Name: Pastrana, Mark Laurenz S.	Date Performed: November 14, 2023
Course/Section: CPE31S5	Date Submitted: November 14, 2023
Instructor: Engr. Richard Roman	Semester and SY: 2023-2024

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

1. Objectives

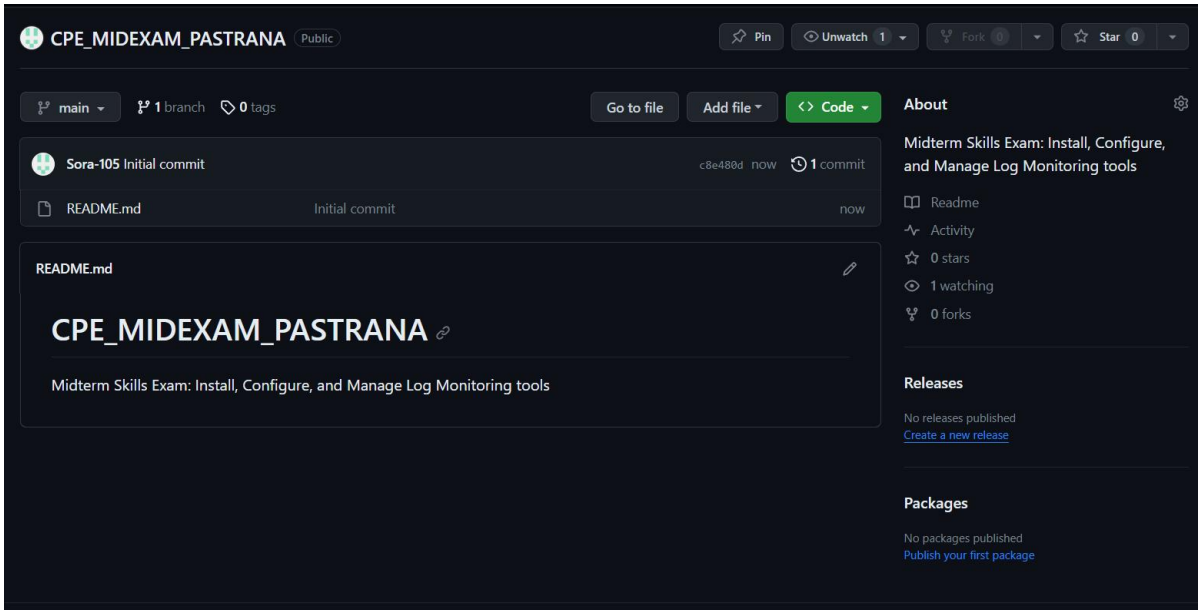
Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

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



Cloning the repository:

```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA
pastrana@localmachina:~$ git clone https://github.com/Sora-105/CPE_MIDEXAM_PASTRANA.git
Cloning into 'CPE_MIDEXAM_PASTRANA'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
pastrana@localmachina:~$ cd CPE_MIDEXAM_PASTRANA
pastrana@localmachina:~/CPE_MIDEXAM_PASTRANA$ ls
README.md
pastrana@localmachina:~/CPE_MIDEXAM_PASTRANA$
```

Config.yaml

```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA
 GNU nano 6.2                                    config.yaml *
---

- 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: Installing installations dependencies
    apt:
      name:
        - wget
      state: latest
    when: ansible_distribution == "Ubuntu"

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

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA

GNU nano 6.2 config.yaml *

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

- hosts: elk_centos
  tags: elk_centos, elk_both
  become: true
  roles:
    - elk_centos

- hosts: elk_ubuntu
  tags: elk_ubuntu, elk_both
  become: true
  roles:
    - elk_ubuntu

- hosts: nagios_centos
  tags: nagios_centos
  become: true
  roles:
    - nagios_centos

- hosts: igp_centos
  tags: igp_centos, igp_both
  become: true
  roles:
    - igp_centos

- hosts: igp_ubuntu
  tags: igp_ubuntu, igp_both
```

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA

GNU nano 6.2 config.yaml *

```
    - elk_ubuntu

- hosts: nagios_centos
  tags: nagios_centos
  become: true
  roles:
    - nagios_centos

- hosts: igp_centos
  tags: igp_centos, igp_both
  become: true
  roles:
    - igp_centos

- hosts: igp_ubuntu
  tags: igp_ubuntu, igp_both
  become: true
  roles:
    - igp_ubuntu

- hosts: lms_centos
  tags: lms_centos, lms_both
  become: true
  roles:
    - lms_centos

- hosts: lms_ubuntu
  tags: lms_ubuntu, lms_both
  become: true
  roles:
    - lms_ubuntu
```

Ansible.cfg

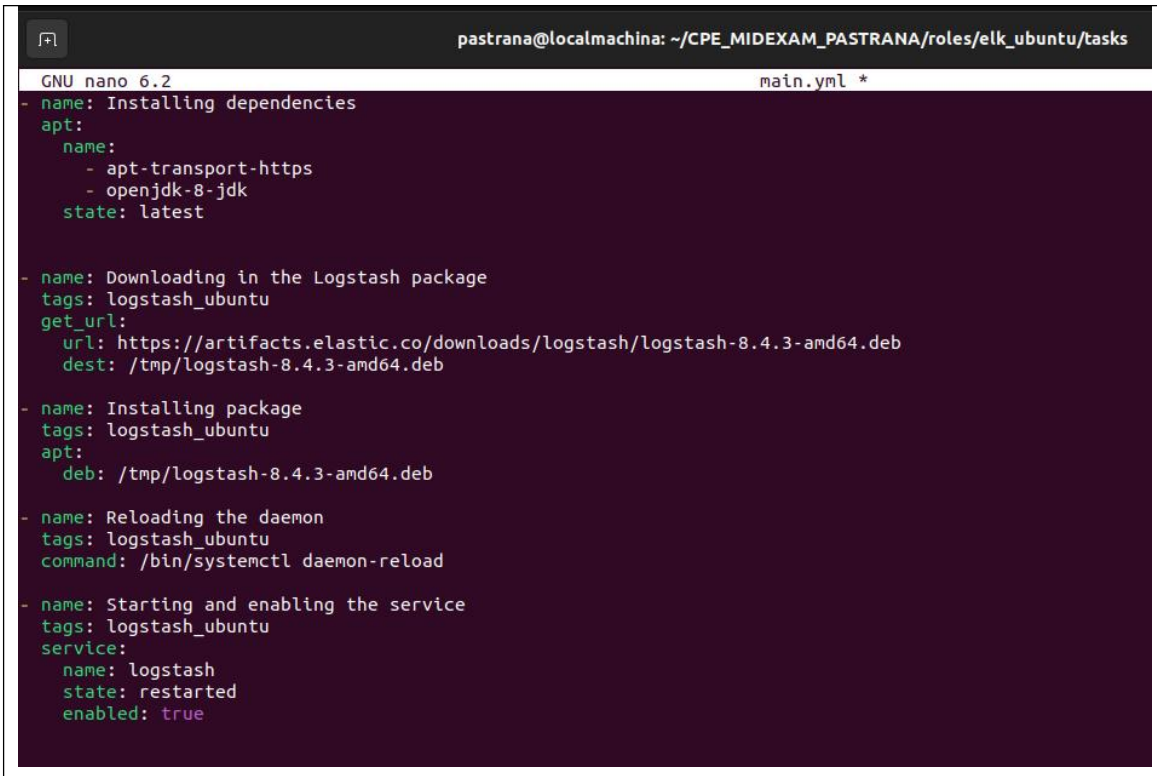
```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA
  GNU nano 6.2                                        ansible.cfg
[defaults]

inventory = inventory
host_key_checking = False

deprecation_warnings = False

remote_user = pastrana
private_key_file = ~/.ssh/
```

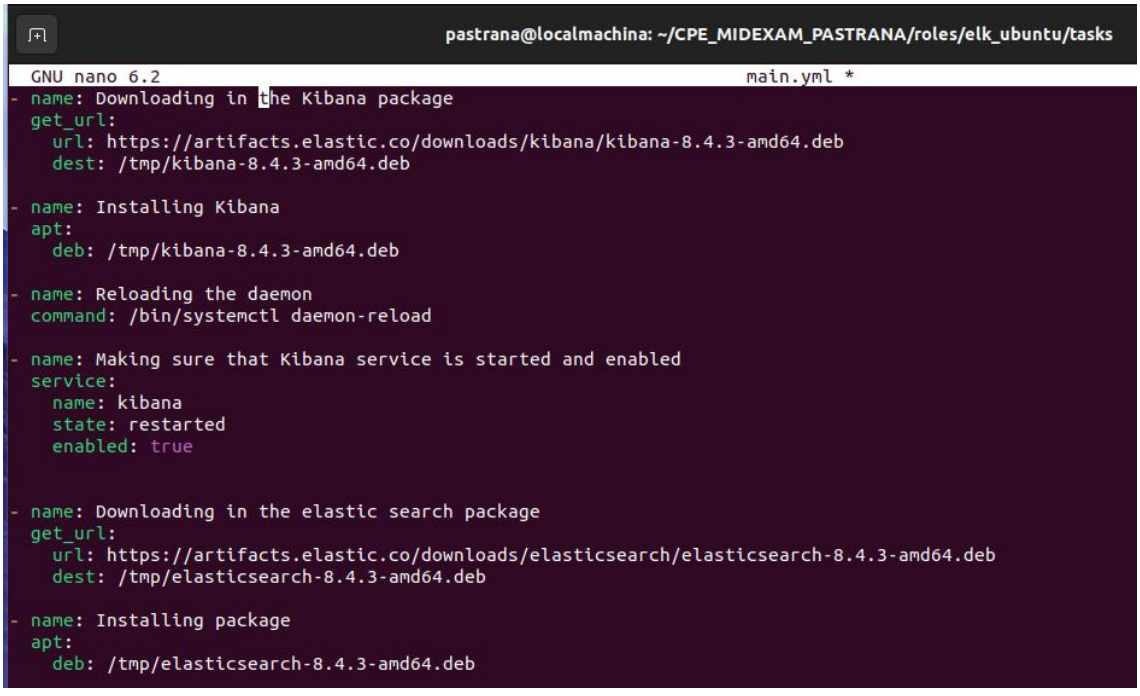
Inventory:

```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA
  GNU nano 6.2                                        inventory
[elk_ubuntu]
192.168.56.116

[igp_ubuntu]
192.168.56.116

[lms_ubuntu]
192.168.56.116

[elk_centos]
192.168.56.109

[igp_centos]
192.168.56.109

[lms_centos]
192.168.56.109

[nagios_centos]
192.168.56.109
```

Roles:

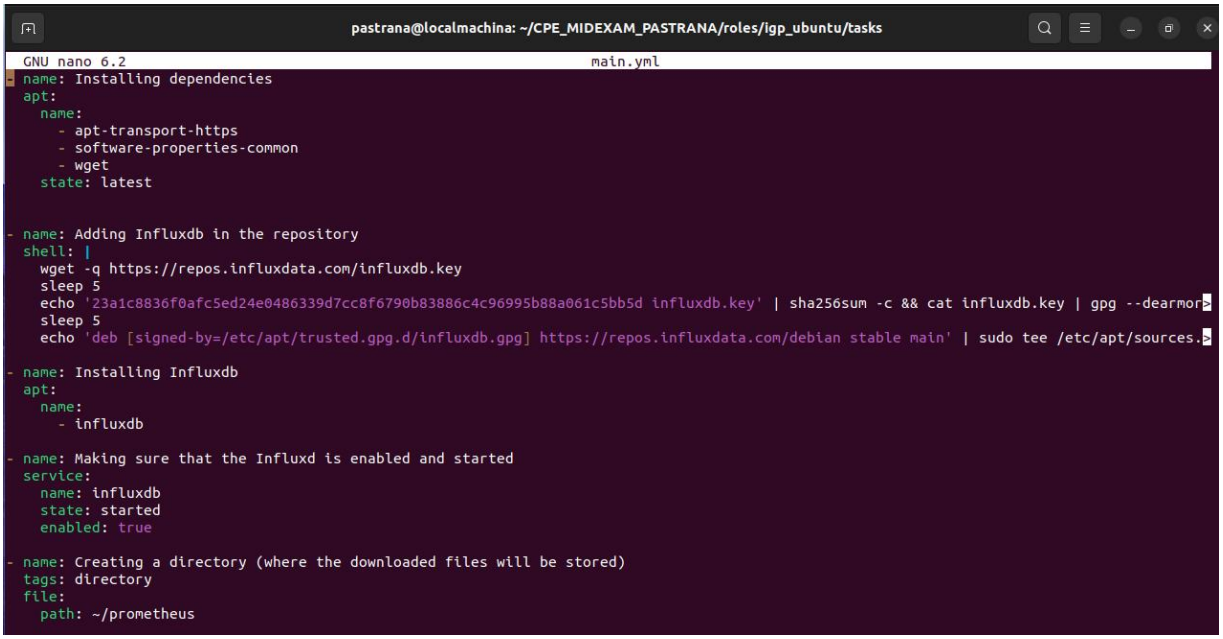
[elk_ubuntu]


pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/elk_ubuntu/tasks
GNU nano 6.2 main.yml *
- name: Installing dependencies
apt:
name:
- apt-transport-https
- openjdk-8-jdk
state: latest
- name: Downloading in the Logstash package
tags: logstash_ubuntu
get_url:
url: https://artifacts.elastic.co/downloads/logstash/logstash-8.4.3-amd64.deb
dest: /tmp/logstash-8.4.3-amd64.deb
- name: Installing package
tags: logstash_ubuntu
apt:
deb: /tmp/logstash-8.4.3-amd64.deb
- name: Reloading the daemon
tags: logstash_ubuntu
command: /bin/systemctl daemon-reload
- name: Starting and enabling the service
tags: logstash_ubuntu
service:
name: logstash
state: restarted
enabled: true


pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/elk_ubuntu/tasks
GNU nano 6.2 main.yml *
- name: Downloading in the Kibana package
get_url:
url: https://artifacts.elastic.co/downloads/kibana/kibana-8.4.3-amd64.deb
dest: /tmp/kibana-8.4.3-amd64.deb
- name: Installing Kibana
apt:
deb: /tmp/kibana-8.4.3-amd64.deb
- name: Reloading the daemon
command: /bin/systemctl daemon-reload
- name: Making sure that Kibana service is started and enabled
service:
name: kibana
state: restarted
enabled: true
- name: Downloading in the elastic search package
get_url:
url: https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.4.3-amd64.deb
dest: /tmp/elasticsearch-8.4.3-amd64.deb
- name: Installing package
apt:
deb: /tmp/elasticsearch-8.4.3-amd64.deb

```
- name: Modifying service file
  tags: es_ubuntu
  replace:
    path: /usr/lib/systemd/system/elasticsearch.service
    regexp: "TimeoutStartSec=75"
    replace: "TimeoutStartSec=500"

- name: Starting and enabling the deamon
  shell: |
    sudo systemctl enable elasticsearch.service
    sleep 10
    sudo systemctl start elasticsearch.service
  ignore_errors: yes
```

[igp_ubuntu]



pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/igp_ubuntu/tasks

GNU nano 6.2 main.yml *

```
    state: directory

- name: Downloading and extracting Prometheus
  tags: source
  unarchive:
    src: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-amd64.tar.gz
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Stopping the Prometheus service if its exist
  shell: |
    sudo systemctl stop prometheus >> /dev/null
  ignore_errors: yes

- 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 service is started and enabled
  tags: serviceon
  service:
    name: prometheus
    state: started
    enabled: true
```

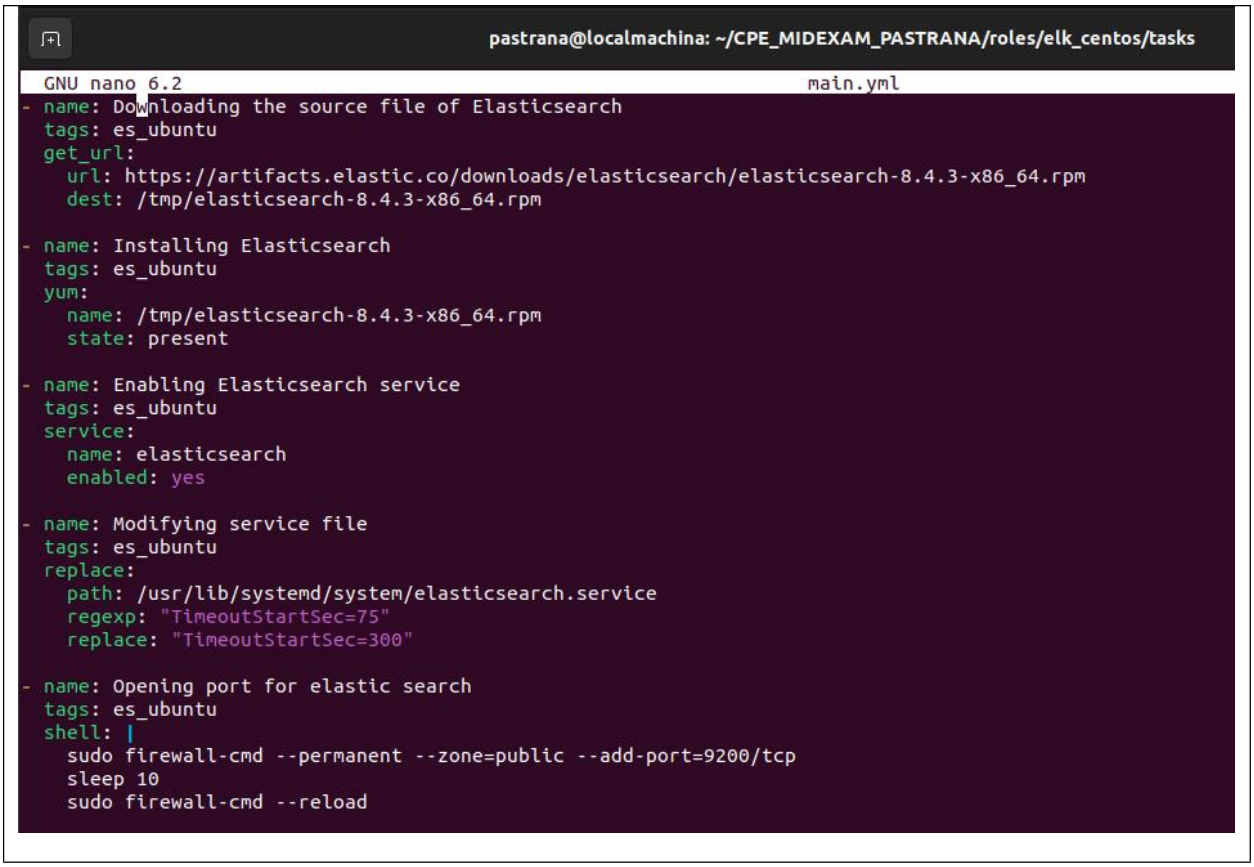
[lms_ubuntu]

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/lms_ubuntu/tasks

GNU nano 6.2 main.yml

```
- name: Installing depedncies
  apt:
    name:
      - apache2
      - mysql-server
      - php
      - libapache2-mod-php
      - php-mysql
    state: latest

- name: Starting the services
  service:
    name: apache2
    state: started
    enabled: true
```

[elk_centos]


pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/elk_centos/tasks
GNU nano 6.2 main.yml
- name: Downloading the source file of Elasticsearch
tags: es_ubuntu
get_url:
url: https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.4.3-x86_64.rpm
dest: /tmp/elasticsearch-8.4.3-x86_64.rpm
- name: Installing Elasticsearch
tags: es_ubuntu
yum:
name: /tmp/elasticsearch-8.4.3-x86_64.rpm
state: present
- name: Enabling Elasticsearch service
tags: es_ubuntu
service:
name: elasticsearch
enabled: yes
- name: Modifying service file
tags: es_ubuntu
replace:
path: /usr/lib/systemd/system/elasticsearch.service
regexp: "TimeoutStartSec=75"
replace: "TimeoutStartSec=300"
- name: Opening port for elastic search
tags: es_ubuntu
shell: |
sudo firewall-cmd --permanent --zone=public --add-port=9200/tcp
sleep 10
sudo firewall-cmd --reload

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/elk_centos/tasks

GNU nano 6.2 main.yml

```
- name: Enabling elastic search service
  tags: es_ubuntu
  shell: |
    systemctl enable elasticsearch.service
    sleep 10
    systemctl start elasticsearch.service
  ignore_errors: yes


- name: Downloading and installing public signing key
  tags: logstash_ubuntu
  rpm_key:
    state: present
    key: https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Creating a repo file for Logstash
  tags: logstash_ubuntu
  copy:
    src: logstash.repo
    dest: /etc/yum.repos.d/logstash.repo
    owner: root
    group: root
    mode: 0777

- name: Updating repo
  tags: logstash_ubuntu
  dnf:
    update_cache: yes

- name: Installing Logstash and its dependencies
  tags: logstash_ubuntu
  dnf:
```

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/elk_centos/tasks

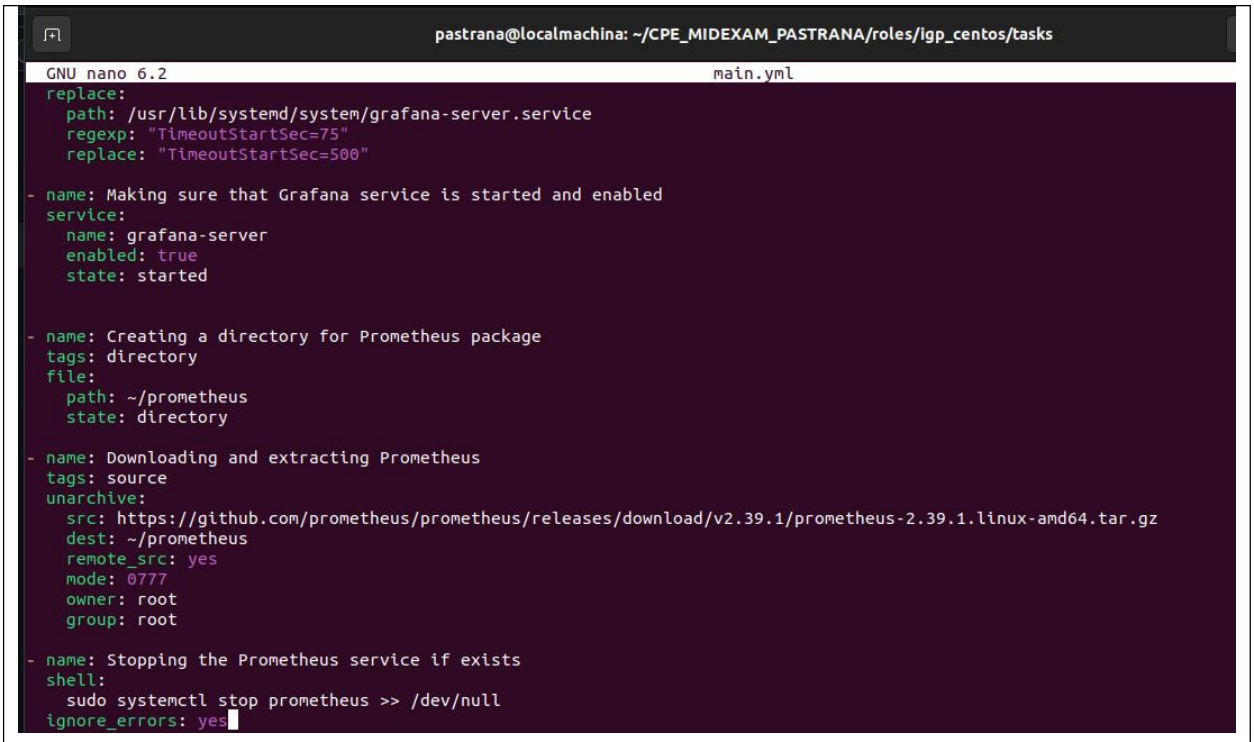
GNU nano 6.2 main.yml *

```
    name:
      - logstash
    state: latest

- name: Opening port for Logstash
  tags: logstash_ubuntul, elk_install
  shell: |
    sudo firewall-cmd --permanent --zone=public --add-port=9600/tcp
    sleep 10
    sudo firewall-cmd --reload

- name: Making sure that logstash is stared and enabled
  tags: logstash_ubuntu, service, logstash_service, elk_service
  service:
    name: logstash
    state: restarted
    enabled: true


- name: Downloading and installing public signing key
  tags: kibana_ubuntu, kibana_install, elk_install
  rpm_key:
    state: present
    key: https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Adding Kibana to the RPM repository
  tags: kibana_ubuntu, kibana_install, elk_install
  copy:
    src: kibana.repo
    dest: /etc/yum.repos.d/kibana.repo
    owner: root
    group: root
```

```
    mode: 777

- name: Updating the repository once again
  tags: kibana_ubuntu, kibana_install, elk_install
  yum:
    name:
      - kibana
    state: latest

- name: Opening port for Kibana
  tags: kibana_ubuntu, kibana_installl, elk_install
  firewalld:
    port: 5601/tcp
    zone: public
    permanent: yes
    state: enabled

- name: Making sure that Kibana is started and enabled
  tags: kibana_ubuntu, elk_service, kibana_service, service
  service:
    name: kibana
    state: restarted
    enabled: true
```

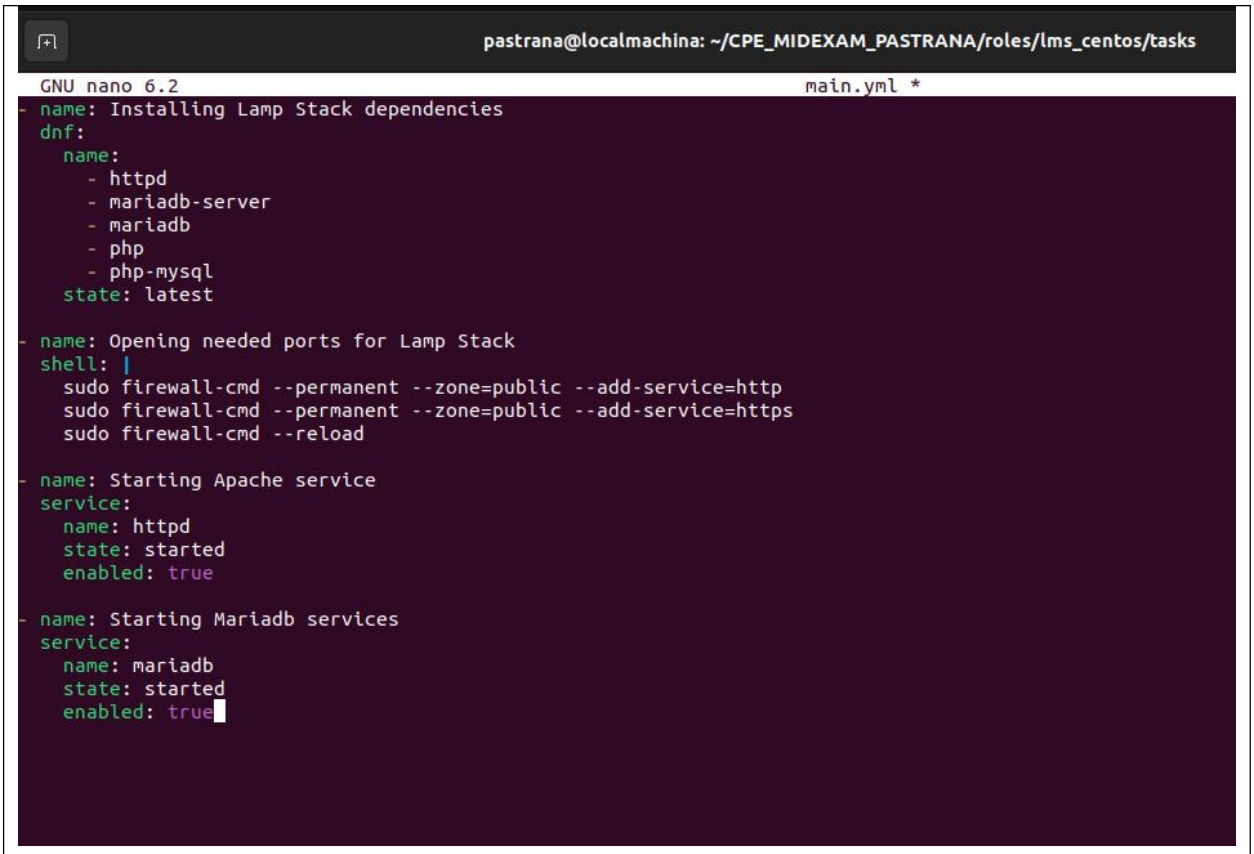
[igp_centos]

```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/igp_centos/tasks
 GNU nano 6.2                                                main.yml
- name: Copying the Influxdb repository file
  unarchive:
    src: https://dl.influxdata.com/influxdb/releases/influxdb2-2.4.0-linux-amd64.tar.gz
    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/


- name: Downloading Grafana package
  get_url:
    url: https://dl.grafana.com/enterprise/release/grafana-enterprise-9.2.2-1.x86_64.rpm
    dest: /tmp/grafana-enterprise-9.2.2-1.x86_64.rpm

- name: Installing Grafana
  yum:
    name: /tmp/grafana-enterprise-9.2.2-1.x86_64.rpm

- name: Enabling Grafana service
  service:
    name: grafana-server
    enabled: yes

- name: Modifying service file
  tags: es_ubuntu
  replace:
```




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


- 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 service is started and enabled
  service:
    name: prometheus
    state: restarted
    enabled: true
```

[lms_centos]



[nagios_centos]

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/nagios_centos/tasks

GNU nano 6.2 main.yml

```
- name: Installing nagios dependecies and libraries
  tags: dependecies, libraries
  yum:
    name:
      - gcc
      - glibc
      - 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
      - python2-pip
    state: latest

- name: Install passlib python package
  pip:
    name: passlib
```

pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles/nagios_centos/tasks

GNU nano 6.2 main.yml

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

- name: Downloading and extracting Nagios
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Compiling, installing, and adding users and groups in nagios
  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
- name: Downloading and extracting Nagios plugins
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.3.tar.gz
    dest: ~/nagios
    remote_src: yes
    mode: 0777
```

```
    owner: root
    group: root

- name: Compiling and installing plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install
- name: Add a user to a password file and ensure permissions are set
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin123

- name: Making sure that nagios is started and enabled
  service:
    name: nagios
    state: restarted
    enabled: true

- name: Making sure that httpd is started and enabled
  service:
    name: httpd
    state: restarted
    enabled: true
```

```
pastrana@localmachina: ~/CPE_MIDEXAM_PASTRANA/roles
pastrana@localmachina:~/CPE_MIDEXAM_PASTRANA/roles$ tree
.
├── elk_centos
│   └── tasks
│       └── main.yml
├── elk_ubuntu
│   └── tasks
│       └── main.yml
├── igp_centos
│   └── tasks
│       └── main.yml
├── igp_ubuntu
│   └── tasks
│       └── main.yml
├── lms_centos
│   └── tasks
│       └── main.yml
├── lms_ubuntu
│   └── tasks
│       └── main.yml
└── nagios_centos
    └── tasks
        └── main.yml

14 directories, 7 files
pastrana@localmachina:~/CPE_MIDEXAM_PASTRANA/roles$
```

GitHub link: *https://github.com/Sora-105/CPE_MIDEXAM_PASTRANA.git*

Conclusions: In conclusion, for this midterm skills exam, Ansible is an effective tool for putting infrastructure as code (IaC) operations into practice. You may save time and minimize errors by automating the configuration,installation, and management of your monitoring tools with Ansible. When doing this activity, I encounter a lot of errors and difficulties because I'm not fully familiar with the tools and syntax. You need to be very precise, using indentions and spaces, in order for the code to run properly without any errors. It was also challenging due to my PC status; even though I have 16 GB of RAM, it was lagging a bit and sometimes it hangs. Furthermore, you can quickly establish your

infrastructure, automate operations with playbooks, and incorporate monitoring tools into your IaC workflow with Ansible. I wasn’t able to run the final code due to PC issues but it was installed.