Name: Renier L. Lope	Date Performed: 11/06/2024
Course/Section: CPE31S2	Date Submitted: 11/06/2024
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st Sem (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 (screenshots and explanations)

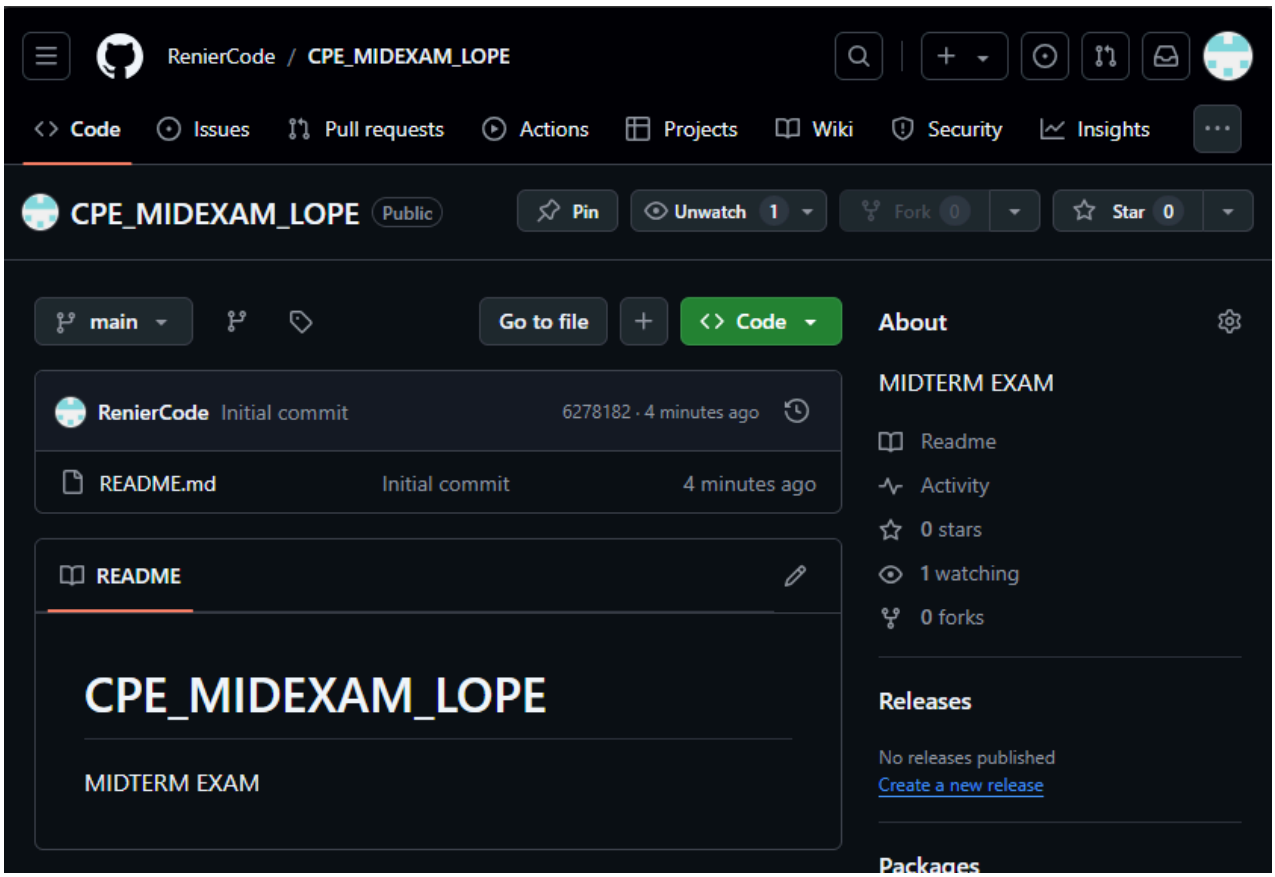


Figure 3.1: Create a repository in your Github Account and name it "CPE_MIDEXAM_LOPE".

```
rnrlope@workstation:~$ git clone git@github.com:RenierCode/CPE_MIDEXAM_LOPE
Cloning into 'CPE_MIDEXAM_LOPE'...
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.
```

Figure 3.2: Use git clone to clone the repository in your local machine.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ nano ansible.cfg
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ cat ansible.cfg
[defaults]
inventory = inventory
remote_user = rnrlope
host_key_checking = True
deprecation_warnings = False
```

Figure 3.3: Create an ansible file to configure default variables.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ nano inventory
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ cat inventory
[web_servers]
server1

[db_servers]
centOS
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$
```

Figure 3.4: Create an inventory file to define manage nodes.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ nano config.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ cat config.yml
---
- hosts: all
  become: true
  pre_tasks:

  - name: update repository index (CentOS)
    tags: always
    dnf:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

- hosts: all
  become: true
  roles:
    -  base

- hosts: web_servers:db_servers
  become: true
  roles:
    - ElasticStack
    - Grafana
    - Prometheus
    - Influxdb
    - LampStack

- hosts: db_servers
  become: true
  roles:
    - Nagios
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$
```

Figure 3.5: Create a playbook file named “config.yml”. This playbook when executed will play the various tasks within the desired roles.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ mkdir roles
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ cd roles
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$
```

Figure 3.6: Create a directory named “roles” to create various roles.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ mkdir base ElasticStack Nagios
Grafana Prometheus Influxdb LampStack
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ ls
base  ElasticStack  Grafana  Influxdb  LampStack  Nagios  Prometheus
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$
```

Figure 3.7: Create a directory named “roles” to create various roles.

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ cd base
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/base$ mkdir tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/base$ cd tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/base/tasks$ nano main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/base/tasks$ cat main.yml
---
- name: install updates (CentOS)
  tags: always
  dnf:
    update_only: yes
    update_cache: yes
  when: ansible_distribution == "CentOS"

- name: install updates (Ubuntu)
  tags: always
  apt:
    upgrade: dist
    update_cache: yes
  when: ansible_distribution == "Ubuntu"
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/base/tasks$
```

Figure 3.8: Under roles/base create another director named “tasks” and inside it create a playbook named “main.yml”. This playbook will update both Ubuntu and CentOS.

Elastic Stack:

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ cd ElasticStack
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/ElasticStack$ mkdir tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/ElasticStack$ cd tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/ElasticStack/tasks$ nano main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/ElasticStack/tasks$ cat main.yml
---
- name: install java (Ubuntu)
  apt:
    name: openjdk-11-jdk
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: install java (CentOS)
  dnf:
    name: java-11-openjdk
    state: latest
  when: ansible_distribution == "CentOS"

- name: Install EPEL repository
  yum:
    name: epel-release
    state: latest
  when: ansible_distribution == "CentOS"

- name: Add GPG key for ElasticSearch (Ubuntu)
  tags: ubuntu
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present
  when: ansible_distribution == "Ubuntu"

- name: Allow Port 9200 through Firewall (CentOS)
  firewalld:
    zone: public
    port: 9200/tcp
    permanent: yes
    state: enabled
    immediate: yes
  when: ansible_distribution == "CentOS"

- name: Allow Port 9200 through Firewall (Ubuntu)
  ufw:
    rule: allow
    port: 9200
    proto: tcp
  when: ansible_distribution == "Ubuntu"

- name: Add ElasticSearch to APT repository (Ubuntu)
  tags: ubuntu
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/7.x/apt stable main"
    #filename: 'elastic-7.x'
  when: ansible_distribution == "Ubuntu"

- name: Install ElasticSearch to Yum repository (CentOS)
  yum_repository:
    name: elasticsearch
    description: ElasticSearch Repository
    baseurl: https://artifacts.elastic.co/packages/7.x/yum
    gpgcheck: yes
    gpgkey: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    enabled: yes
  when: ansible_distribution == "CentOS"
```

```
- name: Configure ElasticSearch
  blockinfile:
    path: /etc/elasticsearch/elasticsearch.yml
    block: |
      # ElasticSearch Configuration

      cluster.name: my-cluster
      node.name: dev-node-1
      network.host: 0.0.0.0
      http.port: 9200
      discovery.type: single-node
      path.data: /var/lib/elasticsearch
      path.logs: /var/log/elasticsearch
      bootstrap.memory_lock: true
    state: present
    create: yes

- name: Configure Kibana
  blockinfile:
    path: /etc/kibana/kibana.yml
    block: |
      # Kibana Configuration

      server.port: 5601
      server.host: "localhost"
      server.name: "Lope-Act10"
      elasticsearch.hosts: ["http://localhost:9200"]
      kibana.index: ".kibana"
      elasticsearch.requestTimeout: 180000
    state: present
    create: yes
```

```
- name: Configure Logstash
  blockinfile:
    path: /etc/logstash/conf.d/logstash.conf
    block: |
      # Logstash Configuration

      input {
        beats {
          port => 5044
          host => "127.0.0.1"
         }
      }

      filter {
      # nginx access log
        if [source] =~ /\/(access)\d{0,10}\.(log)/ {
          grok {
            match => {"message" => "%{COMBINEDAPACHELOG}"}
            add_tag => ["nginx_access_log"]
          }
          mutate {
            rename => {"timestamp" => "log_timestamp"}
          }
        }
      }

      output {
        elasticsearch { hosts => ["localhost:9200"] }
        stdout { codec => rubydebug }
      }
    state: present
    create: yes
```

```
- name: Install ElasticSearch, Kibana, and Logstash
  package:
    name:
      - elasticsearch
      - kibana
      - logstash
    state: latest

- name: Force systemd to reread configs
  systemd:
    daemon_reload: yes

- name: Enable ElasticSearch, Kibana, and Logstash Service
  vars:
     elastic_services:
       - elasticsearch
       - kibana
       - logstash
  service:
     name: "{{ item }}"
     enabled: yes
     state: started
  loop: "{{ elastic_services }}"
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/ElasticStack/tasks$
```

Figure 3.9 - 3.11: : Under roles/ElasticStack create another director named “tasks” and inside it create a playbook named “main.yml”. This playbook will set up ElasticStack on both Ubuntu and CentOS by configuring and installing elasticsearch, kibana, and logstash.

Nagios:

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ cd Nagios
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Nagios$ mkdir tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Nagios$ cd tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Nagios/tasks$ nano main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Nagios/tasks$ cat main.yml
---
- name: Install nagios requirements (CentOS)
  dnf:
    name:
      - gcc
      - glibc
      - glibc-common
      - wget
      - unzip
      - httpd
      - php
      - gd
      - gd-devel
      - perl
      - postfix
      - openssl-devel
    state: latest
  become: true
  when: ansible_distribution == "CentOS"

- name: Install EPEL repo (CentOS)
  dnf:
    name: epel-release
    state: latest
  when: ansible_distribution == "CentOS"

- name: Install Nagios (CentOS)
  dnf:
    name: nagios
    state: latest
  when: ansible_distribution == "CentOS"

- name: Start and Enable Nagios (CentOS)
  service:
    name: nagios
    state: restarted
    enabled: true
  when: ansible_distribution == "CentOS"

rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Nagios/tasks$
```

Figure 3.12: : Under roles/Nagios create another director named “tasks” and inside it create a playbook named “main.yml”. This playbook will Configure and Install the requirements of Nagios and Nagios itself on CentOS.

Grafana:

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ cd Grafana
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Grafana$ mkdir tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Grafana$ cd tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Grafana/tasks$ nano main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Grafana/tasks$ cat main.yml
---
- name: Add GPG key for Grafana (Ubuntu)
  tags: ubuntu
  apt_key:
    url: https://rpm.grafana.com/gpd.key
    state: present
  when: ansible_distribution == "Ubuntu"

- name: Add Grafana to APT repository (Ubuntu)
  tags: ubuntu
  apt_repository:
    repo: "deb https://apt.grafana.com stable main"
  when: ansible_distribution == "Ubuntu"

- name: Download Grafana GPG key
  get_url:
    url: https://rpm.grafana.com/gpg.key
    dest: /tmp/gpg.key

- name: Import Grafana GPG key
  shell: sudo rpm --import /tmp/gpg.key
- name: Add Grafana YUM Repository (CentOS)
  copy:
    content: |
      [grafana]
      name=grafana
      baseurl=https://rpm.grafana.com
      repo_gpgcheck=1
      enabled=1
      gpgcheck=1
      gpgkey=https://rpm.grafana.com/gpg.key
      sslverify=1
      sslcacert=/etc/pki/tls/certs/ca-bundle.crt
      exclude=*beta*
    dest: /etc/yum.repos.d/grafana.repo
  when: ansible_distribution == "CentOS"

- name: Install Grafana
  package:
    name: grafana
    state: latest

- name: Enabled Grafana Service
  service:
    name: grafana
    enabled: yes
    state: started

rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Grafana/tasks$
```

Figure 3.13: : Under roles/Grafana create another director named “tasks” and inside it create a playbook named “main.yml”. This playbook will Configure and Install Grafana on both Ubuntu and CentOS.

Prometheus:

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles$ cd Prometheus
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Prometheus$ mkdir tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Prometheus$ cd tasks
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Prometheus/tasks$ nano main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Prometheus/tasks$ cat main.yml
---
- name: Copy Prometheus binaries
  copy:
    src: /usr/local/bin/prometheus-2.30.0.linux-amd64/prometheus
    dest: /usr/local/bin/prometheus
    mode: 0755
    remote_src: yes
  when: ansible_distribution == "CentOS"

- 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:
    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
```

```
- name: Force systemd to reread configs
  systemd:
    daemon_reload: yes
  when: ansible_distribution == "CentOS"

- name: Open port 9090 (Ubuntu)
  ufw:
    rule: allow
    port: 9090
    proto: tcp
    state: enabled
  when: ansible_distribution == "Ubuntu"

- name: Allow Prometheus for Firewall (CentOS)
  firewalld:
    port: 9090/tcp
    permanent: yes
    state: enabled
  when: ansible_distribution == "CentOS"

- 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-amd6
4.tar.gz
    dest: /usr/local/bin
    remote_src: yes
    mode: 0755
    owner: root
    group: root
  when: ansible_distribution == "CentOS"

- name: install apache2 (Ubuntu)
  apt:
    name: apache2
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: install apache (CentOS)
  dnf:
    name: httpd
    state: latest
  when: ansible_distribution == "CentOS"

- name: Start apache2 (Ubuntu)
  systemd:
     name: apache2
     enabled: yes
     state: started
  when: ansible_distribution == "Ubuntu"
```

```
- name: Start apache2 (CentOS)
  systemd:
     name: httpd
     enabled: yes
     state: started
  when: ansible_distribution == "CentOS"

- name: Start Prometheus Service
  systemd:
     name: prometheus
     enabled: yes
     state: started

rnrlope@workstation:~/CPE_MIDEXAM_LOPE/roles/Prometheus/tasks$
```

Figure 3.14 - 3.17: : Under roles/Prometheus create another director named “tasks” and inside it create a playbook named “main.yml”. This playbook will Configure and Install Prometheus on both Ubuntu and CentOS.

```
TASK [Prometheus : Start apache2 (Ubuntu)] *************************************************************
skipping: [centOS]
ok: [server1]

TASK [Prometheus : Start apache2 (CentOS)] *************************************************************
skipping: [server1]
ok: [centOS]

TASK [Start Prometheus Service] ************************************************************************
changed: [server1]
ok: [centOS]

PLAY [db_servers] **************************************************************************************

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

TASK [Nagios : Install nagios requirements (CentOS)] ***************************************************
ok: [centOS]

TASK [Nagios : Install EPEL repo (CentOS)] *************************************************************
ok: [centOS]

TASK [Install Nagios (CentOS)] *************************************************************************
ok: [centOS]

TASK [Start and Enable Nagios (CentOS)] ****************************************************************
changed: [centOS]

PLAY RECAP *********************************************************************************************
centOS                     : ok=33   changed=4    unreachable=0    failed=0    skipped=10   rescued=0
 ignored=0
server1                    : ok=21   changed=1    unreachable=0    failed=0    skipped=17   rescued=0
 ignored=0
```

Figure 3.18: : Executing config.yml.

webb_servers (Ubuntu)

ElasticStack

```
rnrlope@server1:~$ systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vend
   Active: active (running) since Wed 2024-11-06 07:49:38 +08; 2h 16min ago
     Docs: https://www.elastic.co
 Main PID: 1183 (java)
    Tasks: 67 (limit: 4541)
   CGroup: /system.slice/elasticsearch.service
           ├─1183 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.netwo
           └─2510 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86

Warning: Journal has been rotated since unit was started. Log output is incompl
rnrlope@server1:~$ systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-11-06 07:45:41 +08; 2h 20min ago
     Docs: https://www.elastic.co
 Main PID: 1193 (node)
    Tasks: 11 (limit: 4541)
   CGroup: /system.slice/kibana.service
           └─1193 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/
```

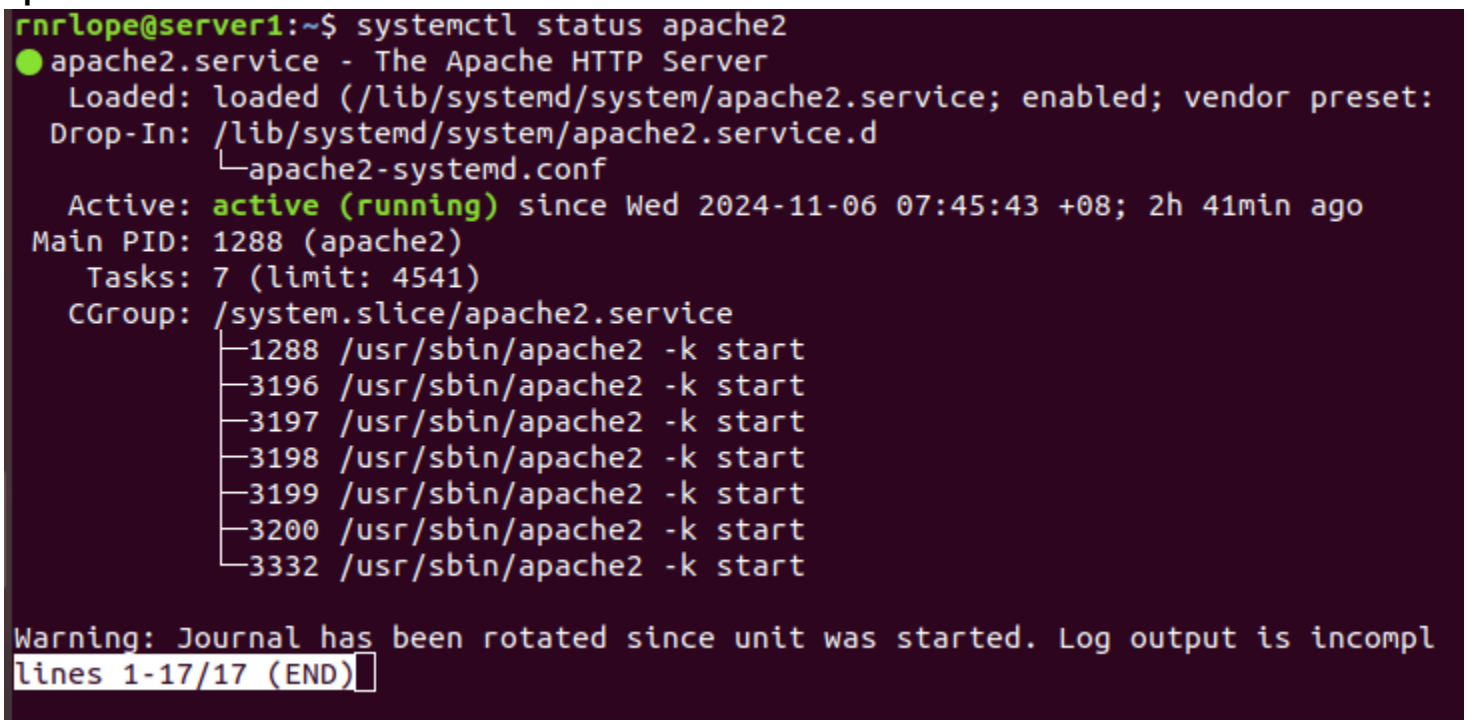
```
rnrlope@server1:~$ systemctl status logstash
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor preset
   Active: active (running) since Wed 2024-11-06 07:45:34 +08; 2h 20min ago
 Main PID: 801 (java)
    Tasks: 36 (limit: 4541)
   CGroup: /system.slice/logstash.service
           └─801 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -XX:+UseConcMar

Warning: Journal has been rotated since unit was started. Log output is incompl
rnrlope@server1:~$
```

Grafana

Prometheus

Apache2



db_servers (CentOS)

ElasticStack

```
[rnrlope@localhost ~]$ systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendo
r preset: disabled)
   Active: active (running) since Tue 2024-11-05 18:49:47 EST; 2h 23min ago
     Docs: https://www.elastic.co
 Main PID: 1271 (java)
    Tasks: 63
   CGroup: /system.slice/elasticsearch.service
           ├─1271 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.net...
           └─2566 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x...

Nov 05 18:47:25 localhost.localdomain systemd[1]: Starting Elasticsearch...
Nov 05 18:48:24 localhost.localdomain systemd-entrypoint[1271]: Nov 05, 2024 ...
Nov 05 18:48:24 localhost.localdomain systemd-entrypoint[1271]: WARNING: COMP...
Nov 05 18:49:47 localhost.localdomain systemd[1]: Started Elasticsearch.
Hint: Some lines were ellipsized, use -l to show in full.
[rnrlope@localhost ~]$ systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset: d
isabled)
   Active: active (running) since Tue 2024-11-05 18:47:25 EST; 2h 26min ago
     Docs: https://www.elastic.co
 Main PID: 1270 (node)
    Tasks: 11
   CGroup: /system.slice/kibana.service
           └─1270 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bi...
```

```
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor preset:
 disabled)
   Active: active (running) since Tue 2024-11-05 18:46:41 EST; 2h 27min ago
 Main PID: 758 (java)
    Tasks: 36
   CGroup: /system.slice/logstash.service
           └─758 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -XX:+UseConcM...

Nov 05 18:50:41 localhost.localdomain logstash[758]: [2024-11-05T18:50:41,778...
Nov 05 18:50:41 localhost.localdomain logstash[758]: [2024-11-05T18:50:41,987...
Nov 05 18:50:42 localhost.localdomain logstash[758]: [2024-11-05T18:50:42,005...
Nov 05 18:50:42 localhost.localdomain logstash[758]: [2024-11-05T18:50:42,078...
Nov 05 18:50:42 localhost.localdomain logstash[758]: [2024-11-05T18:50:42,679...
Nov 05 18:50:51 localhost.localdomain logstash[758]: [2024-11-05T18:50:51,589...
Nov 05 18:50:51 localhost.localdomain logstash[758]: [2024-11-05T18:50:51,607...
Nov 05 18:50:52 localhost.localdomain logstash[758]: [2024-11-05T18:50:52,869...
Nov 05 18:50:54 localhost.localdomain logstash[758]: [2024-11-05T18:50:54,762...
Nov 05 18:50:54 localhost.localdomain logstash[758]: [2024-11-05T18:50:54,909...
Hint: Some lines were ellipsized, use -l to show in full.
```

Nagios

```
[rnrlope@localhost ~]$ systemctl status nagios
● nagios.service - Nagios Core 4.4.14
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; vendor prese
t: disabled)
   Active: active (running) since Tue 2024-11-05 20:39:25 EST; 34min ago
     Docs: https://www.nagios.org/documentation
 Main PID: 13901 (nagios)
    Tasks: 6
   CGroup: /system.slice/nagios.service
           ├─13901 /usr/sbin/nagios -d /etc/nagios/nagios.cfg
           ├─13902 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
           ├─13903 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
           ├─13904 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
           ├─13905 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
           └─13909 /usr/sbin/nagios -d /etc/nagios/nagios.cfg

Nov 05 20:39:25 localhost.localdomain nagios[13901]: qh: help for the query h...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: wproc: Successfully regi...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: wproc: Registry request:...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: wproc: Registry request:...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: wproc: Registry request:...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: wproc: Registry request:...
Nov 05 20:39:25 localhost.localdomain nagios[13901]: Successfully launched co...
Nov 05 21:11:14 localhost.localdomain nagios[13901]: SERVICE ALERT: localhost...
Nov 05 21:12:22 localhost.localdomain nagios[13901]: SERVICE ALERT: localhost...
Nov 05 21:13:31 localhost.localdomain nagios[13901]: SERVICE ALERT: localhost...
```

Grafana

Prometheus

```
[rnrlope@localhost ~]$ systemctl status prometheus
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor prese
t: disabled)
   Active: active (running) since Tue 2024-11-05 18:47:25 EST; 2h 26min ago
 Main PID: 1268 (prometheus)
    Tasks: 9
   CGroup: /system.slice/prometheus.service
           └─1268 /usr/local/bin/prometheus --config.file /etc/prometheus/pro...

Nov 05 18:48:25 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:27 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:27 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:27 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:28 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:28 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 18:48:29 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 20:48:35 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 20:48:35 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Nov 05 20:48:35 localhost.localdomain prometheus[1268]: level=info ts=2024-11...
Hint: Some lines were ellipsized, use -l to show in full.
```

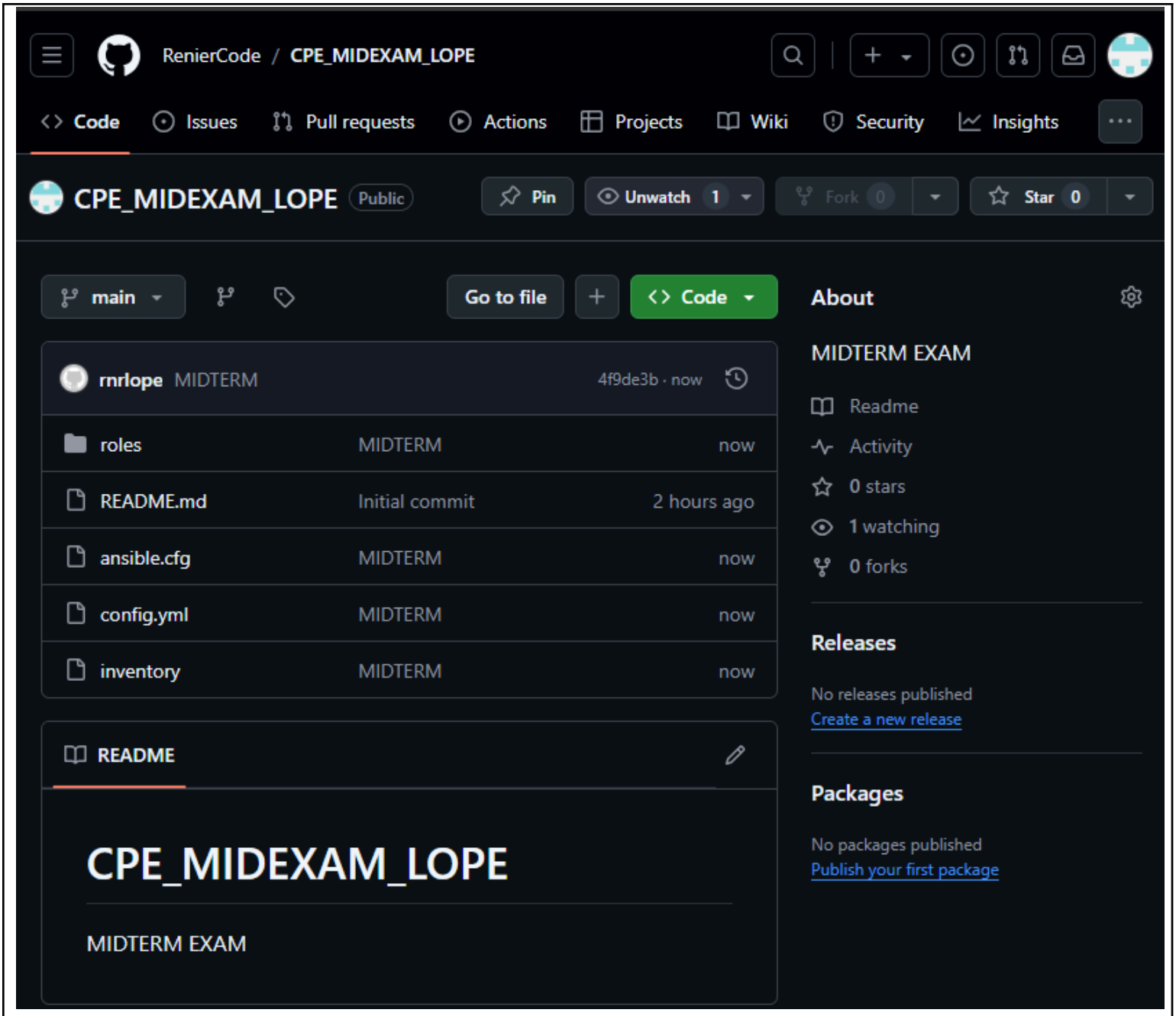
Httpd

```
[rnrlope@localhost ~]$ systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset
: disabled)
   Active: active (running) since Tue 2024-11-05 18:47:37 EST; 2h 40min ago
     Docs: man:httpd(8)
           man:apachectl(8)
 Main PID: 1296 (httpd)
   Status: "Total requests: 32; Current requests/sec: 0; Current traffic:   0 B/
sec"
    Tasks: 7
   CGroup: /system.slice/httpd.service
           ├─1296 /usr/sbin/httpd -DFOREGROUND
           ├─2137 /usr/sbin/httpd -DFOREGROUND
           ├─2138 /usr/sbin/httpd -DFOREGROUND
           ├─2139 /usr/sbin/httpd -DFOREGROUND
           ├─2140 /usr/sbin/httpd -DFOREGROUND
           ├─2141 /usr/sbin/httpd -DFOREGROUND
           └─3015 /usr/sbin/httpd -DFOREGROUND

Nov 05 18:47:25 localhost.localdomain systemd[1]: Starting The Apache HTTP Se...
Nov 05 18:47:35 localhost.localdomain httpd[1296]: AH00558: httpd: Could not ...
Nov 05 18:47:37 localhost.localdomain systemd[1]: Started The Apache HTTP Ser...
Hint: Some lines were ellipsized, use -l to show in full.
```

GITPUSH:

```
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ nano config.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ git add --all
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ git commit -m "MIDTERM"
[main 4f9de3b] MIDTERM
 10 files changed, 426 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 config.yml
 create mode 100644 inventory
 create mode 100644 roles/ElasticStack/tasks/.main.yml.swp
 create mode 100644 roles/ElasticStack/tasks/main.yml
 create mode 100644 roles/Grafana/tasks/.main.swp
 create mode 100644 roles/Grafana/tasks/main.yml
 create mode 100644 roles/Nagios/tasks/main.yml
 create mode 100644 roles/Prometheus/tasks/main.yml
 create mode 100644 roles/base/tasks/main.yml
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$ git push origin main
Counting objects: 23, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (14/14), done.
Writing objects: 100% (23/23), 4.35 KiB | 4.35 MiB/s, done.
Total 23 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:RenierCode/CPE_MIDEXAM_LOPE
   6278182..4f9de3b  main -> main
rnrlope@workstation:~/CPE_MIDEXAM_LOPE$
```



GitHub link:
https://github.com/RenierCode/CPE_MIDEXAM_LOPE.git

Conclusions: (link your conclusion from the objective)

- **I manage to create and design a workflow that Configure and Install elasticsearch, kibana, and logstash to set up ElasticStack and also a workflow that Configure and Install Nagios, Prometheus, and Grafana. This Midterm Exam makes use of what we have learned in the midterm period such as using ansible playbooks, applying roles and using tags. This Exam have given me insights on how to more efficiently utilize ansible playbooks**