    name: logstash
    enabled: yes
    state: started
  become: yes

- name: Restart Elasticsearch and Kibana
  systemd:
    name: "{{ item }}"
    state: restarted
  loop:
```

```
    - elasticsearch
    - kibana
```

Elastic Stack for Ubuntu

```
tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO/roles/ubuntu_elk/tasks
File  Edit  View  Search  Terminal  Help
 GNU nano 2.9.3                  main.yml                        Mo

- name: Install ALL prerequisites
  apt:
    name:
      - default-jre
      - apt-transport-https
      - curl
      - software-properties-common
    state: present
  become: yes

- name: Add Elasticsearch APT Repository Key
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
  become: yes

- name: Add Elasticsearch APT repository
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/7.x/apt stable mai
    state: present
  become: yes

- name: Install Elasticsearch fot Ubuntu
  apt:
```

```
    name: elasticsearch
    state: present
  become: yes

- name: Enable and start Elasticsearch service
  systemd:
    name: elasticsearch
    enabled: yes
    state: started
  become: yes

- name: Install Kibana for Ubuntu
```

```
    apt:
      name: kibana
      state: present
    become: yes

  - name: Enable and start Kibana Service
    systemd:
      name: kibana
      enabled: yes
      state: started
    become: yes
```

```
- name: Install Logstash for Ubuntu
  apt:
    name: logstash
    state: present
  become: yes

- name: Enable and start Logstash Service
  systemd:
    name: logstash
    enabled: yes
    state: started
  become: yes
```

```
- name: Restart Elasticsearch and Kibana
  systemd:
    name: "{{ item }}"
    state: restarted
  loop:
    - elasticsearch
    - kibana
```

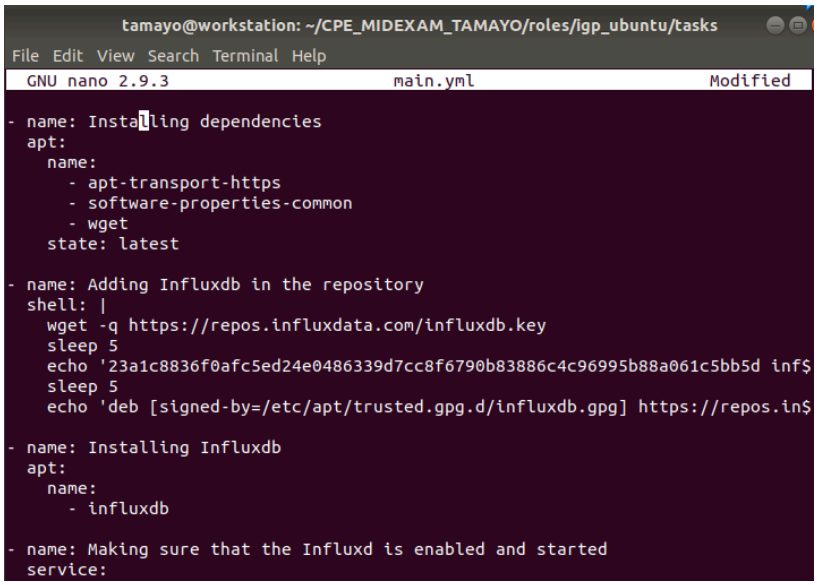
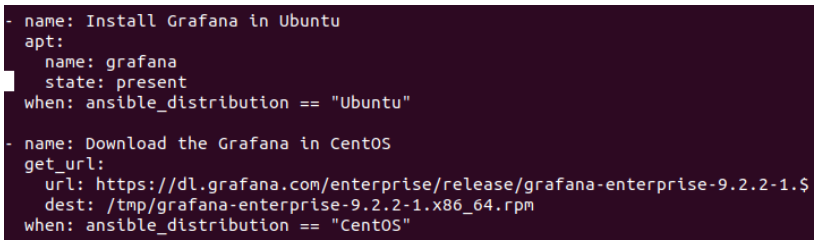
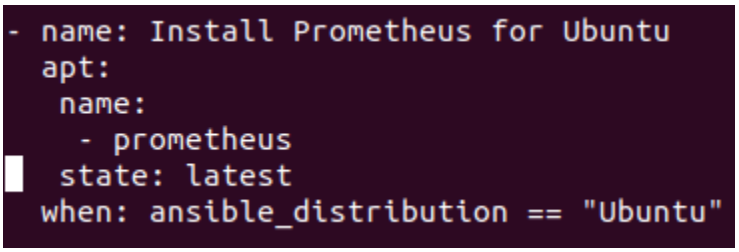
	Nagios for Ubuntu	tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO/roles/ubuntu_nagios/tasks File Edit View Search Terminal Help GNU nano 2.9.3 main.yml Mod --- - name: nagios libraries and dependencies (Ubuntu) tags: ubuntu, dependencies, libraries apt: name: - autoconf - libc6 - gcc - make - wget - unzip - apache2 - php - libapache2-mod-php - libgd-dev - openssl - libssl-dev - bc - gawk - dc - build-essential - snmp - libnet-snmp-perl - gettext - python3 - python3-pip state: latest - name: passlib package pip: name: passlib - name: nagios directory PATH file: path: ~/nagios state: directory - name: downloading nagios unarchive: src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.$ dest: ~/nagios remote_src: yes mode: 0777 owner: root group: root - name: downloading nagios plugins unarchive: src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.$ dest: ~/nagios remote_src: yes mode: 0777 owner: root group: root - name: install, compile, adding users and groups 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

- name: compile and install plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install

- name: adding users to nagios
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin
```

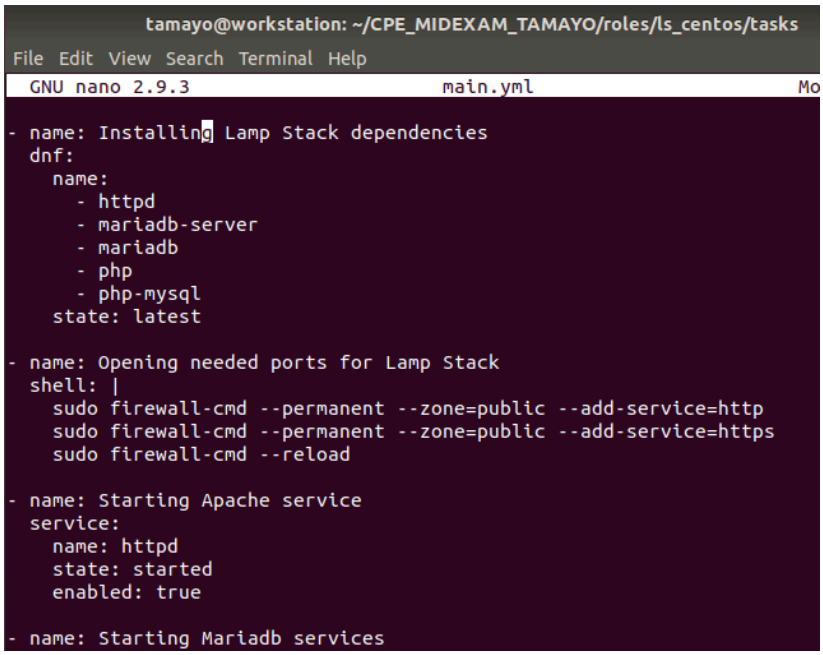
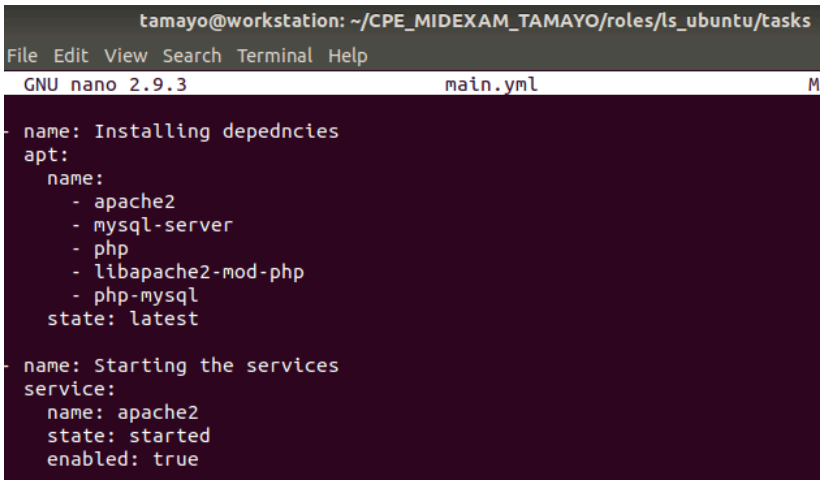
```
- name: Nagios Start/Enable Check
  service:
    name: nagios
    state: restarted
    enabled: true

- name: Apache/httpd Start/Enable check
  service:
    name: apache2
    state: restarted
    enabled: true
```

5. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)

Influxdb for CentOS	tamayo@workstation: ~/CPE_MIDEXAM_TAMAYO/roles/igp_centos/tasks File Edit View Search Terminal Help GNU nano 2.9.3 main.yml - name: Copying the Influxdb repository file unarchive: src: https://dl.influxdata.com/influxdb/releases/influxdb2-2.4.0-li dest: /tmp/ remote_src: yes mode: 0777 owner: root group: root - name: Adding the executables to the PATH shell: cd /tmp/influxdb2* sudo cp influxdb2-2.4.0-linux-amd64/influxd /usr/local/bin/
Grafana for CentOS	- name: Downloading Grafana package get_url: url: https://dl.grafana.com/enterprise/release/grafana-enterprise-9.2.2-1.$ dest: /tmp/grafana-enterprise-9.2.2-1.x86_64.rpm - name: Installing Grafana dnf: name: /tmp/grafana-enterprise-9.2.2-1.x86_64.rpm state: present - name: Enabling Grafana service service: name: grafana-server enabled: yes
Prometheus for CentOS	- name: Install Prometheus for CentOS dnf: name: - epel-release - snapd state: latest when: ansible_distribution == "CentOS" - name: Enabling snapd command: systemctl enable --now snapd.socket when: ansible_distribution == "CentOS" - name: Prometheus for CentOS command: snap install prometheus --classic when: ansible_distribution == "CentOS"

Influxdb for Ubuntu	
Grafana for Ubuntu	
Prometheus for Ubuntu	

6. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)

Lamp Stack for CentOS	
Lamp Stack for Ubuntu	

7. Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.

8. Document the push and commit from the local repository to GitHub.

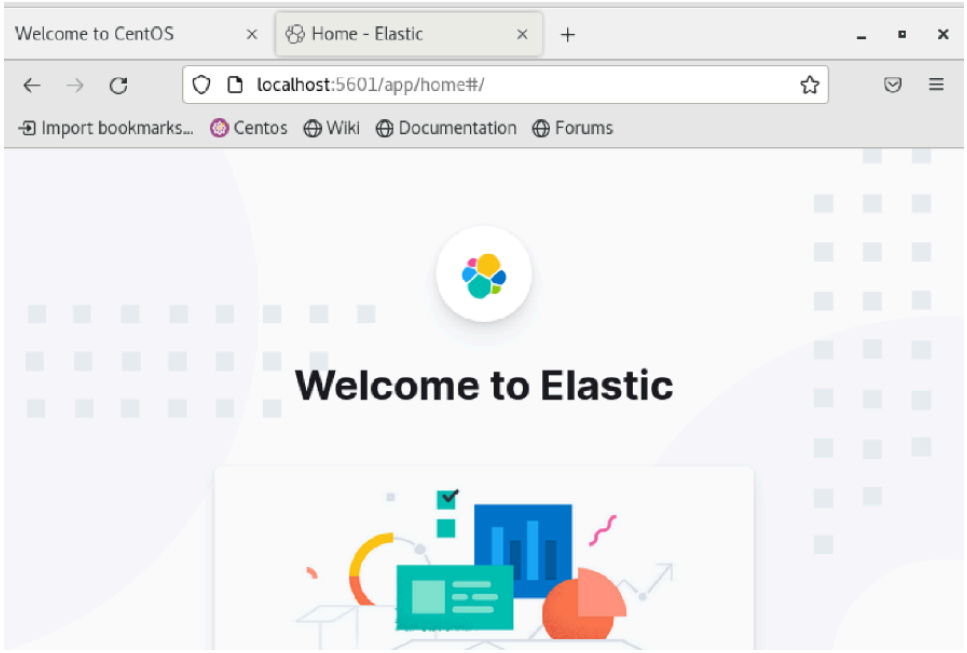
```
tamayo@workstation:~/CPE_MIDEXAM_TAMAYO$ git add .
tamayo@workstation:~/CPE_MIDEXAM_TAMAYO$ git commit -m "MIDTERMS EXAM"
[main 655edcb] MIDTERMS EXAM
 13 files changed, 641 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 config.yaml
 create mode 100644 files/grafana.repo
 create mode 100644 inventory
 create mode 100644 prometheus.service
 create mode 100644 roles/centos_elk/tasks/main.yml
 create mode 100644 roles/igp_centos/tasks/main.yml
 create mode 100644 roles/igp_centos/ubuntu/tasks/main.yml
 create mode 100644 roles/igp_ubuntu/tasks/main.yml
 create mode 100644 roles/ls_centos/tasks/main.yml
 create mode 100644 roles/ls_ubuntu/tasks/main.yml
 create mode 100644 roles/ubuntu_elk/tasks/main.yml
 create mode 100644 roles/ubuntu_nagios/tasks/main.yml
tamayo@workstation:~/CPE_MIDEXAM_TAMAYO$ git push origin main
Counting objects: 33, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (17/17), done.
Writing objects: 100% (33/33), 6.20 KiB | 6.20 MiB/s, done.
Total 33 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:raylantamayo/CPE_MIDEXAM_TAMAYO.git
   36d0d57..655edcb  main -> main
tamayo@workstation:~/CPE_MIDEXAM_TAMAYO$
```

9. Finally, paste also the link of your GitHub repository in the documentation.

OUTPUT

```
PLAY [centos_elk] ***********************************************************
*

TASK [Gathering Facts] ******************************************************
*
ok: [192.168.56.105]

TASK [centos_elk : Install ALL Prerequisites] *******************************
*
ok: [192.168.56.105]

TASK [centos_elk : Add Elasticsearch RPM Repository] ************************
*
changed: [192.168.56.105]

TASK [centos_elk : Add Elasticsearch repository] ****************************
*
ok: [192.168.56.105]

TASK [centos_elk : Install Elasticsearch for CentOS] ************************
*
ok: [192.168.56.105]

TASK [centos_elk : Enable and Start Elasticsearch Service] ******************
*
ok: [192.168.56.105]

TASK [centos_elk : Install Kibana for CentOS] *******************************
```

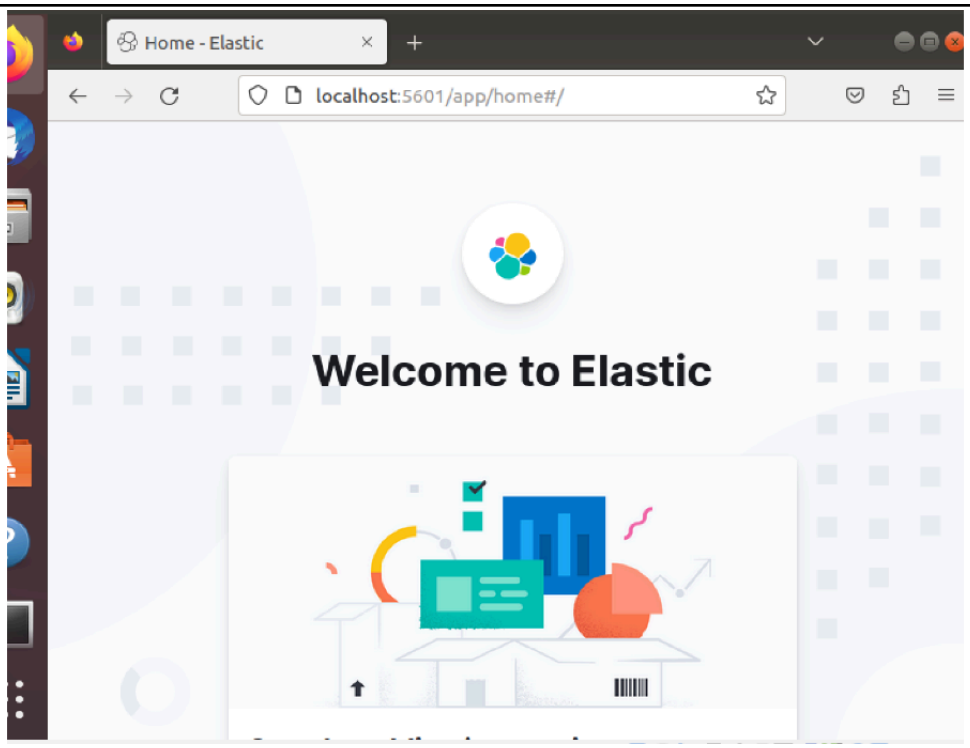
```
TASK [centos_elk : Enable and start Kibana Service] *************************
ok: [192.168.56.105]

TASK [centos_elk : Install Logstash for CentOS] *****************************
ok: [192.168.56.105]

TASK [centos_elk : Enable and start Logstash service] ************************
ok: [192.168.56.105]

TASK [centos_elk : Restart Elasticsearch and Kibana] *************************
changed: [192.168.56.105] => (item=elasticsearch)
changed: [192.168.56.105] => (item=kibana)
```




```
PLAY [ubuntu_elk] ***************************************************************

TASK [Gathering Facts] **********************************************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install ALL prerequisites] ***********************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Add Elasticsearch APT Repository Key] ************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Add Elasticsearch APT repository] ****************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Elasticsearch fot Ubuntu] ****************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Enable and start Elasticsearch service] **********************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Kibana for Ubuntu] ***********************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Enable and start Kibana Service] *****************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Logstash for Ubuntu] *********************************
ok: [192.168.56.103]
```

Help

```
TASK [ubuntu_elk : Enable and start Logstash Service] ***************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Restart Elasticsearch and Kibana] ****************************
changed: [192.168.56.103] => (item=elasticsearch)
changed: [192.168.56.103] => (item=kibana)
```




```
PLAY [ubuntu_nagios] ***********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : nagios libraries and dependencies (Ubuntu)] **************
ok: [192.168.56.103]

TASK [ubuntu_nagios : passlib package] *****************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : nagios directory PATH] ***********************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : downloading nagios] **************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : downloading nagios plugins] ******************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : install, compile, adding users and groups] ***************
changed: [192.168.56.103]

TASK [ubuntu_nagios : compile and install plugins] *****************************
changed: [192.168.56.103]

TASK [ubuntu_nagios : adding users to nagios] **********************************
ok: [192.168.56.103]
```

GitHub link:

https://github.com/raylantamayo/CPE_MIDEXAM_TAMAYO.git

Conclusions:

I installed Nagios and ElasticStack using all of my playbook when I completed my midterm exam. To sum up, taking the midterm exam was an enlightening experience that opened my eyes to the world of system monitoring and administration. I learned the value of centralized logging and real-time data analysis while installing Elastic Stack on both Ubuntu and CentOS. My knowledge of network monitoring and alerting improved once I set up Nagios entirely on CentOS. Grafana, InfluxDB, and Prometheus were deployed on both operating systems to show off the capabilities of performance monitoring and data visualization. Finally, setting up LAMP stacks on Ubuntu and CentOS demonstrated how flexible web servers are. My perspectives have expanded as a result of this encounter, and I now possess priceless abilities for overseeing and maximizing various server environments.