Name:Bernardo, Christian Emmanuel M.	**Date Performed: 01/04/2024**
Course/Section: CPE 232-CPE31S1	**Date Submitted: 01/04/2024**
Instructor: Dr. Jonathan Taylar	**Semester and SY: 2nd, 2023-2024**

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

1. Objectives

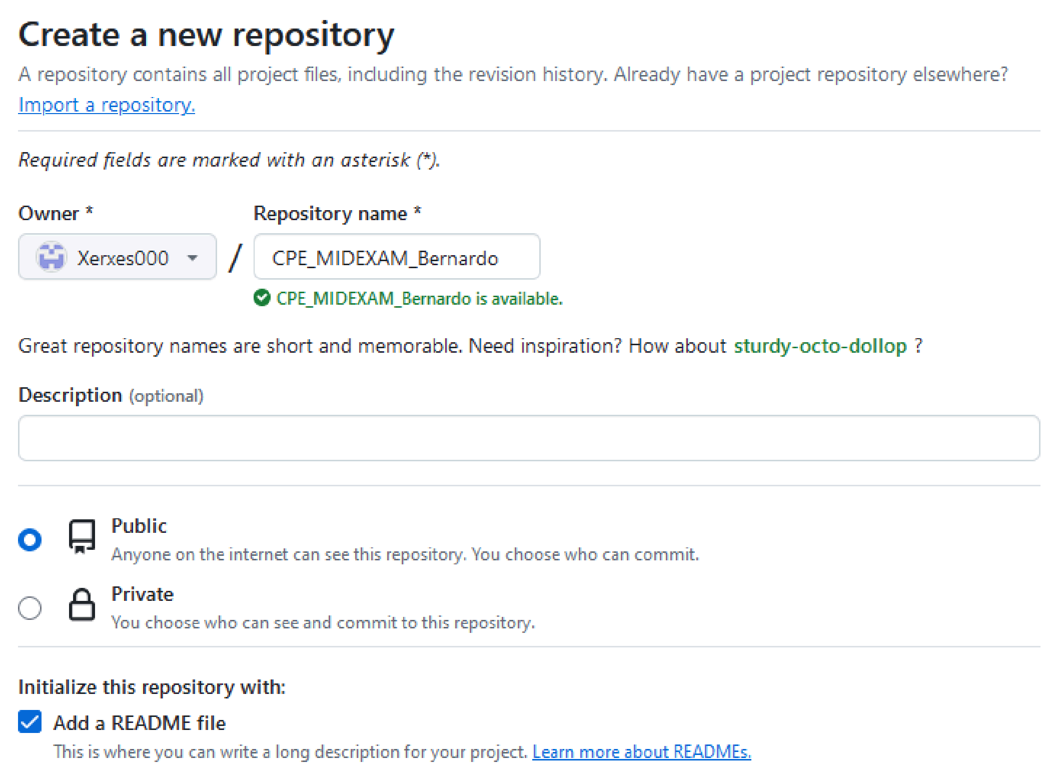
Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Instructions

1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.
2. Clone the repository and do the following:
   Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:
   Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash) • Install Nagios in one host
   Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)
   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)

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



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner * Xerxes000 / Repository name * CPE_MIDEXAM_Bernardo

✓ CPE_MIDEXAM_Bernardo is available.

Great repository names are short and memorable. Need inspiration? How about sturdy-octo-dollop ?

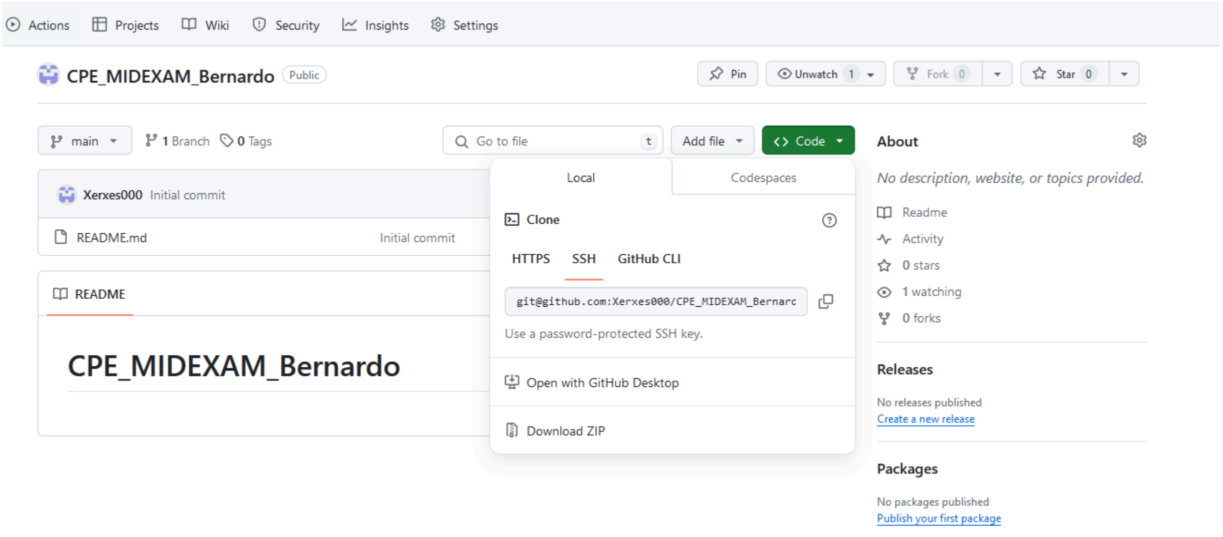
Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.



CPE_MIDEXAM_Bernardo Public

Xerxes000 Initial commit

README.md Initial commit

README

CPE_MIDEXAM_Bernardo

Clone

HTTPS SSH GitHub CLI

```
git@github.com:Xerxes000/CPE_MIDEXAM_Bernarc
```

Use a password-protected SSH key.

Open with GitHub Desktop

Download ZIP

About

No description, website, or topics provided.

Readme
Activity
0 stars
1 watching
0 forks

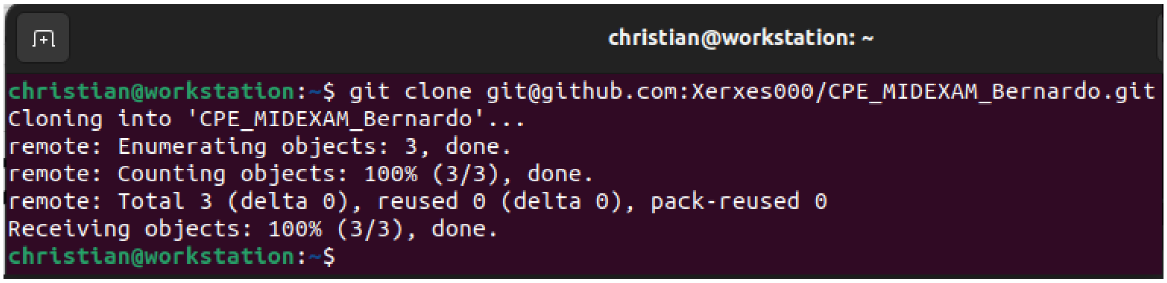
Releases

No releases published
Create a new release

Packages

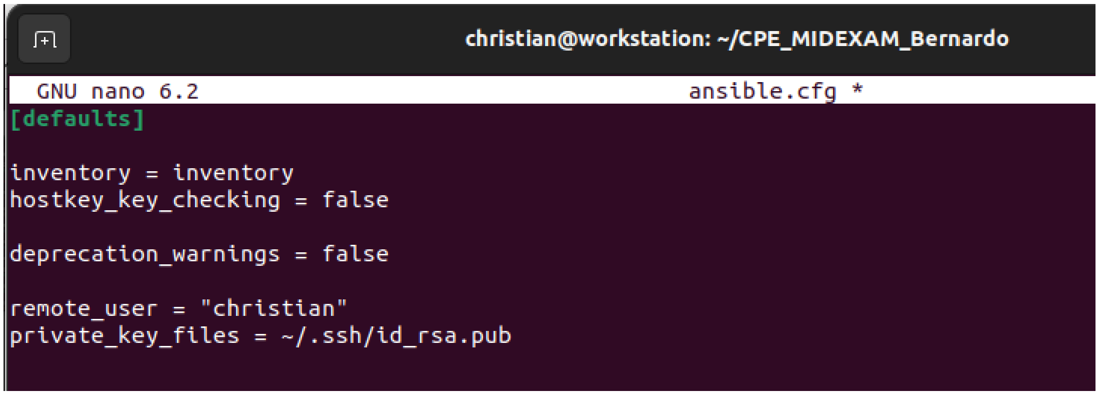
No packages published
Publish your first package

Clone the repository and do the following

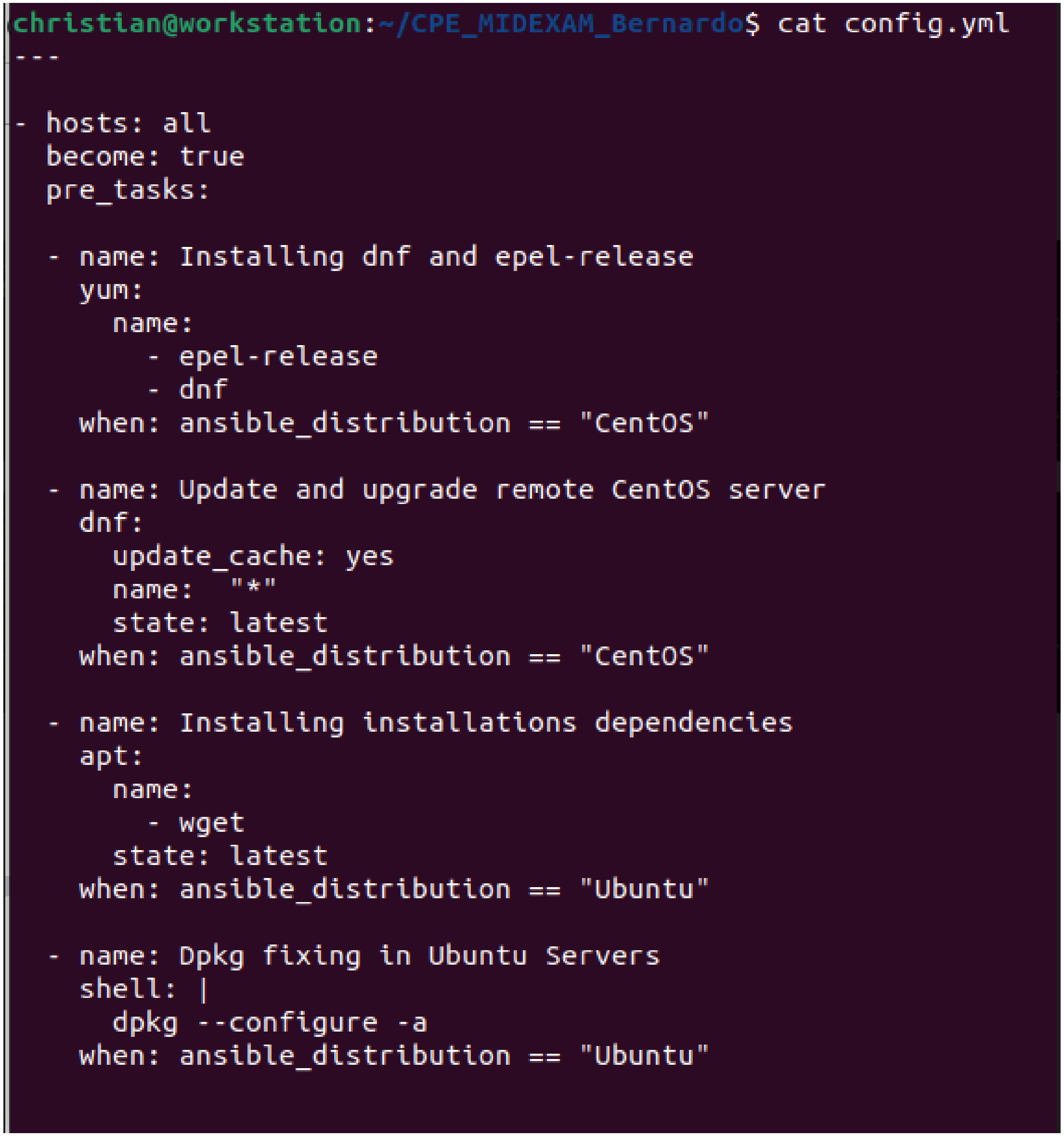


```
christian@workstation:~$ git clone git@github.com:Xerxes000/CPE_MIDEXAM_Bernardo.git
Cloning into 'CPE_MIDEXAM_Bernardo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
christian@workstation:~$
```

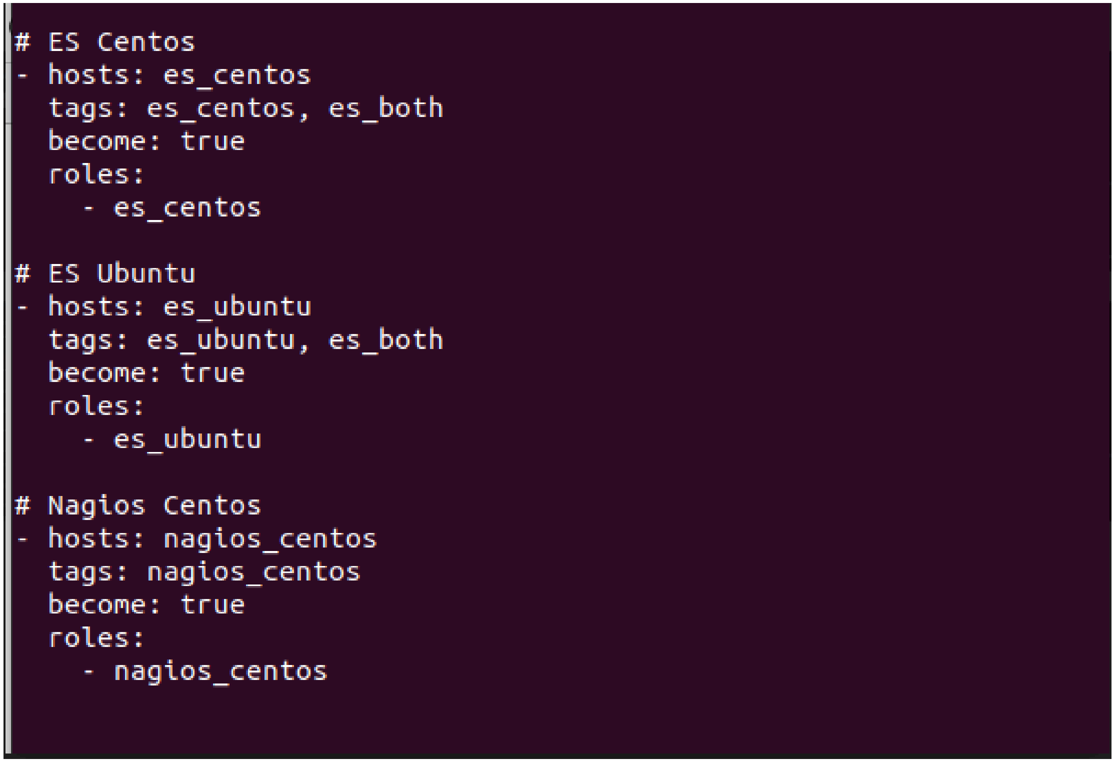
Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:



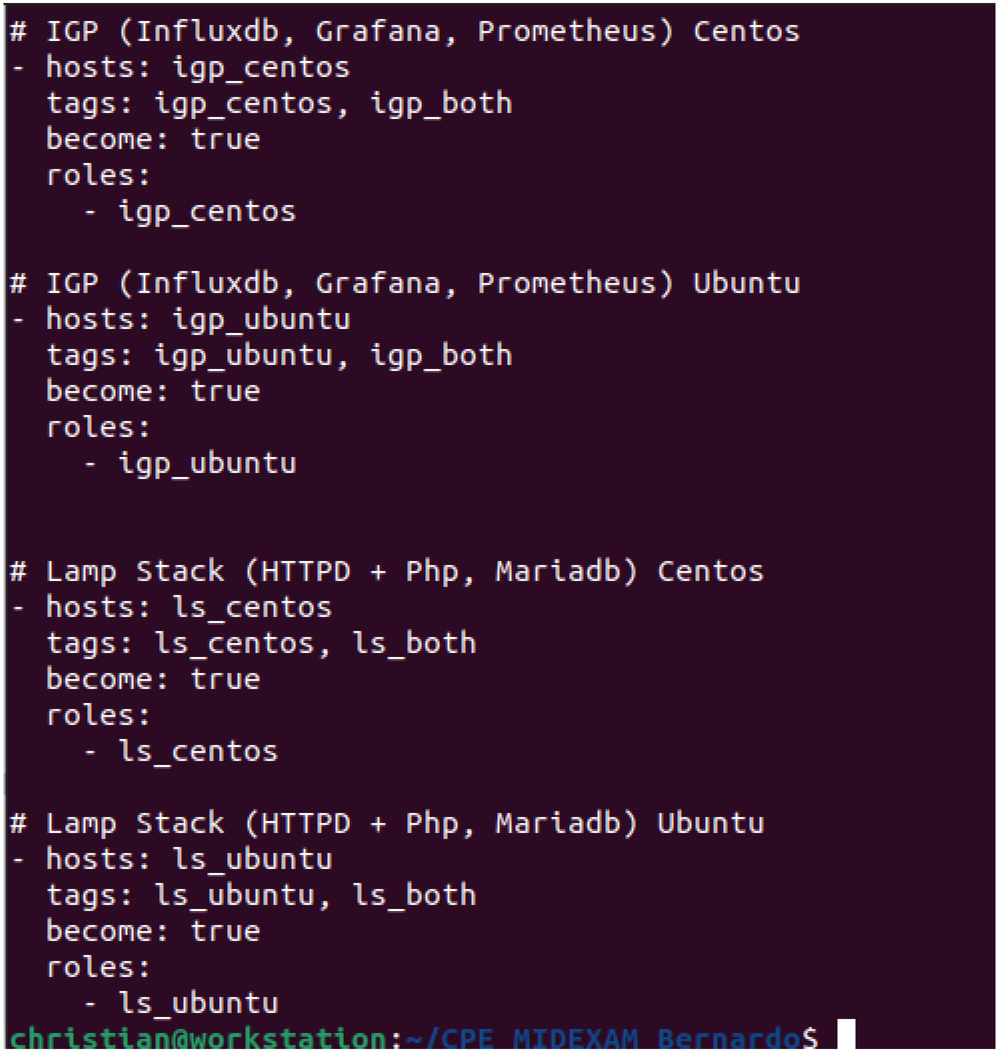
```
  GNU nano 6.2                                          ansible.cfg *
[defaults]

inventory = inventory
hostkey_key_checking = false

deprecation_warnings = false

remote_user = "christian"
private_key_files = ~/.ssh/id_rsa.pub
```



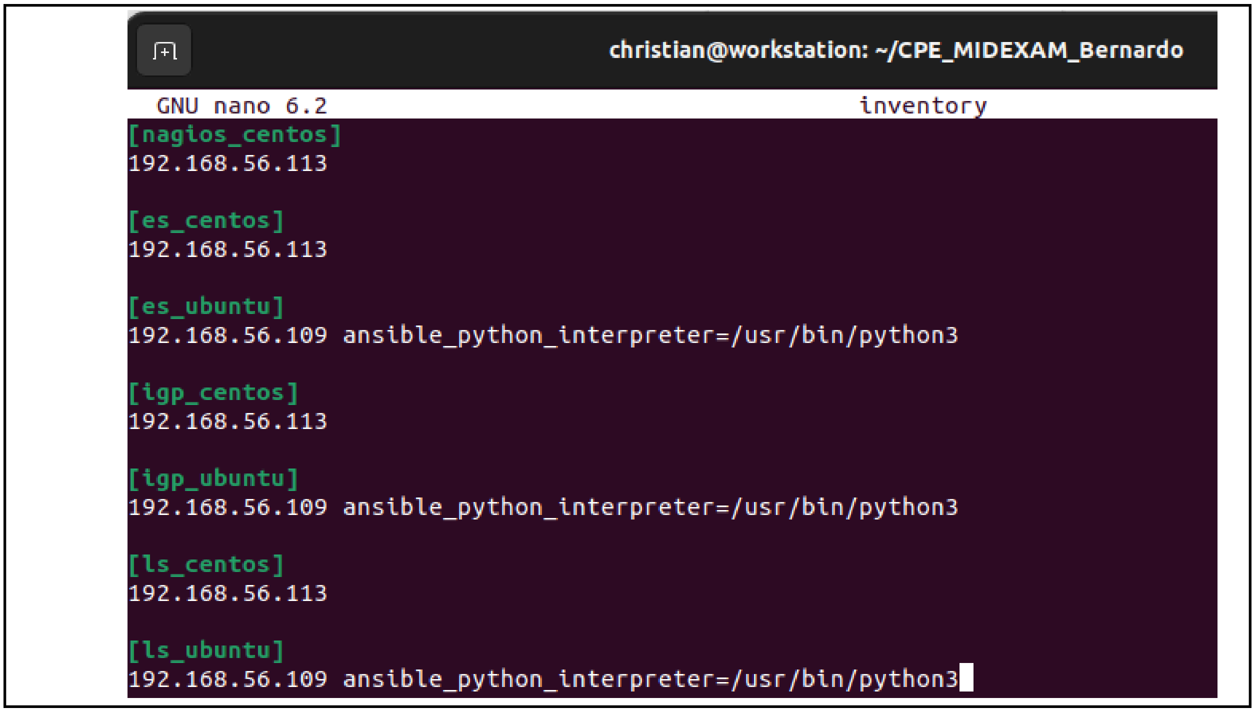
```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ cat config.yml
---

- 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"
```



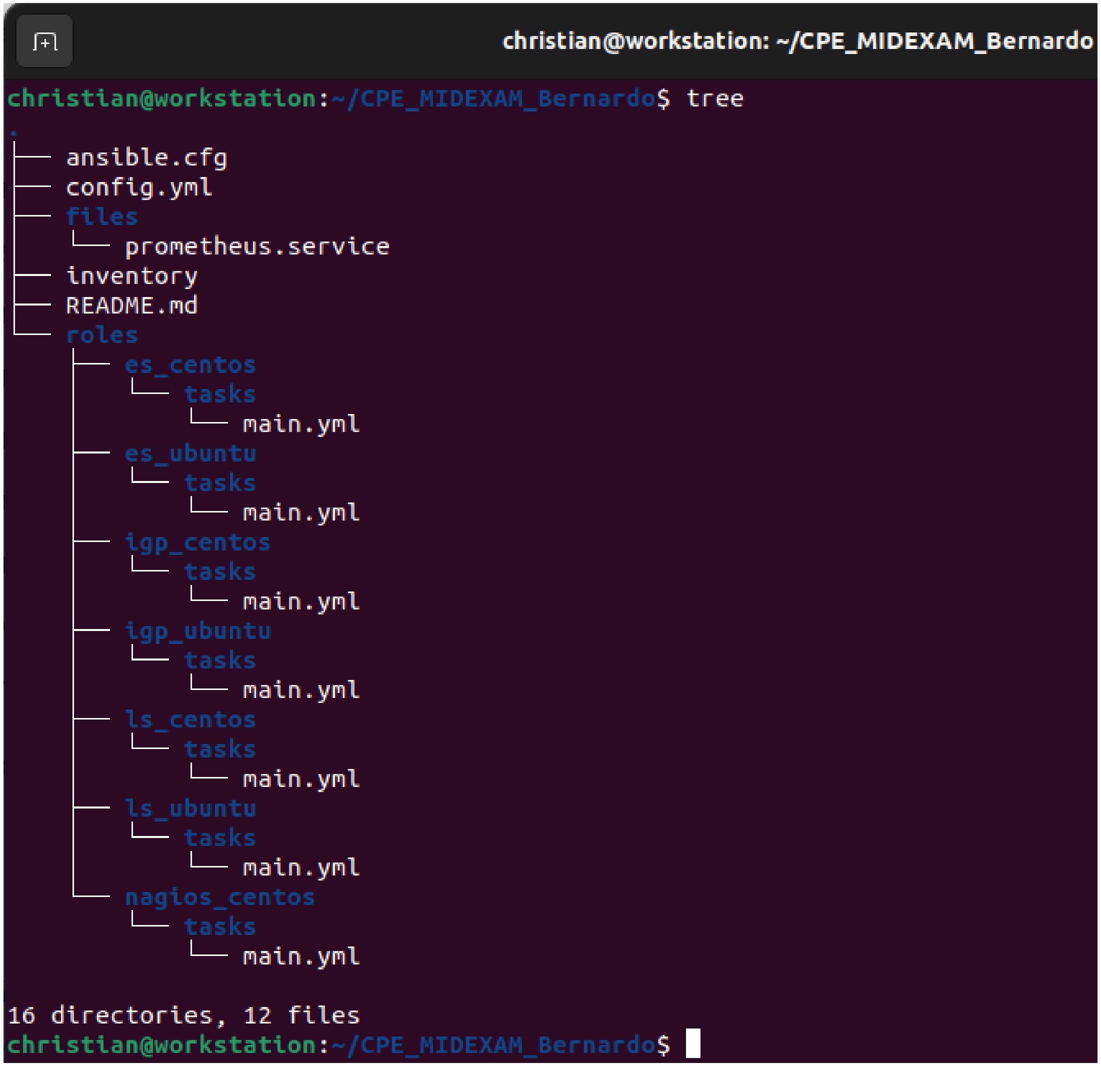
```
# ES Centos
- hosts: es_centos
  tags: es_centos, es_both
  become: true
  roles:
    - es_centos

# ES Ubuntu
- hosts: es_ubuntu
  tags: es_ubuntu, es_both
  become: true
  roles:
    - es_ubuntu

# Nagios Centos
- hosts: nagios_centos
  tags: nagios_centos
  become: true
  roles:
    - nagios_centos
```



```
# IGP (Influxdb, Grafana, Prometheus) Centos
- hosts: igp_centos
  tags: igp_centos, igp_both
  become: true
  roles:
    - igp_centos

# IGP (Influxdb, Grafana, Prometheus) Ubuntu
- hosts: igp_ubuntu
  tags: igp_ubuntu, igp_both
  become: true
  roles:
    - igp_ubuntu


# Lamp Stack (HTTPD + Php, Mariadb) Centos
- hosts: ls_centos
  tags: ls_centos, ls_both
  become: true
  roles:
    - ls_centos

# Lamp Stack (HTTPD + Php, Mariadb) Ubuntu
- hosts: ls_ubuntu
  tags: ls_ubuntu, ls_both
  become: true
  roles:
    - ls_ubuntu
christian@workstation:~/CPE_MIDEXAM_Bernardo$ 
```

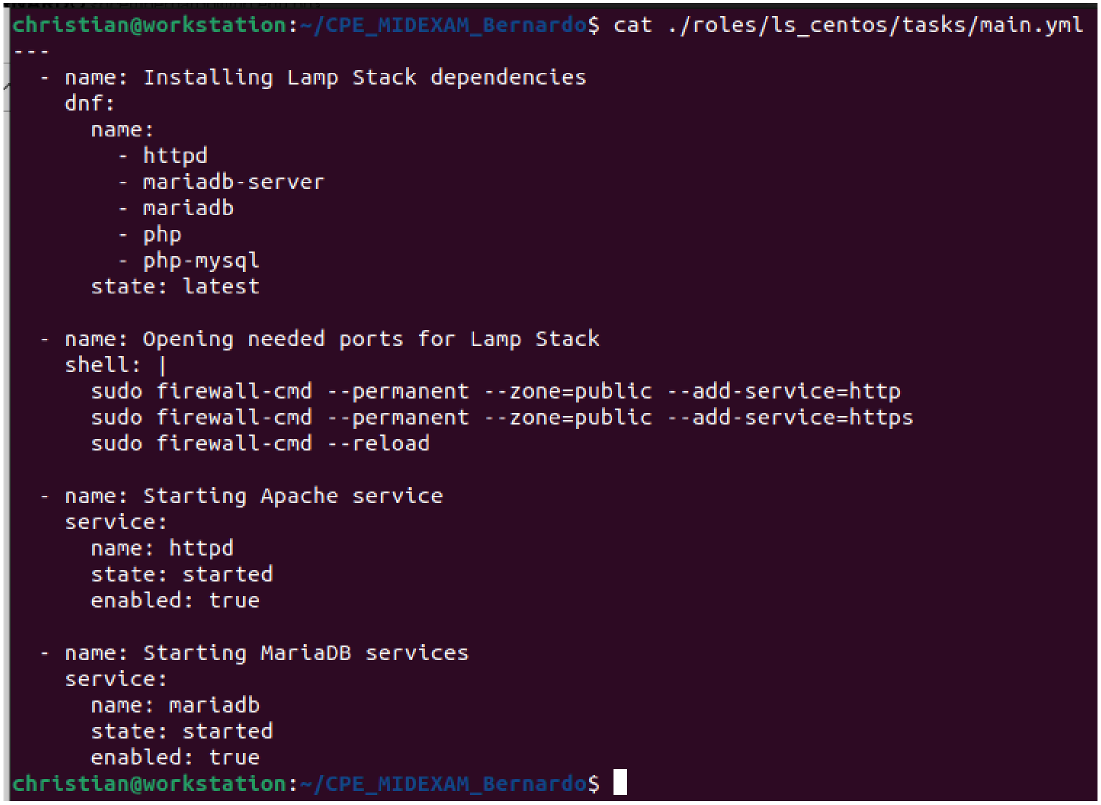


```
  GNU nano 6.2                                    inventory
[nagios_centos]
192.168.56.113

[es_centos]
192.168.56.113

[es_ubuntu]
192.168.56.109 ansible_python_interpreter=/usr/bin/python3

[igp_centos]
192.168.56.113

[igp_ubuntu]
192.168.56.109 ansible_python_interpreter=/usr/bin/python3

[ls_centos]
192.168.56.113

[ls_ubuntu]
192.168.56.109 ansible_python_interpreter=/usr/bin/python3
```



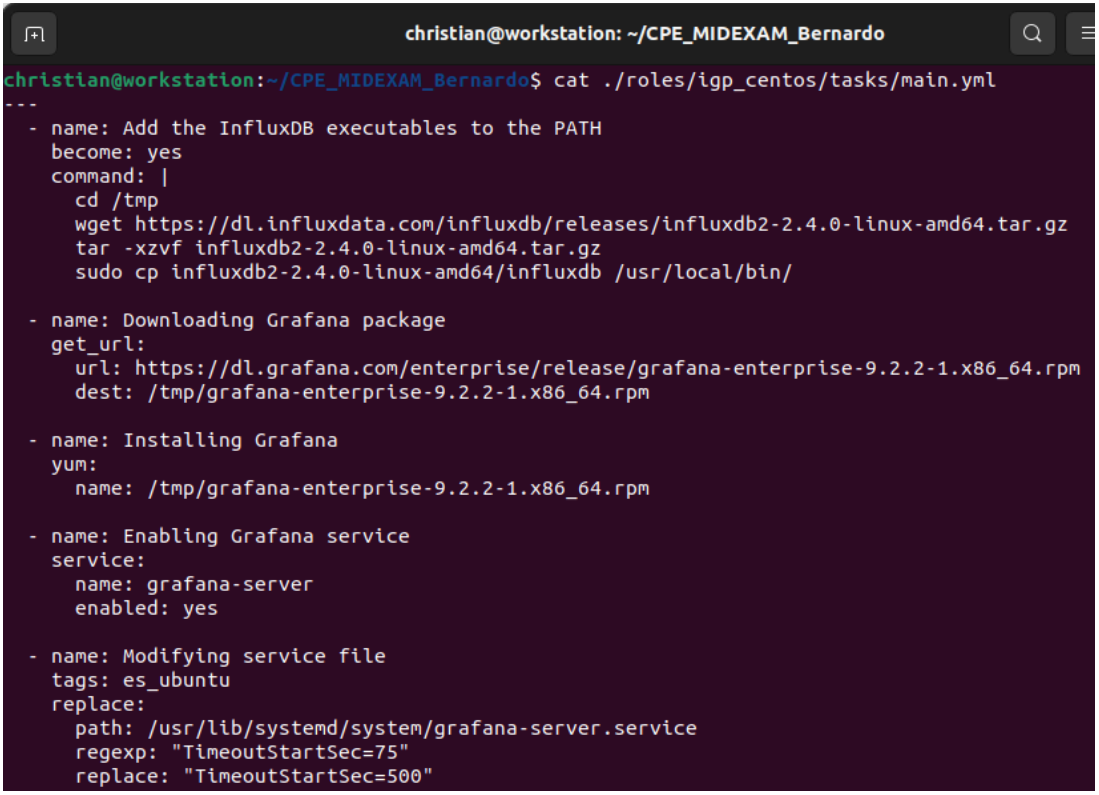
```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ tree
.
├── ansible.cfg
├── config.yml
├── files
│   └── prometheus.service
├── inventory
├── README.md
└── roles
    ├── es_centos
    │   └── tasks
    │       └── main.yml
    ├── es_ubuntu
    │   └── tasks
    │       └── main.yml
    ├── igp_centos
    │   └── tasks
    │       └── main.yml
    ├── igp_ubuntu
    │   └── tasks
    │       └── main.yml
    ├── ls_centos
    │   └── tasks
    │       └── main.yml
    ├── ls_ubuntu
    │   └── tasks
    │       └── main.yml
    └── nagios_centos
        └── tasks
            └── main.yml

16 directories, 12 files
christian@workstation:~/CPE_MIDEXAM_Bernardo$
```

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

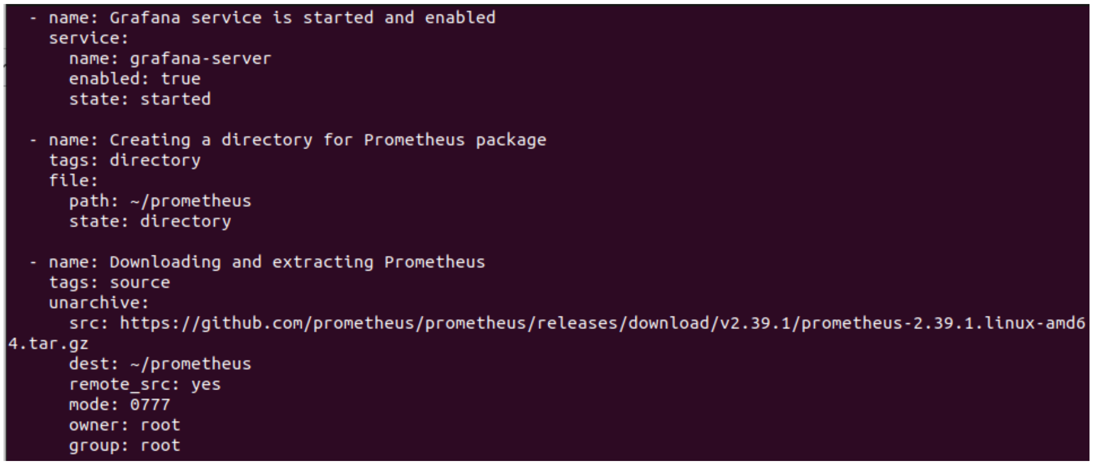


```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ cat ./roles/ls_centos/tasks/main.yml
---
  - name: Installing Lamp Stack dependencies
    dnf:
      name:
        - httpd
        - mariadb-server
        - mariadb
        - php
        - php-mysql
      state: latest

  - name: Opening needed ports for Lamp Stack
    shell: |
      sudo firewall-cmd --permanent --zone=public --add-service=http
      sudo firewall-cmd --permanent --zone=public --add-service=https
      sudo firewall-cmd --reload

  - name: Starting Apache service
    service:
      name: httpd
      state: started
      enabled: true

  - name: Starting MariaDB services
    service:
      name: mariadb
      state: started
      enabled: true
christian@workstation:~/CPE_MIDEXAM_Bernardo$
```

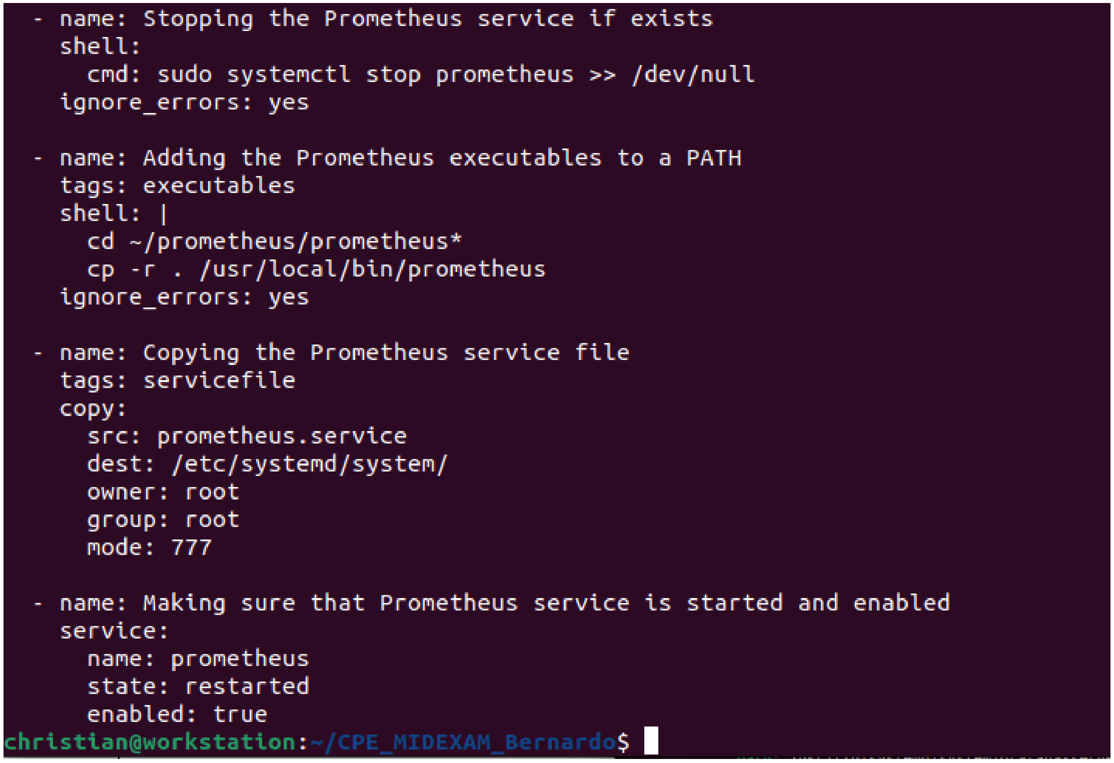
Install Grafana,Prometheus and Influxdb in separate hosts (Influxdb,Grafana,Prometheus)



```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ cat ./roles/igp_centos/tasks/main.yml
---
  - name: Add the InfluxDB executables to the PATH
    become: yes
    command: |
      cd /tmp
      wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.4.0-linux-amd64.tar.gz
      tar -xzvf influxdb2-2.4.0-linux-amd64.tar.gz
      sudo cp influxdb2-2.4.0-linux-amd64/influxdb /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:
      path: /usr/lib/systemd/system/grafana-server.service
      regexp: "TimeoutStartSec=75"
      replace: "TimeoutStartSec=500"
```

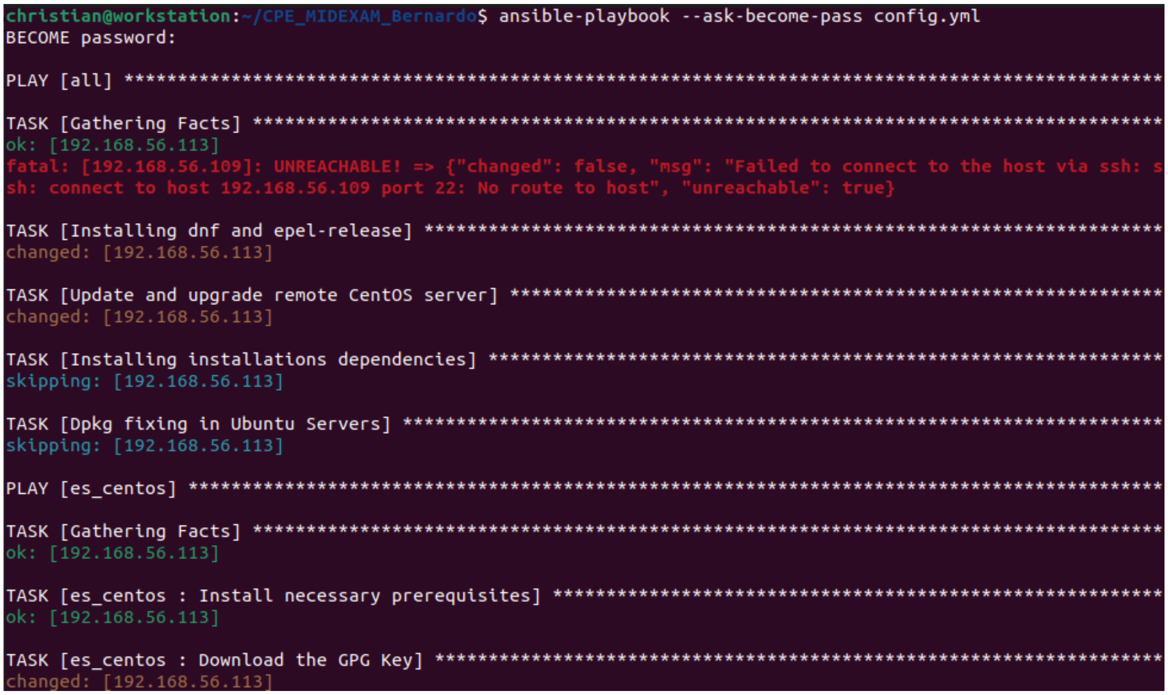


```
  - name: Grafana service is started and enabled
    service:
      name: grafana-server
      enabled: true
      state: started

  - name: Creating a directory for Prometheus package
    tags: directory
    file:
      path: ~/prometheus
      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-amd6
4.tar.gz
      dest: ~/prometheus
      remote_src: yes
      mode: 0777
      owner: root
      group: root
```

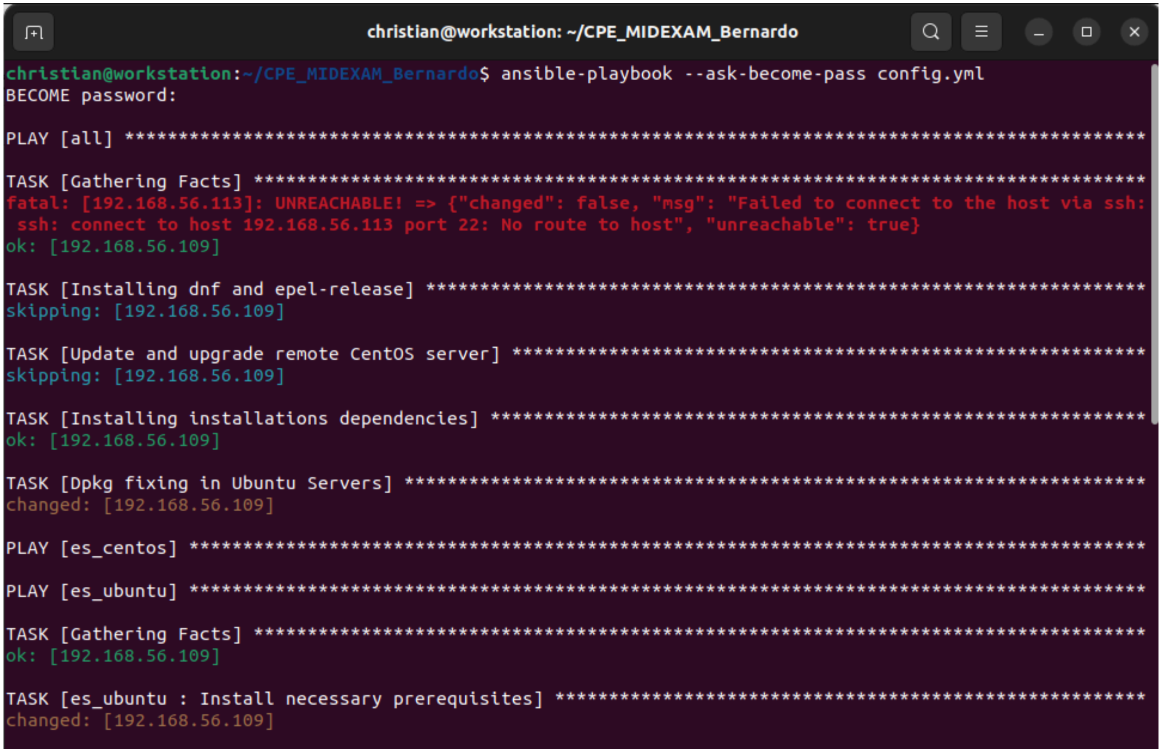


```
  - name: Stopping the Prometheus service if exists
    shell:
      cmd: sudo systemctl stop prometheus >> /dev/null
    ignore_errors: yes

  - name: Adding the Prometheus executables to a PATH
    tags: executables
    shell: |
      cd ~/prometheus/prometheus*
      cp -r . /usr/local/bin/prometheus
    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
christian@workstation:~/CPE_MIDEXAM_Bernardo$ 
```

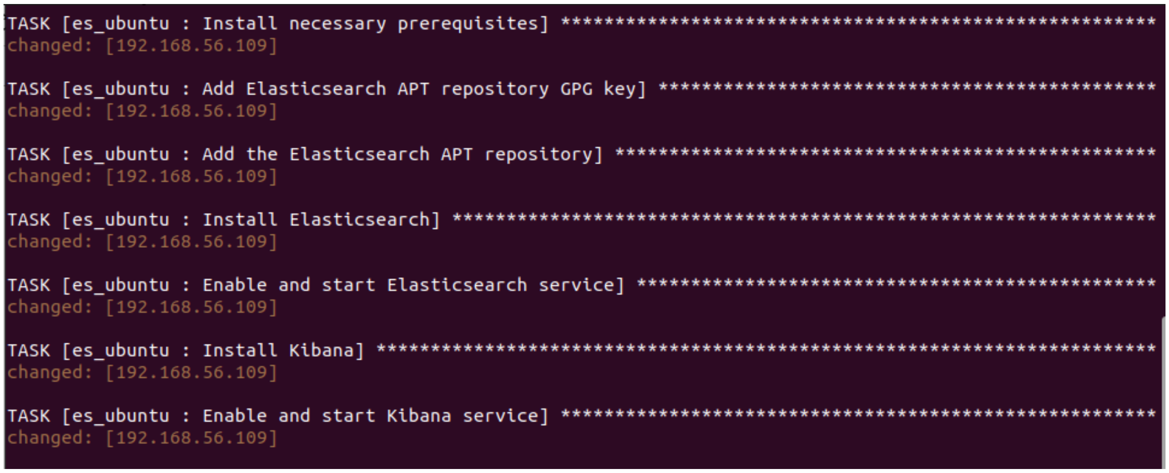
Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.



```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ ansible-playbook --ask-become-pass config.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.113]
fatal: [192.168.56.109]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: s
sh: connect to host 192.168.56.109 port 22: No route to host", "unreachable": true}

TASK [Installing dnf and epel-release] *****************************************************
changed: [192.168.56.113]

TASK [Update and upgrade remote CentOS server] *********************************************
changed: [192.168.56.113]

TASK [Installing installations dependencies] ***********************************************
skipping: [192.168.56.113]

TASK [Dpkg fixing in Ubuntu Servers] *******************************************************
skipping: [192.168.56.113]

PLAY [es_centos] ***************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.113]

TASK [es_centos : Install necessary prerequisites] *****************************************
ok: [192.168.56.113]

TASK [es_centos : Download the GPG Key] ****************************************************
changed: [192.168.56.113]
```

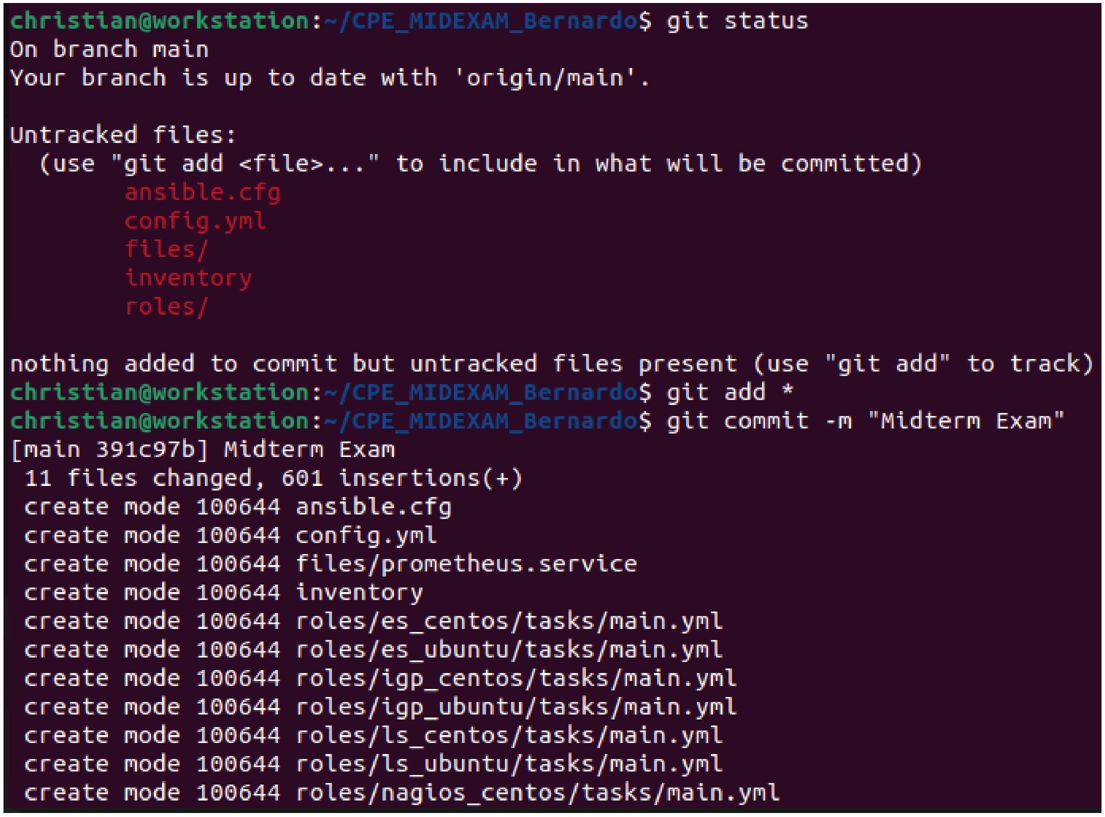


```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ ansible-playbook --ask-become-pass config.yml
BECOME password:

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

TASK [Gathering Facts] ***********************************************************************************
fatal: [192.168.56.113]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh:
 ssh: connect to host 192.168.56.113 port 22: No route to host", "unreachable": true}
ok: [192.168.56.109]

TASK [Installing dnf and epel-release] *******************************************************************
skipping: [192.168.56.109]

TASK [Update and upgrade remote CentOS server] ***********************************************************
skipping: [192.168.56.109]

TASK [Installing installations dependencies] *************************************************************
ok: [192.168.56.109]

TASK [Dpkg fixing in Ubuntu Servers] *********************************************************************
changed: [192.168.56.109]

PLAY [es_centos] *****************************************************************************************

PLAY [es_ubuntu] *****************************************************************************************

TASK [Gathering Facts] ***********************************************************************************
ok: [192.168.56.109]

TASK [es_ubuntu : Install necessary prerequisites] *******************************************************
changed: [192.168.56.109]
```

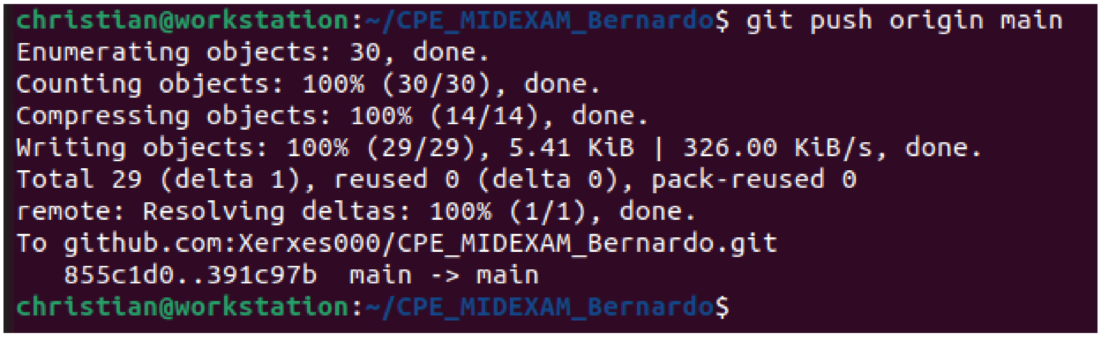


```
TASK [es_ubuntu : Install necessary prerequisites] *******************************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Add Elasticsearch APT repository GPG key] **********************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Add the Elasticsearch APT repository] **************************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Install Elasticsearch] *****************************************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Enable and start Elasticsearch service] ************************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Install Kibana] ************************************************************************
changed: [192.168.56.109]

TASK [es_ubuntu : Enable and start Kibana service] *******************************************************
changed: [192.168.56.109]
```

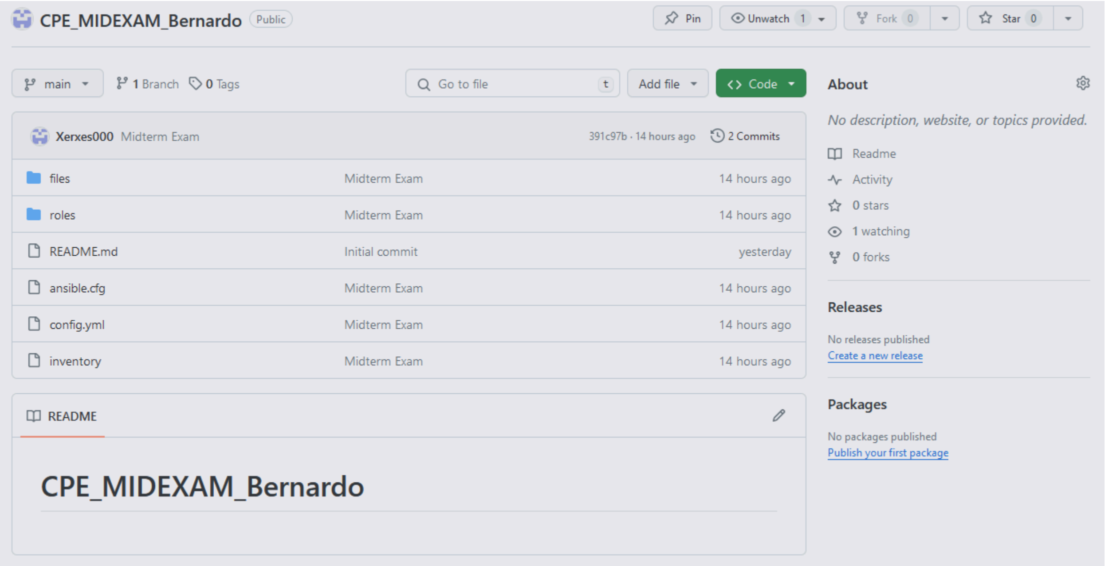
Document the push and commit from the local repository to GitHub.



```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible.cfg
        config.yml
        files/
        inventory
        roles/

nothing added to commit but untracked files present (use "git add" to track)
christian@workstation:~/CPE_MIDEXAM_Bernardo$ git add *
christian@workstation:~/CPE_MIDEXAM_Bernardo$ git commit -m "Midterm Exam"
[main 391c97b] Midterm Exam
 11 files changed, 601 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 config.yml
 create mode 100644 files/prometheus.service
 create mode 100644 inventory
 create mode 100644 roles/es_centos/tasks/main.yml
 create mode 100644 roles/es_ubuntu/tasks/main.yml
 create mode 100644 roles/igp_centos/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/nagios_centos/tasks/main.yml
```



```
christian@workstation:~/CPE_MIDEXAM_Bernardo$ git push origin main
Enumerating objects: 30, done.
Counting objects: 100% (30/30), done.
Compressing objects: 100% (14/14), done.
Writing objects: 100% (29/29), 5.41 KiB | 326.00 KiB/s, done.
Total 29 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:Xerxes000/CPE_MIDEXAM_Bernardo.git
   855c1d0..391c97b  main -> main
christian@workstation:~/CPE_MIDEXAM_Bernardo$
```



Xerxes000 Midterm Exam 391c97b · 14 hours ago 2 Commits

files	Midterm Exam	14 hours ago
roles	Midterm Exam	14 hours ago
README.md	Initial commit	yesterday
ansible.cfg	Midterm Exam	14 hours ago
config.yml	Midterm Exam	14 hours ago
inventory	Midterm Exam	14 hours ago

No description, website, or topics provided.

No releases published

No packages published

CPE_MIDEXAM_Bernardo

GitHub link:

- my github link: https://github.com/Xerxes000/CPE_MIDEXAM_Bernardo.git

Conclusions: (link your conclusion from the objective)

During this midterm practical exam, I successfully created an Ansible playbook to configure and manage a range of monitoring tools for enterprise systems. The first step in the process was to create a structured version control GitHub repository. This script simplifies the process of installing and configuring various tools, including Elastic Stack, Nagios, Grafana, Prometheus, InfluxDB, and LampStack. Roles, tags, and conditional statements were used to make the playbook adaptable and easy to debug. This hands-on training helped me become more proficient with Ansible and demonstrated the practical advantages of automation when managing complex system configurations. The playbook offers a practical method for putting monitoring tools into practice, aligning with the effectiveness and ease of use required for modern infrastructure management.