Name: Baltazar, Paul Eimar R.	**Date Performed:** Nov 6, 2024
Course/Section: CPE 212 - CPE 31S2	**Date Submitted:** Nov 6, 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.

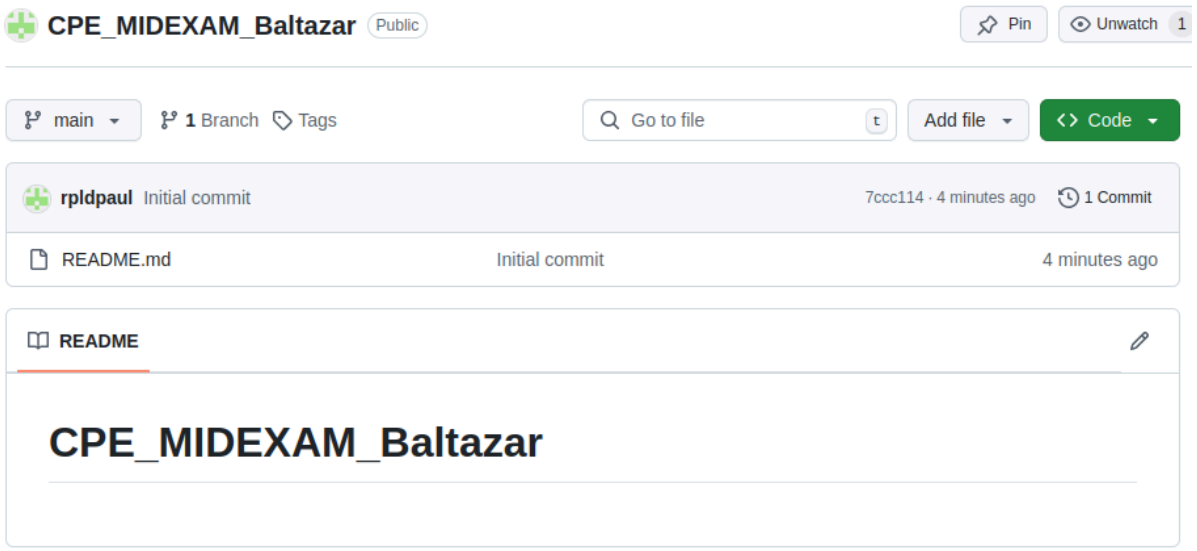
2. Instructions

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)

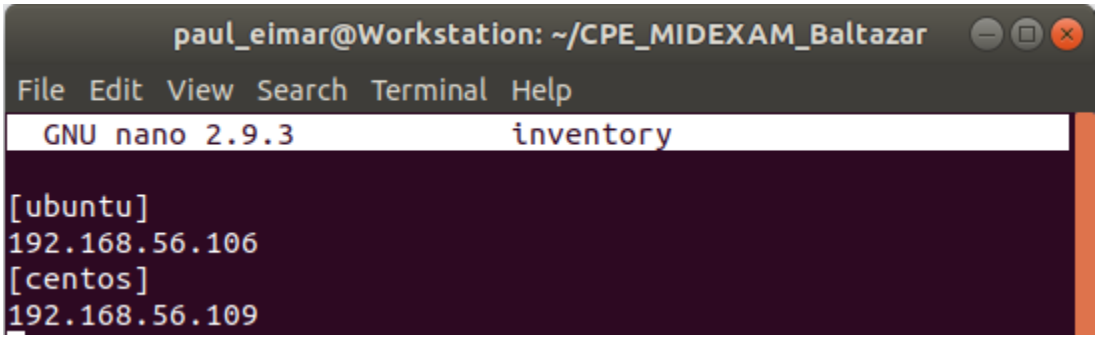
Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.

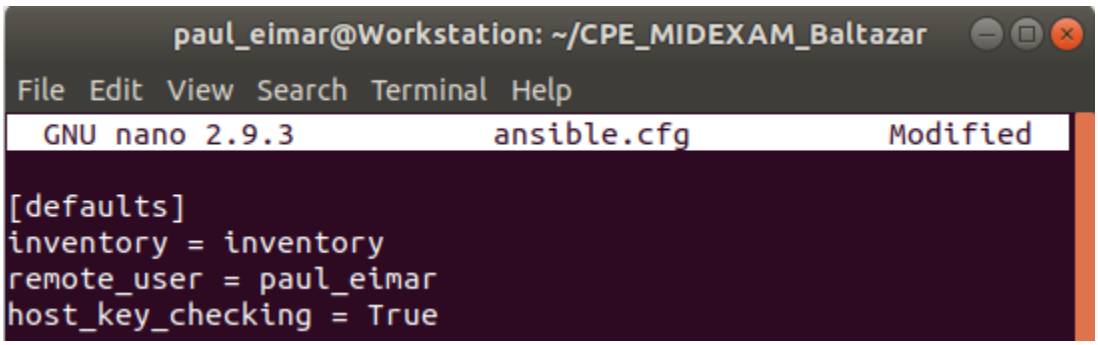
https://github.com/rpldpaul/CPE_MIDEXAM_Baltazar



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:





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

midexam.yml (main playbook)

```yaml
---
- name: Install Elasticsearch
  hosts: centos, ubuntu
  become: true
  roles:
   - elasticsearch
   - elasticsearchubuntu

- name: Install Kibana
  hosts: ubuntu
  become: true
  roles:
   - kibana

- name: Install Logstash
  hosts: ubuntu
  become: true
  roles:
   - logstash

- name: Install Nagios
  hosts: ubuntu
  become: true
  roles:
    - nagios

- name: Install MariaDB
  hosts: ubuntu, centos
  become: true
  roles:
    - mariadb
```

roles/elasticsearch/tasks/main.yml (CentOS)

```
- name: Installing Java
  tags: elasticsearch
  yum:
    name: java-11-openjdk
    state: present
  when: ansible_distribution == "CentOS"

- name: Installing EPEL repository
  tags: elasticsearch
  yum:
    name: epel-release
    state: latest
  when: ansible_distribution == "CentOS"

- name: Installing Elastic Search YUM repository
  tags: elasticsearch
  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: Elastic Search Installation
  tags: elasticsearch
  yum:
    name: elasticsearch
    state: present
  when: ansible_distribution == "CentOS"

- name: Configuring Elastic Search
  tags: elasticsearch
  template:
    src: elasticsearch.yml.j2
    dest: /etc/elasticsearch/elasticsearch.yml
  when: ansible_distribution == "CentOS"

- name: Starting Elastic Search
  tags: elasticsearch
  service:
    name: elasticsearch
    state: restarted
    enabled: yes
  when: ansible_distribution == "CentOS"
```

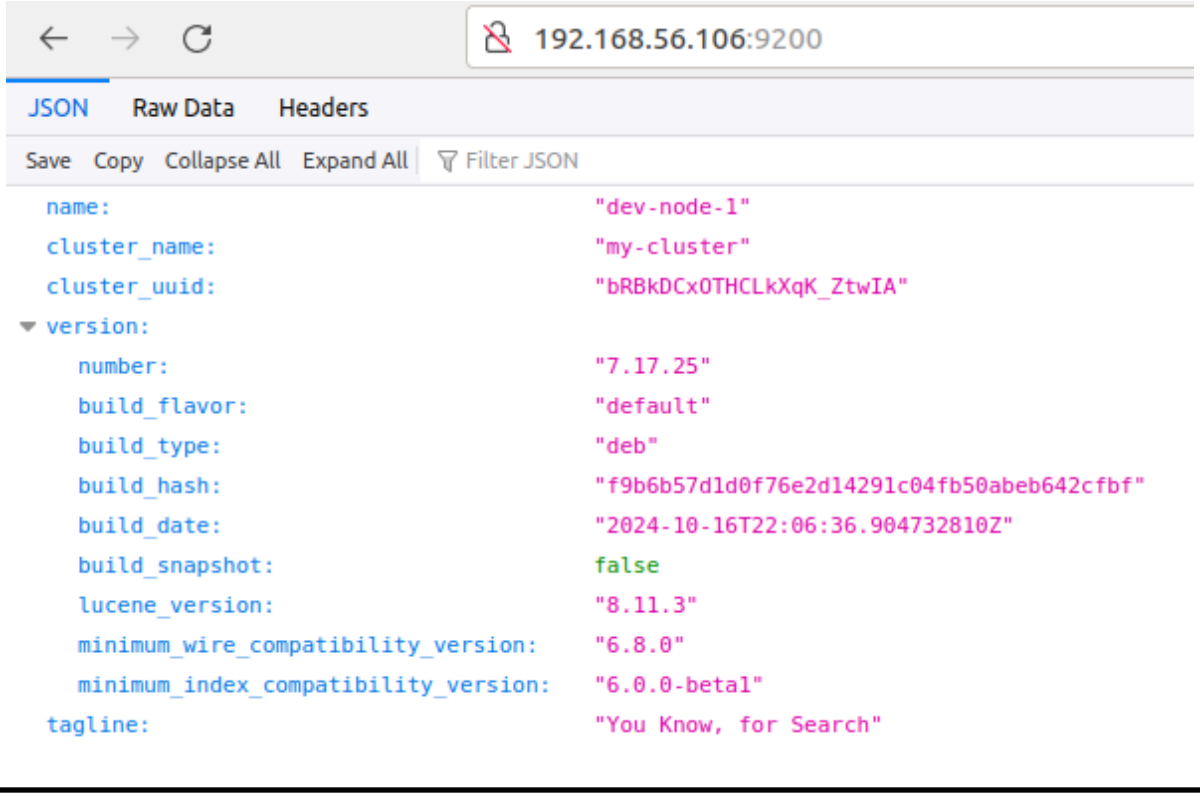
```
- name: Allowing port 9200 through the firewall
  tags: elasticsearch
  command: firewall-cmd --zone=public --add-port=9200/tcp --permanent
  register: firewall_result
  ignore_errors: true
  when: ansible_distribution == "CentOS"
```

roles/elasticsearch/tasks/elasticsearch.yml.j2 (CentOS)

```
# 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
```

Elasticsearch Proof (Ubuntu)

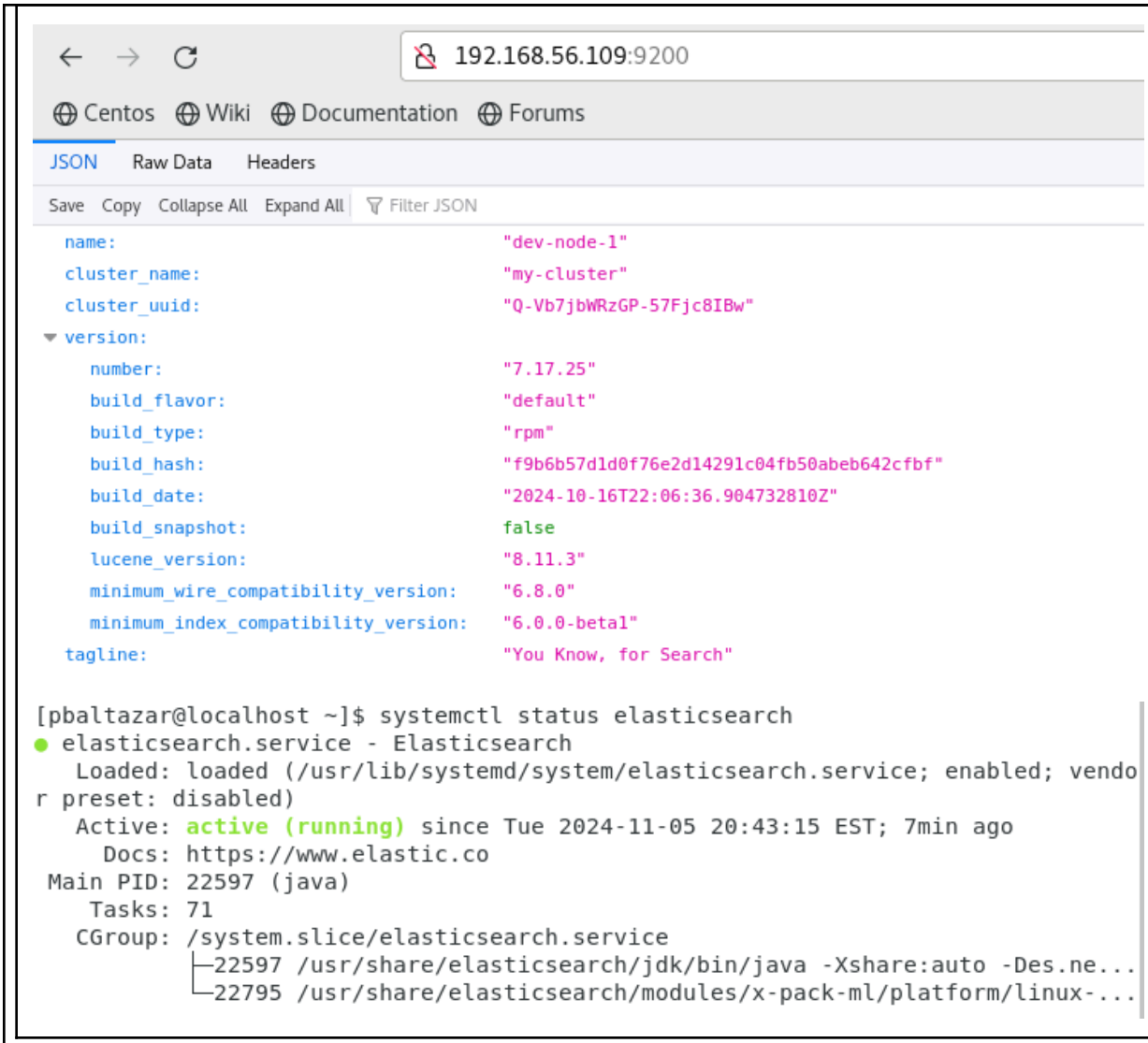
```
TASK [elasticsearchubuntu : Installing Java] ****************
skipping: [192.168.56.109]
ok: [192.168.56.106]

TASK [elasticsearchubuntu : Elastic Search Installation] *****
skipping: [192.168.56.109]
ok: [192.168.56.106]

TASK [elasticsearchubuntu : Configuring Elastic Search] ******
skipping: [192.168.56.109]
ok: [192.168.56.106]

TASK [elasticsearchubuntu : Starting Elastic Search] *********
skipping: [192.168.56.109]
changed: [192.168.56.106]
```



```
paul_eimar@Server1:~$ systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendo
   Active: active (running) since Wed 2024-11-06 10:10:54 +08; 1min 56s ago
     Docs: https://www.elastic.co
 Main PID: 7633 (java)
    Tasks: 69 (limit: 4915)
   CGroup: /system.slice/elasticsearch.service
           ├─7633 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networ
           └─7852 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_
```

Elasticsearch Proof (CentOS)

```
PLAY [Install Elasticsearch] ***************************************

TASK [Gathering Facts] *********************************************
ok: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Installing Java] *****************************
skipping: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Installing EPEL repository] ******************
skipping: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Installing Elastic Search YUM repository] ****
skipping: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Elastic Search Installation] *****************
skipping: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Configuring Elastic Search] ******************
skipping: [192.168.56.106]
ok: [192.168.56.109]

TASK [elasticsearch : Starting Elastic Search] *********************
skipping: [192.168.56.106]
changed: [192.168.56.109]

TASK [elasticsearch : Allowing port 9200 through the firewall] *****
skipping: [192.168.56.106]
changed: [192.168.56.109]
```


192.168.56.109:9200
Centos
Wiki
Documentation
Forums
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON
name: "dev-node-1"
cluster_name: "my-cluster"
cluster_uuid: "Q-Vb7jbWRzGP-57Fjc8IBw"
version:
number: "7.17.25"
build_flavor: "default"
build_type: "rpm"
build_hash: "f9b6b57d1d0f76e2d14291c04fb50abeb642cfbf"
build_date: "2024-10-16T22:06:36.904732810Z"
build_snapshot: false
lucene_version: "8.11.3"
minimum_wire_compatibility_version: "6.8.0"
minimum_index_compatibility_version: "6.0.0-beta1"
tagline: "You Know, for Search"
[pbaltazar@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 20:43:15 EST; 7min ago
Docs: https://www.elastic.co
Main PID: 22597 (java)
Tasks: 71
CGroup: /system.slice/elasticsearch.service
├─22597 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.ne...
└─22795 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-...

roles/kibana/tasks/main.yml (Ubuntu)

```
- name: Adding GPG key for Elastic APT repository
  tags: kibana
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present
  when: ansible_distribution == "Ubuntu"

- name: Adding Kibana APT repository
  tags: kibana
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/7.x/apt stable main"
    state: present
  when: ansible_distribution == "Ubuntu"

- name: Installing of Kibana
  tags: kibana
  apt:
    name: kibana
    state: present
  when: ansible_distribution == "Ubuntu"
```

```
- name: Creating directory for Kibana systemd override
  tags: kibana
  file:
    path: /etc/systemd/system/kibana.service.d
    state: directory
    mode: '0755'
    owner: root
    group: root
  when: ansible_distribution == "Ubuntu"

- name: Checking if the directory was created
  tags: kibana
  stat:
    path: /etc/systemd/system/kibana.service.d
  register: kibana_override_dir

- debug:
    msg: "Directory already exists: {{ kibana_override_dir.stat.exists }}"
```

```
- name: Creating Kibana service override configuration
  tags: kibana
  file:
    path: /etc/systemd/system/kibana.service.d/override.conf
    state: touch  # Ensures the file exists
    owner: root
    group: root
    mode: '0644'
  when: ansible_distribution == "Ubuntu"

- name: Configure Kibana (Setting OpenSSL Legacy Provider)
  tags: kibana
  blockinfile:
    path: /etc/systemd/system/kibana.service.d/override.conf
    block: |
      [Service]
      Environment=NODE_OPTIONS=--openssl-legacy-provider
    owner: root
    group: root
    mode: '0644'
  when: ansible_distribution == "Ubuntu"

- name: Configuring Kibana
  tags: kibana
  template:
    src: kibana.yml.j2
    dest: /etc/kibana/kibana.yml
  when: ansible_distribution == "Ubuntu"
```

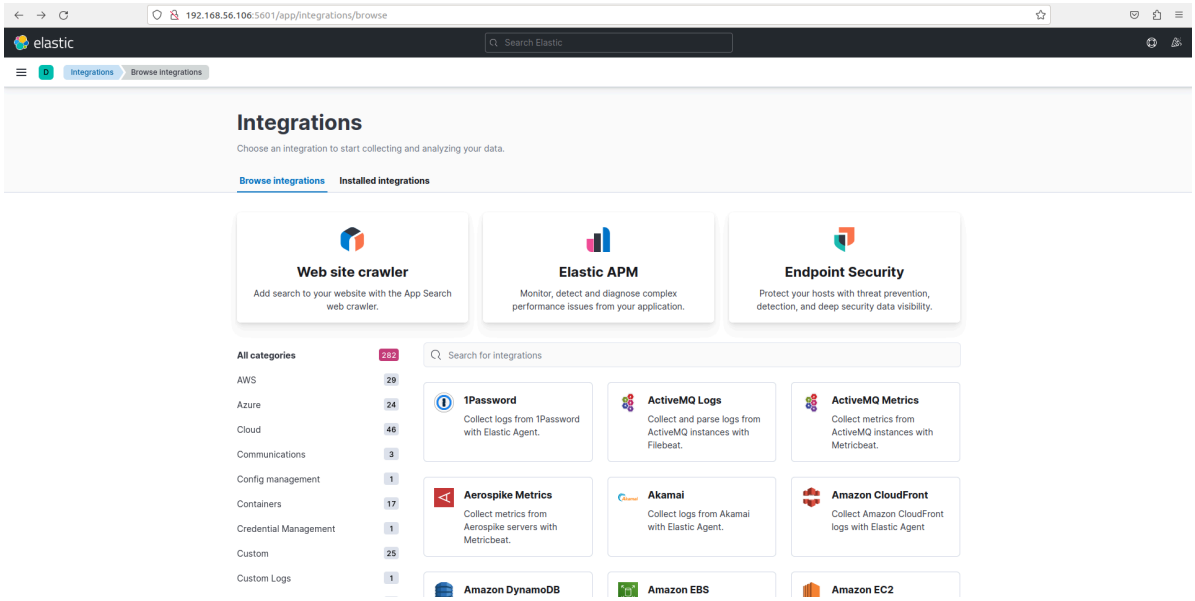
```
- name: Reloading systemd
  tags: kibana
  command: systemctl daemon-reload
  when: ansible_distribution == "Ubuntu"

- name: Enabling Kibana service
  tags: kibana
  service:
    name: kibana
    state: restarted
  become: yes
  when: ansible_distribution == "Ubuntu"
```

roles/kibana/tasks/kibana.yml.j2 (Ubuntu)

```
# Kibana Configuration

# Set the port that the Kibana server will listen on
server.port: 5601

# Specify the host address that the Kibana server will bind to
server.host: "192.168.56.106"

# Set the public base URL for Kibana
server.publicBaseUrl: "http://192.168.56.106:5601"

# Elasticsearch server URL
elasticsearch.hosts: ["http://192.168.56.106:9200"]
elasticsearch.hosts: ["http://192.168.56.109:9200"]
```

Kibana Proof (Ubuntu)

```
PLAY [Install Kibana] **********************************************

TASK [Gathering Facts] *********************************************
ok: [192.168.56.106]

TASK [kibana : Adding GPG key for Elastic APT repository] **********
ok: [192.168.56.106]

TASK [kibana : Adding Kibana APT repository] ***********************
ok: [192.168.56.106]

TASK [kibana : Installing of Kibana] *******************************
ok: [192.168.56.106]

TASK [kibana : Creating directory for Kibana systemd override] *****
ok: [192.168.56.106]

TASK [kibana : Checking if the directory was created] **************
ok: [192.168.56.106]

TASK [kibana : debug] **********************************************
ok: [192.168.56.106] => {
    "msg": "Directory already exists: True"
}

TASK [kibana : Creating Kibana service override configuration] *****
changed: [192.168.56.106]

TASK [kibana : Configure Kibana (Setting OpenSSL Legacy Provider)] *
ok: [192.168.56.106]

TASK [kibana : Configuring Kibana] *********************************
changed: [192.168.56.106]

TASK [kibana : Reloading systemd] **********************************
changed: [192.168.56.106]

TASK [kibana : Enabling Kibana service] ****************************
changed: [192.168.56.106]
```




```
paul_eimar@Server1:~$ systemctl status kibana
● kibana.service - Kibana
     Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset: e
    Drop-In: /etc/systemd/system/kibana.service.d
             └─override.conf
     Active: active (running) since Wed 2024-11-06 09:58:39 +08; 7min ago
       Docs: https://www.elastic.co
   Main PID: 5262 (node)
      Tasks: 11 (limit: 4915)
     CGroup: /system.slice/kibana.service
             └─5262 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/.
```

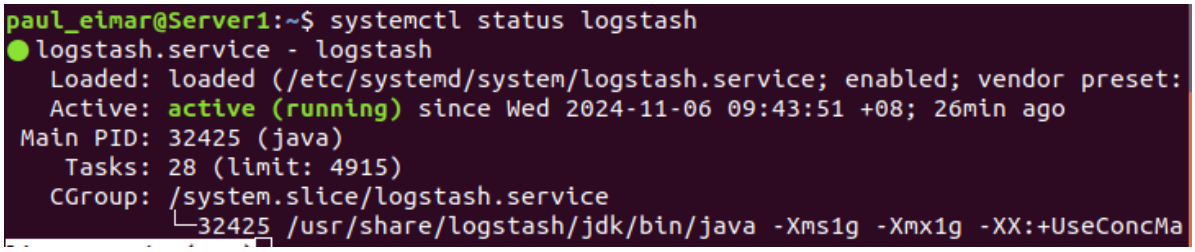
roles/logstash/task/main.yml (Ubuntu)

```
- name: Installing dependencies
  tags: logstash
  apt:
    name: gnupg
    state: present
    update_cache: yes
  become: yes

- name: Adding Elastic APT repository key
  tags: logstash
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present

- name: Adding Elastic APT repository
  tags: logstash
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/7.x/apt stable main"
    state: present

- name: Installing Logstash
  tags: logstash
  apt:
    name: logstash
    state: present

- name: Starting and Enabling Logstash service
  tags: logstash
  systemd:
    name: logstash
    enabled: yes
    state: started
```

roles/logstash/task/logstash.conf.j2 (Ubuntu)

```
nput {
  beats {
    port => 5044
  }
}

filter {
  # Add any filters here
}

output {
  elasticsearch {
    hosts => ["http://192.168.56.109:9200"]
    index => "logstash-%{+YYYY.MM.dd}"
  }
}
```

Logstash proof (Ubuntu)



roles/nagios/task/main.yml (Ubuntu)

```
---
  - name: Install required dependencies on Ubuntu
    tags: nagios
    apt:
      name:
        - gcc
        - libc6
        - make
        - wget
        - unzip
        - apache2
        - php
        - libgd-dev
        - openssl
        - libssl-dev
        - autoconf
        - bc
        - gawk
        - dc
        - build-essential
        - snmp
        - libnet-snmp-perl
        - gettext
      state: present
    when: ansible_distribution == "Ubuntu"

  - name: Download Nagios Core source code
    tags: nagios
    get_url:
      url: "https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.6.tar.gz"
      dest: /tmp/nagios-4.5.6.tar.gz

  - name: Extract Nagios source code
    tags: nagios
    unarchive:
      src: /tmp/nagios-4.5.6.tar.gz
      dest: /tmp
      remote_src: yes

  - name: Download Nagios Plugins
    tags: nagios
    get_url:
      url: "https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz"
      dest: /tmp/nagios-plugins-2.4.11.tar.gz
```

```
  - name: Extract Nagios Plugins
    tags: nagios
    unarchive:
      src: /tmp/nagios-plugins-2.4.11.tar.gz
      dest: /tmp
      remote_src: yes

  - name: Create Nagios group
    tags: nagios
    group:
      name: nagios

  - name: Create Nagios user and group
    tags: nagios
    user:
      name: nagios
      group: nagios

  - name: Create nagcmd group
    tags: nagios
    group:
      name: nagcmd
```

```
- name: Add nagios and apache/httpd users to nagcmd group
  tags: nagios
  user:
    name: "{{ item }}"
    groups: nagcmd
    append: yes
  loop:
    - nagios
    - "{{ 'www-data' if ansible_os_family == 'Debian' else 'apache' }}"

- name: Compile and install Nagios Core
  tags: nagios
  shell: |
    cd /tmp/nagios-4.5.6
    ./configure --with-command-group=nagcmd
    make all
    make install
    make install-init
    make install-commandmode
    make install-config
    make install-webconf
  args:
    creates: /usr/local/nagios/bin/nagios
```

```
- name: Install Nagios Plugins
  tags: nagios
  shell: |
    cd /tmp/nagios-plugins-2.4.11
    ./configure --with-nagios-user=nagios --with-nagios-group=nagios
    make
    make install
  args:
    creates: /usr/local/nagios/libexec/check_http

- name: Set Nagios admin password
  tags: nagios
  command: htpasswd -b -c /usr/local/nagios/etc/htpasswd.users paul_admin "sample"

- name: Enable and start Apache/Httpd service on Ubuntu
  tags: nagios
  service:
    name: apache2
    enabled: yes
    state: started
  when: ansible_distribution == "Ubuntu"
```

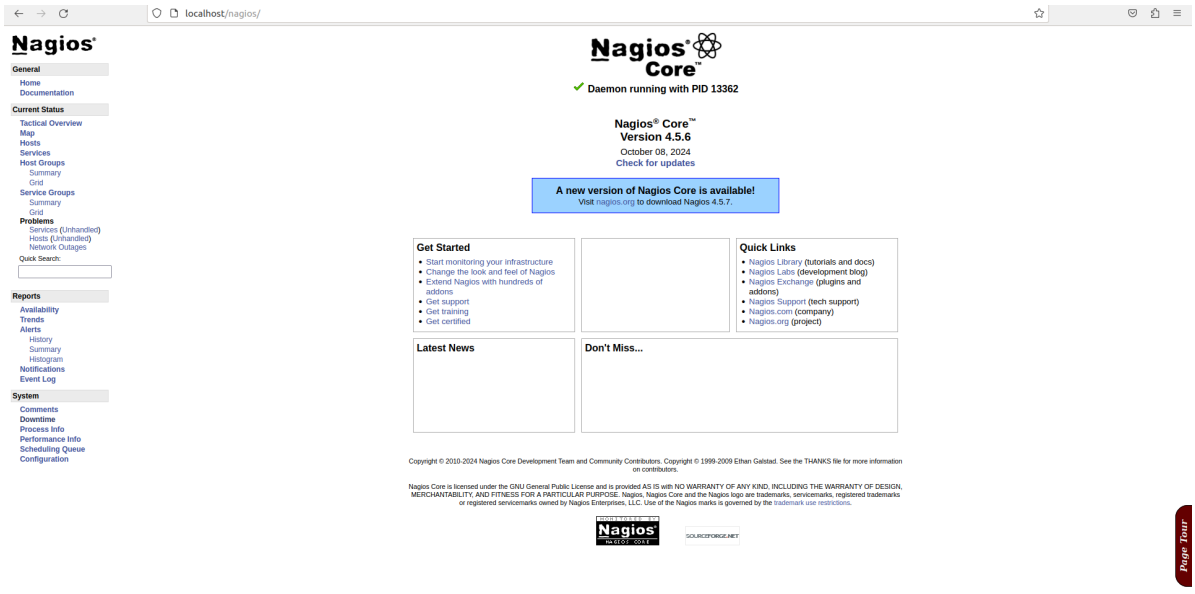
```
- name: Enable and start Apache/Httpd service on Ubuntu
  tags: nagios
  service:
    name: apache2
    enabled: yes
    state: started
  when: ansible_distribution == "Ubuntu"


- name: Enable and start Nagios service
  tags: nagios
  service:
    name: nagios
    enabled: yes
    state: started

- name: Enable external command execution in Nagios
  tags: nagios
  lineinfile:
    path: /usr/local/nagios/etc/nagios.cfg
    regexp: '^#?check_external_commands='
    line: 'check_external_commands=1'

- name: Restart Nagios service to apply changes
  tags: nagios
  service:
    name: nagios
    state: restarted

- name: Restart Apache/Httpd to apply changes on Ubuntu
  tags: nagios
  service:
    name: apache2
    state: restarted
  when: ansible_distribution == "Ubuntu"
```

Nagios Proof

```
PLAY [Install Nagios] ***********************************************

TASK [Gathering Facts] ***********************************************
ok: [192.168.56.106]

TASK [nagios : Install required dependencies on Ubuntu] *************
ok: [192.168.56.106]

TASK [nagios : Download Nagios Core source code] ********************
ok: [192.168.56.106]

TASK [nagios : Extract Nagios source code] ***************************
ok: [192.168.56.106]

TASK [nagios : Download Nagios Plugins] ******************************
ok: [192.168.56.106]

TASK [nagios : Extract Nagios Plugins] *******************************
ok: [192.168.56.106]

TASK [nagios : Create Nagios group] **********************************
ok: [192.168.56.106]

TASK [nagios : Create Nagios user and group] *************************
ok: [192.168.56.106]

TASK [nagios : Create nagcmd group] **********************************
ok: [192.168.56.106]

TASK [nagios : Add nagios and apache/httpd users to nagcmd group] ***
ok: [192.168.56.106] => (item=nagios)
ok: [192.168.56.106] => (item=www-data)

TASK [nagios : Compile and install Nagios Core] **********************
ok: [192.168.56.106]

TASK [nagios : Install Nagios Plugins] *******************************
ok: [192.168.56.106]

TASK [nagios : Set Nagios admin password] ****************************
changed: [192.168.56.106]

TASK [nagios : Enable and start Apache/Httpd service on Ubuntu] *****
ok: [192.168.56.106]
```

```
TASK [nagios : Extract Nagios source code] ************************
ok: [192.168.56.106]

TASK [nagios : Download Nagios Plugins] ***************************
ok: [192.168.56.106]

TASK [nagios : Extract Nagios Plugins] ****************************
ok: [192.168.56.106]

TASK [nagios : Create Nagios group] *******************************
ok: [192.168.56.106]

TASK [nagios : Create Nagios user and group] **********************
ok: [192.168.56.106]

TASK [nagios : Create nagcmd group] *******************************
ok: [192.168.56.106]

TASK [nagios : Add nagios and apache/httpd users to nagcmd group] **
ok: [192.168.56.106] => (item=nagios)
ok: [192.168.56.106] => (item=www-data)

TASK [nagios : Compile and install Nagios Core] *******************
ok: [192.168.56.106]

TASK [nagios : Install Nagios Plugins] ****************************
ok: [192.168.56.106]

TASK [nagios : Set Nagios admin password] *************************
changed: [192.168.56.106]

TASK [nagios : Enable and start Apache/Httpd service on Ubuntu] ****
ok: [192.168.56.106]

TASK [nagios : Enable and start Nagios service] *******************
ok: [192.168.56.106]

TASK [nagios : Enable external command execution in Nagios] ********
ok: [192.168.56.106]

TASK [nagios : Restart Nagios service to apply changes] ************
changed: [192.168.56.106]

TASK [nagios : Restart Apache/Httpd to apply changes on Ubuntu] ****
changed: [192.168.56.106]
```

```
paul_eimar@Server1:~$ systemctl status nagios
● nagios.service - Nagios Core 4.5.6
     Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: e
     Active: active (running) since Wed 2024-11-06 09:58:58 +08; 38s ago
       Docs: https://www.nagios.org/documentation
    Process: 6834 ExecStopPost=/bin/rm -f /usr/local/nagios/var/rw/nagios.cmd (cod
    Process: 6833 ExecStop=/bin/kill -s TERM ${MAINPID} (code=exited, status=0/SUC
    Process: 6836 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/
    Process: 6835 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/e
   Main PID: 6837 (nagios)
      Tasks: 11 (limit: 4915)
     CGroup: /system.slice/nagios.service
             ├─6837 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.c
             ├─6838 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6839 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6840 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6841 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6842 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6843 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6844 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6845 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             ├─6846 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
             └─6848 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.c
```



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

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

MariaDB (Ubuntu)

GitHub link: https://github.com/rpldpaul/CPE_MIDEXAM_Baltazar
Conclusions: (link your conclusion from the objective) In this exam, we have