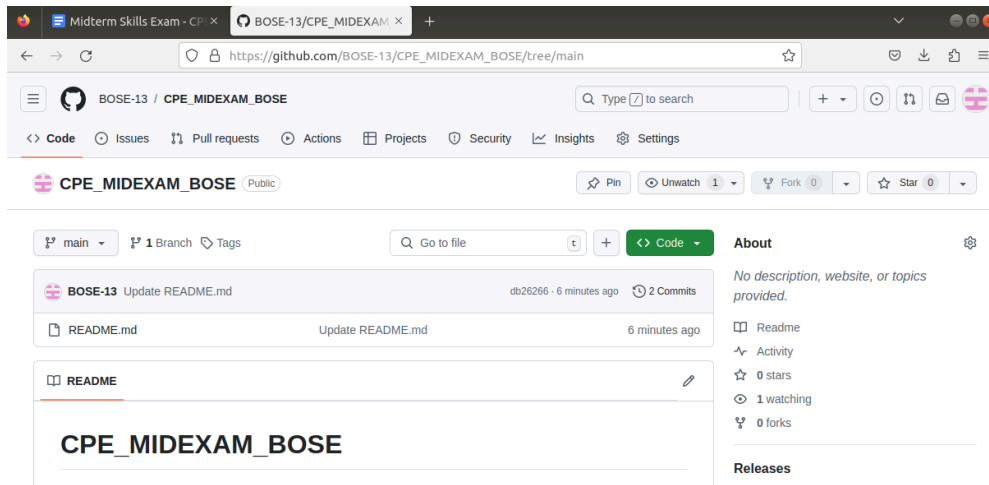


<b>Name: Vincent Bryan Bose</b>	<b>Date Performed: 11/06/2024</b>
<b>Course/Section: CpE31S2</b>	<b>Date Submitted: 11/06/2024</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY: 2024-2025</b>
<b>Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools</b>	
<b>1. Objectives</b>	
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.	
<b>2. Instructions</b>	
<ol style="list-style-type: none"> <li>1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.</li> <li>2. Clone the repository and do the following: <ol style="list-style-type: none"> <li>2.1. Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:</li> <li>2.2. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash) • Install Nagios in one host</li> <li>2.3. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)</li> <li>2.4. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)</li> </ol> </li> <li>3. Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.</li> <li>4. Document the push and commit from the local repository to GitHub.</li> <li>5. Finally, paste also the link of your GitHub repository in the documentation.</li> </ol>	

### 3. Output (screenshots and explanations)

1. Create a repository in your GitHub account and label it CPE\_MIDEXAM\_SURNAME.



2. Clone the repository and do the following:

```
File Edit View Search Terminal Help
vbbose@workstation:~$ git clone git@github.com:BOSE-13/CPE_MIDEXAM_BOSE.git
Cloning into 'CPE_MIDEXAM_BOSE'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (6/6), done.
```

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

```
GNU nano 2.9.3 inventory
[Ubuntu]
192.168.56.106
[CentOS]
192.168.56.113 ansible_user=vbbose
```

- Edit inventory using the ip address of your server and centos.

```
File Edit View Search Terminal Help
GNU nano 2.9.3 ansible.cfg
[defaults]
inventory = inventory
remote_user = vbbose
host_key_checking = True
```

- Contents of ansible.cfg

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

```
└─ elasticsearch
    └─ tasks
        └─ elasticsearch.yml.j2
        └─ main.yml
```

Open ▾



elasticsearch.yml.j2

~/CPE\_MIDEXAM\_BOSE/roles/elasticsearch/tasks

# 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
```

Open ▾



main.yml

~/CPE\_MIDEXAM\_BOSE/roles/elasticsearch/tasks

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

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

- name: Install 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: Install Elastic Search
  tags: elasticsearch
  dnf:
    name: elasticsearch
    state: present
  when: ansible_distribution == "CentOS"

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

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

-
- name: Allow 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"
```

```

TASK [elasticsearch : Install Elastic Search YUM repository] *****
ok: [192.168.56.113]

TASK [elasticsearch : Install Elastic Search] *****
ok: [192.168.56.113]

TASK [elasticsearch : Configure Elastic Search] *****
ok: [192.168.56.113]

TASK [elasticsearch : Start Elastic Search] *****
changed: [192.168.56.113]

TASK [elasticsearch : Allow port 9200 through the firewall] *****
changed: [192.168.56.113]

```

## PROOF CENTOS:

BOSE CENTOS [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Firefox Nov 6 10:02

Restore Session × 192.168.56.113:9200/ × +

← → ↻ 192.168.56.113:9200

CentOS Blog Documentation Forums

JSON Raw Data Headers

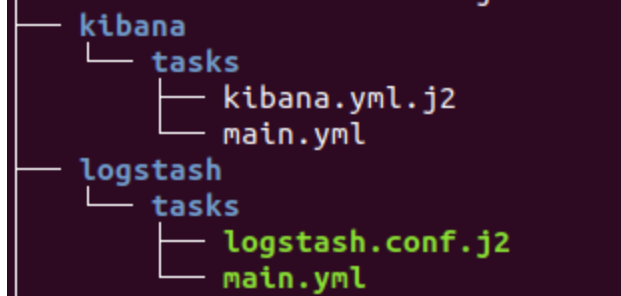
Save Copy Collapse All Expand All Filter JSON

```

name: "dev-node-1"
cluster_name: "my-cluster"
cluster_uuid: "nRwFKHz3Qb0xZDN0fskuCw"
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"

```

- The contents of elastic search



Open  kibana.yml.j2  
~/CPE\_MIDEXAM\_BOSE/roles/kibana/tasks

# 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://192.168.56.113:9200"]

Open ▾



main.yml

~/CPE\_MIDEXAM\_BOSE/roles/kibana/tasks

```
---
- name: Add 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: Add 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: Install specific version of Kibana
  tags: kibana
  apt:
    name: kibana
    state: present
  when: ansible_distribution == "Ubuntu"

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

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

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

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

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



```

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

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

TASK [kibana : Install specific version of Kibana] *****
ok: [192.168.56.106]

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

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

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

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

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

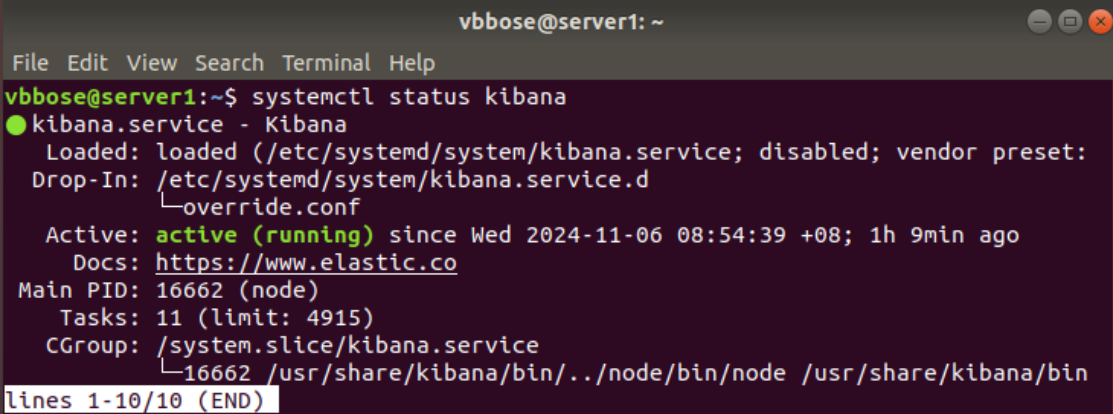
TASK [kibana : Configure Kibana] *****
ok: [192.168.56.106]

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

TASK [kibana : Enable Kibana service] *****
changed: [192.168.56.106]

```

PROOF:




```

vbbose@server1: ~
File Edit View Search Terminal Help
vbbose@server1:~$ systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; disabled; vendor preset:
   Drop-In: /etc/systemd/system/kibana.service.d
            └─override.conf
   Active: active (running) since Wed 2024-11-06 08:54:39 +08; 1h 9min ago
     Docs: https://www.elastic.co
    Main PID: 16662 (node)
      Tasks: 11 (limit: 4915)
    CGroup: /system.slice/kibana.service
            └─16662 /usr/share/kibana/bin/./node/bin/node /usr/share/kibana/bin
lines 1-10/10 (END)

```


- Installed kibana

Open  **logstash.conf.j2**  
~/CPE\_MIDEXAM\_BOSE/roles/logstash/tasks

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

filter {
  # Add any filters here
}

output {
  elasticsearch {
    hosts => ["http://192.168.56.104:9200"]
    index => "logstash-%{+YYYY.MM.dd}"
  }
}
```

Open  **main.yml**  
~/CPE\_MIDEXAM\_BOSE/roles/logstash/tasks

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

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

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

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

- name: Start and Enable Logstash service
  tags: logstash
  systemd:
    name: logstash
    enabled: yes
    state: started
```

```

TASK [logstash : Install dependencies] *****
ok: [192.168.56.106]

TASK [logstash : Add Elastic APT repository key] *****
ok: [192.168.56.106]

TASK [logstash : Add Elastic APT repository] *****
ok: [192.168.56.106]

TASK [logstash : Install Logstash] *****
ok: [192.168.56.106]

TASK [logstash : Start and Enable Logstash service] *****
ok: [192.168.56.106]

```

PROOF:

```

vbbose@server1: ~
File Edit View Search Terminal Help

vbbose@server1:~$ systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; disabled; vendor preset:
   Drop-In: /etc/systemd/system/kibana.service.d
            └─override.conf
   Active: active (running) since Wed 2024-11-06 08:54:39 +08; 1h 9min ago
     Docs: https://www.elastic.co
  Main PID: 16662 (node)
    Tasks: 11 (limit: 4915)
   CGroup: /system.slice/kibana.service
           └─16662 /usr/share/kibana/bin/./node/bin/node /usr/share/kibana/bin

vbbose@server1:~$ systemctl status logstash
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor preset
   Active: activating (auto-restart) (Result: exit-code) since Wed 2024-11-06 1
   Process: 30681 ExecStart=/usr/share/logstash/bin/logstash --path.settings /et
   Main PID: 30681 (code=exited, status=1/FAILURE)
lines 1-5/5 (END)

```

- Installed logstash

```

├─ nagios
│   └─ tasks
│       └─ main.yml

```

Open ▾



main.yml

~/CPE\_MIDEXAM\_BOSE/roles/nagios/tasks

```
---
- 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 vbbose "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 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"
```

```

PLAY [Install Nagios in Ubuntu] *****

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] *****
changed: [192.168.56.106]

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

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

TASK [nagios : Extract Nagios Plugins] *****
changed: [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 [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]

```

PROOF:

```

vbb@server1:~$ systemctl status nagios
● nagios.service - Nagios Core 4.5.6
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-11-06 08:55:15 +08; 1h 16min ago
     Docs: https://www.nagios.org/documentation
   Process: 18610 ExecStopPost=/bin/rm -f /usr/local/nagios/var/rw/nagios.cmd (c
   Process: 18609 ExecStop=/bin/kill -s TERM ${MAINPID} (code=exited, status=0/S
   Process: 18612 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/et
   Process: 18611 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios
   Main PID: 18613 (nagios)
      Tasks: 8 (limit: 4915)
     CGroup: /system.slice/nagios.service
             └─18613 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios
                └─18614 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                   └─18615 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                      └─18616 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                         └─18617 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                            └─18618 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                               └─18619 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/
                                  └─18688 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios
lines 1-19/19 (END)

```

## 2.3 Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)

```

└─ grafana
   └─ tasks
      └─ main.yml
         └─ templates
            └─ grafana.ini.j2

```

```

└─ ubuntu-prometheus
   └─ tasks
      └─ main.yml

```

```

Open ▾  main.yml
~/CPE_MIDEXAM_B05E/roles/ubuntu-prometheus/tasks
---
- name: Install Prometheus on Ubuntu
  tags: prometheus
  apt:
    name: prometheus
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: Start Prometheus Service (Ubuntu)
  tags: prometheus
  systemd:
    name: prometheus
    enabled: yes
    state: started
  when: ansible_distribution == "Ubuntu"


```

```

└─ centos-prometheus
   └─ tasks
      └─ main.yml

```



Open  main.yml  
~/CPE\_MIDEXAM\_B05E/roles/centosprometheus/tasks

```
---  
- name: Install Prometheus (CentOS)  
  tags: prometheus  
  unarchive:  
    src: https://github.com/prometheus/prometheus/releases/download/v2.30.0/prometheus-2.30.0.linux-amd64.tar.gz  
    dest: /usr/local/bin  
    remote_src: yes  
    mode: 0755  
    owner: root  
    group: root  
  when: ansible_distribution == "CentOS"  
  
- name: Copy Prometheus binaries  
  tags: prometheus  
  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  
  tags: prometheus  
  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  
  tags: prometheus  
  file:  
    path: "{{ item }}"  
    state: directory  
  loop:  
    - /etc/prometheus  
    - /var/lib/prometheus  
  when: ansible_distribution == "CentOS"  
  
- name: Copy prometheus.yml to /etc/prometheus  
  tags: 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  
  tags: prometheus  
  command: cp -r /usr/local/bin/prometheus-2.30.0.linux-amd64/consoles /etc/prometheus  
  when: ansible_distribution == "CentOS"
```

```

- name: Copy prometheus.yml to /etc/prometheus
  tags: 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
  tags: 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
  tags: 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
  tags: prometheus
  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
  when: ansible_distribution == "CentOS"

- name: Reload systemd
  tags: prometheus
  command: systemctl daemon-reload
  when: ansible_distribution == "CentOS"

- name: Reload systemd
  tags: prometheus
  command: systemctl daemon-reload
  when: ansible_distribution == "CentOS"

- name: Start Prometheus Service
  tags: prometheus
  systemd:
    name: prometheus
    enabled: yes
    state: started
  when: ansible_distribution == "CentOS"

```

```

PLAY [Install Prometheus] *****

TASK [Gathering Facts] *****
ok: [192.168.56.106]
ok: [192.168.56.113]

TASK [ubuntuprometheus : Install Prometheus on Ubuntu] *****
skipping: [192.168.56.113]
ok: [192.168.56.106]

TASK [ubuntuprometheus : Start Prometheus Service (Ubuntu)] *****
skipping: [192.168.56.113]
ok: [192.168.56.106]

TASK [centosprometheus : Install Prometheus (CentOS)] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Copy Prometheus binaries] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Copy Promtool binaries] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Create Prometheus directories] *****
skipping: [192.168.56.106] => (item=/etc/prometheus)
skipping: [192.168.56.106] => (item=/var/lib/prometheus)
changed: [192.168.56.113] => (item=/etc/prometheus)
changed: [192.168.56.113] => (item=/var/lib/prometheus)

TASK [centosprometheus : Copy prometheus.yml to /etc/prometheus] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Copy consoles directory to /etc/prometheus] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Copy console_libraries directory to /etc/prometheus] ***
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Create prometheus.service file] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

TASK [centosprometheus : Reload systemd] *****
skipping: [192.168.56.106]
changed: [192.168.56.113]

```

## 2.4 Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)

```

└─ apache
   └─ tasks
      └─ main.yml

```

## main.yml

```
---
- name: Install Apache in Ubuntu (httpd)
  tags: apache2
  become: yes
  apt:
    name: apache2
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: Install Apache in CentOS (httpd)
  tags: apache2
  become: yes
  dnf:
    name: httpd
    state: latest
  when: ansible_distribution == "CentOS"

- name: start httpd (CentOS)
  tags: apache2
  service:
    name: httpd
    state: started
  when: ansible_distribution == "CentOS"

- name: Allow port 9200 through the firewall
  tags: apache2
  command: firewall-cmd --add-port=80/tcp --permanent
  register: firewall_result
  ignore_errors: true
  when: ansible_distribution == "CentOS"
```

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

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

- We need to install apache so that we can process requests and serves web assets and content via HTTP.

```
vbbose@BOSECENTOS:~ — systemctl status httpd
• httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
  Drop-In: /usr/lib/systemd/system/httpd.service.d
           └─php-fpm.conf
  Active: active (running) since Wed 2024-11-06 07:22:29 PST; 2h 54min ago
  Docs: man:httpd.service(8)
  Main PID: 863 (httpd)
  Status: "Total requests: 35; Idle/Busy workers 100/0;Requests/sec: 0.00334; Bytes served/sec: 1 B/se
  Tasks: 230 (limit: 10950)
  Memory: 12.3M
  CPU: 3.722s
  CGroup: /system.slice/httpd.service
          └─ 863 /usr/sbin/httpd -DFOREGROUND
             950 /usr/sbin/httpd -DFOREGROUND
             951 /usr/sbin/httpd -DFOREGROUND
             952 /usr/sbin/httpd -DFOREGROUND
             953 /usr/sbin/httpd -DFOREGROUND
            3143 /usr/sbin/httpd -DFOREGROUND

Nov 06 07:22:22 BOSECENTOS systemd[1]: Starting The Apache HTTP Server...
Nov 06 07:22:29 BOSECENTOS httpd[863]: AH00558: httpd: Could not reliably determine the server's fully qual
Nov 06 07:22:29 BOSECENTOS systemd[1]: Started The Apache HTTP Server.
Nov 06 07:22:29 BOSECENTOS httpd[863]: Server configured, listening on: port 80
~
~
~
~
~
```

```
Open main.yml ~/CPE_MIDEXAM_BOSE/roles/mariadb/tasks
main.yml x
---
- name: Install MariaDB
  tags: mariadb
  become: yes
  apt:
    name: mariadb-server
    state: present
  when: ansible_distribution == "Ubuntu"

- name: Install MariaDB
  tags: mariadb
  become: yes
  yum:
    name: mariadb-server
    state: present
  when: ansible_distribution == "CentOS"
```

PROOF:

```

vbbose@server1:~$ systemctl status mariadb
● mariadb.service - MariaDB 10.1.48 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-11-06 07:22:15 +08; 3h 1min ago
     Docs: man:mysql(8)
           https://mariadb.com/kb/en/library/systemd/
   Main PID: 1082 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 27 (limit: 4915)
    CGroup: /system.slice/mariadb.service
            └─1082 /usr/sbin/mysqld

Warning: Journal has been rotated since unit was started. Log output is incompl

```

4.

```

vbbose@workstation:~/CPE_MIDEXAM_B0SE$ git add *
vbbose@workstation:~/CPE_MIDEXAM_B0SE$ git commit -m "DONE"
[main 9065b65] DONE
18 files changed, 642 insertions(+), 1 deletion(-)
create mode 100644 ansible.cfg
create mode 100644 inventory
create mode 100644 midterm.yml
create mode 100644 roles/apache/tasks/main.yml
create mode 100644 roles/centosprometheus/tasks/main.yml
create mode 100644 roles/elasticsearch/tasks/elasticsearch.yml.j2
create mode 100644 roles/elasticsearch/tasks/main.yml
create mode 100644 roles/grafana/tasks/main.yml
create mode 100644 roles/grafana/tasks/templates/grafana.ini.j2
create mode 100644 roles/kibana/tasks/kibana.yml.j2
create mode 100644 roles/kibana/tasks/main.yml
create mode 100755 roles/logstash/tasks/logstash.conf.j2
create mode 100755 roles/logstash/tasks/main.yml
create mode 100644 roles/mariadb/tasks/main.yml
create mode 100644 roles/nagios/tasks/main.yml
create mode 100644 roles/php/tasks/main.yml
create mode 100644 roles/ubuntuprometheus/tasks/main.yml
vbbose@workstation:~/CPE_MIDEXAM_B0SE$ git push
Counting objects: 42, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (23/23), done.
Writing objects: 100% (42/42), 6.63 KiB | 6.63 MiB/s, done.
Total 42 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To github.com:BOSE-13/CPE_MIDEXAM_B0SE.git
   4907642..9065b65  main -> main

```

5. [git@github.com:BOSE-13/CPE\\_MIDEXAM\\_B0SE.git](https://github.com/BOSE-13/CPE_MIDEXAM_B0SE.git)

**GitHub link:**

[git@github.com:BOSE-13/CPE\\_MIDEXAM\\_B0SE.git](https://github.com/BOSE-13/CPE_MIDEXAM_B0SE.git)

**Conclusions:** (link your conclusion from the objective)

