Name: Jaira Biane Maculada	**Date Performed:**11/06/23
Course/Section:CPE232/CPE31S6	**Date Submitted:**11/06/23
Instructor: Dr. Jonathan V. Taylar	**Semester and SY:** 1st Sem(2023-2024)

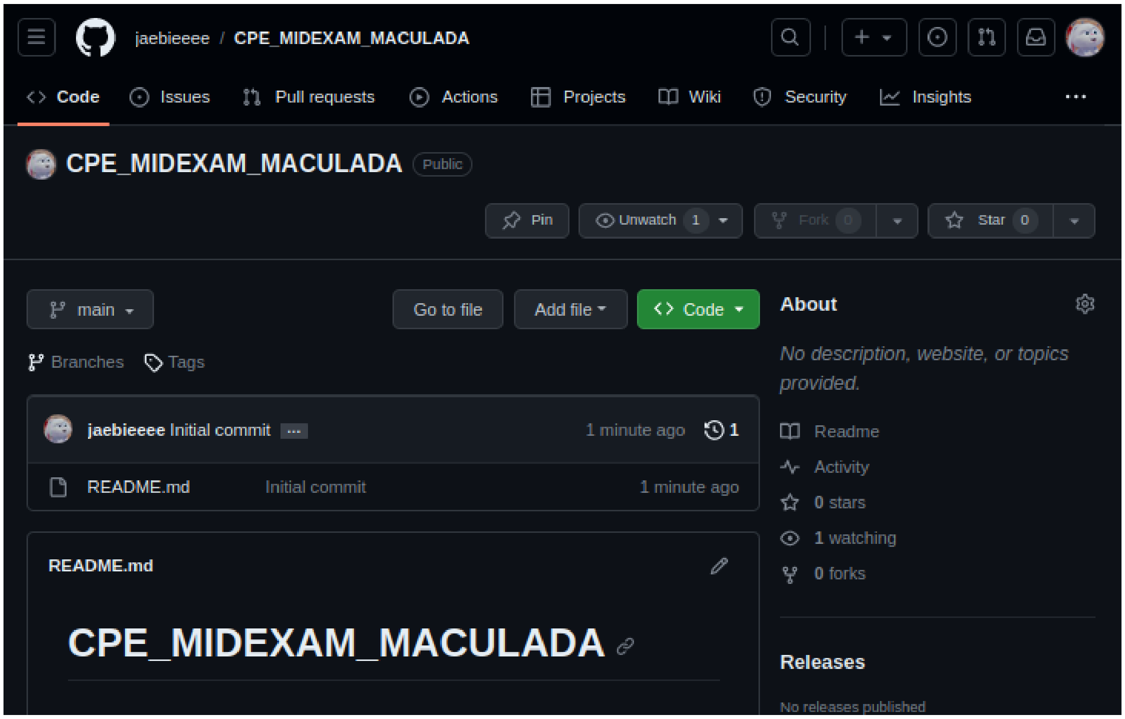
Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools

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.



No description, website, or topics provided.

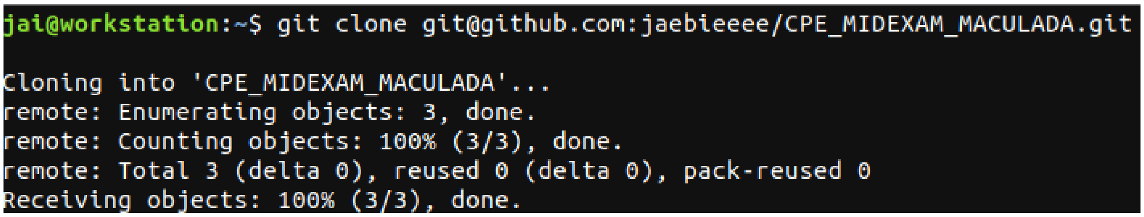
jaebieeee Initial commit ··· 1 minute ago 1

README.md Initial commit 1 minute ago

README.md

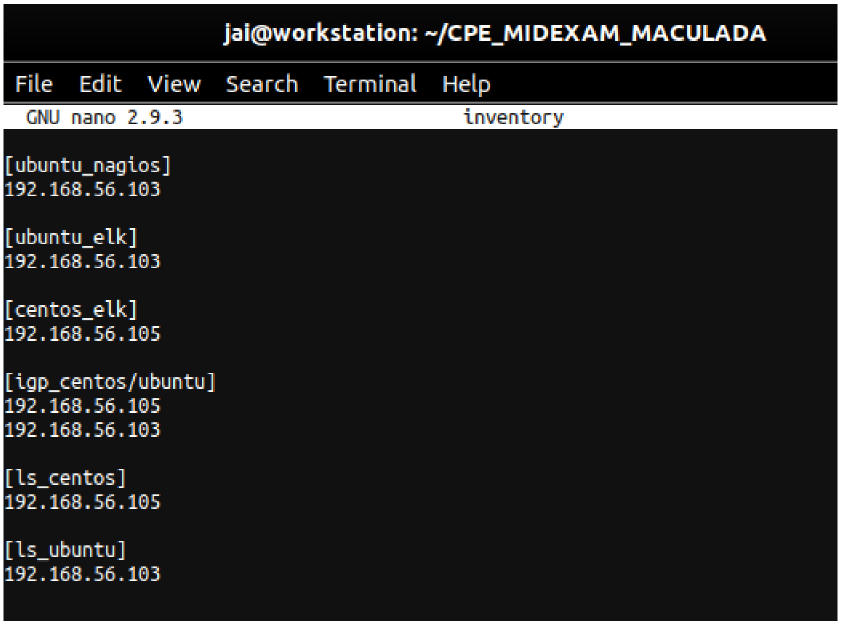
CPE_MIDEXAM_MACULADA

2. Clone the repository and do the following:



```
jai@workstation:~$ git clone git@github.com:jaebieeee/CPE_MIDEXAM_MACULADA.git

Cloning into 'CPE_MIDEXAM_MACULADA'...
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.
```

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

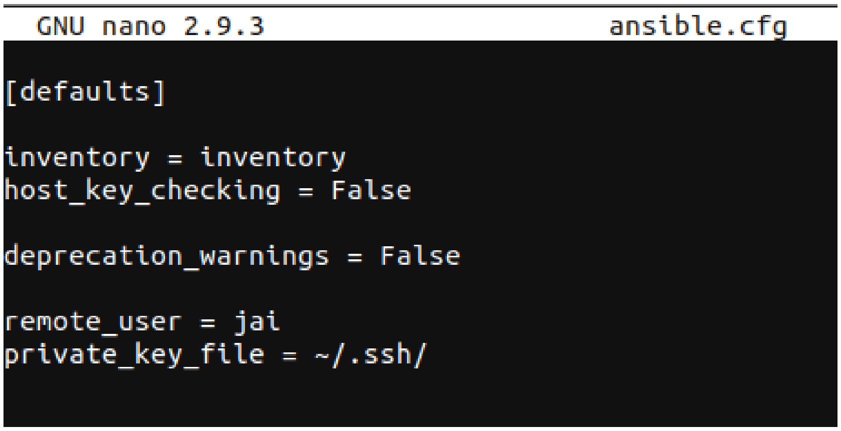
<table>
<tr>
<td>Inventory</td>
<td>
jai@workstation: ~/CPE_MIDEXAM_MACULADA
File Edit View Search Terminal Help
GNU nano 2.9.3 inventory
<pre>
[ubuntu_nagios]
192.168.56.103

[ubuntu_elk]
192.168.56.103

[centos_elk]
192.168.56.105

[igp_centos/ubuntu]
192.168.56.105
192.168.56.103

[ls_centos]
192.168.56.105

[ls_ubuntu]
192.168.56.103
</pre></td>
</tr>
<tr>
<td>ansible.cfg</td>
<td>

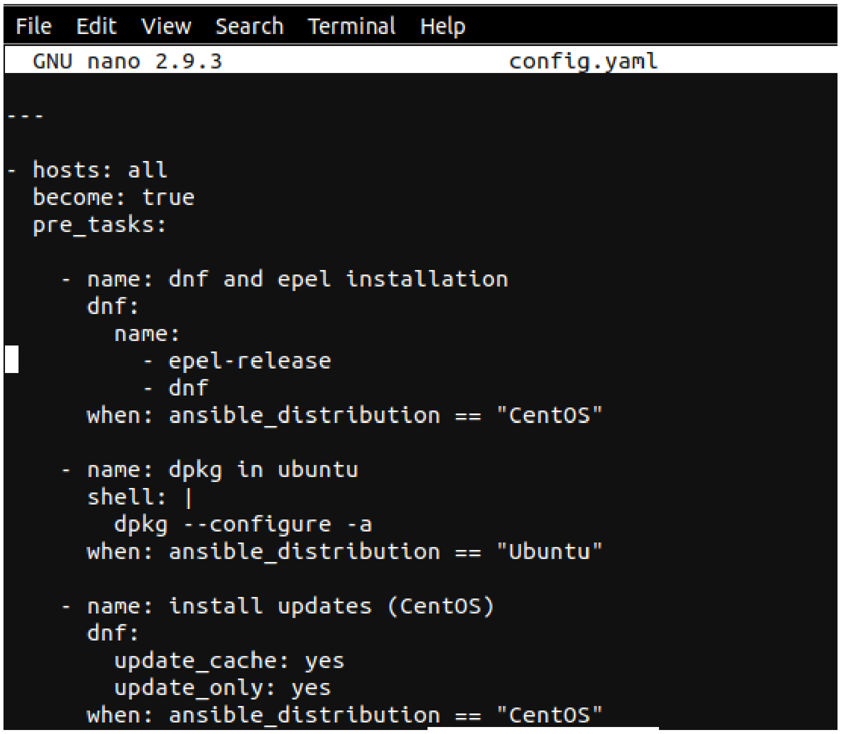
<pre>
[defaults]

inventory = inventory
host_key_checking = False

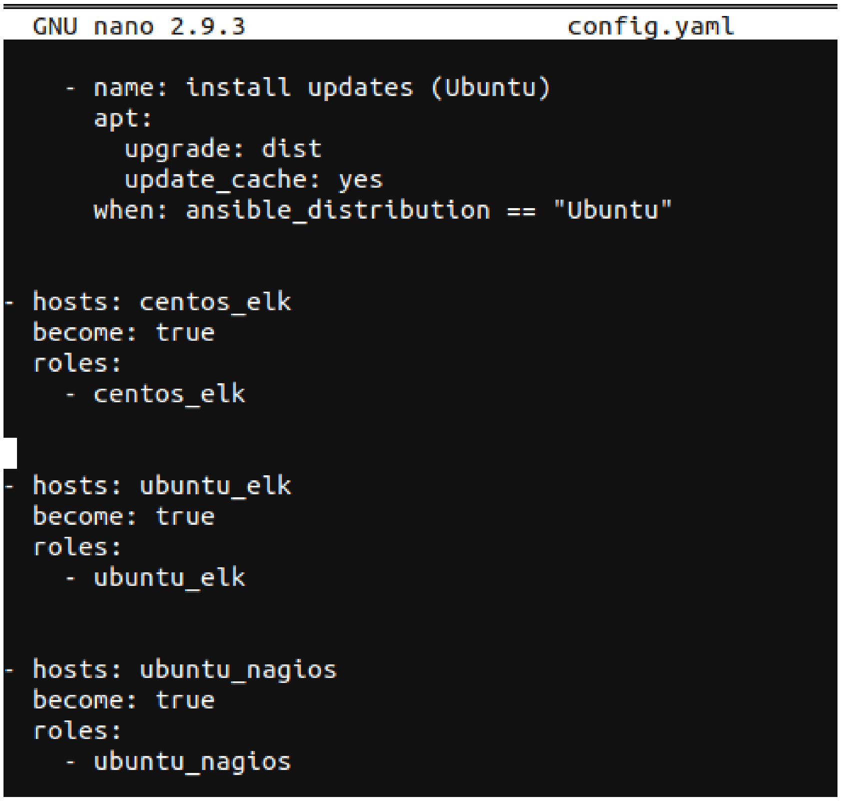
deprecation_warnings = False

remote_user = jai
private_key_file = ~/.ssh/
</pre></td>
</tr>
</table>

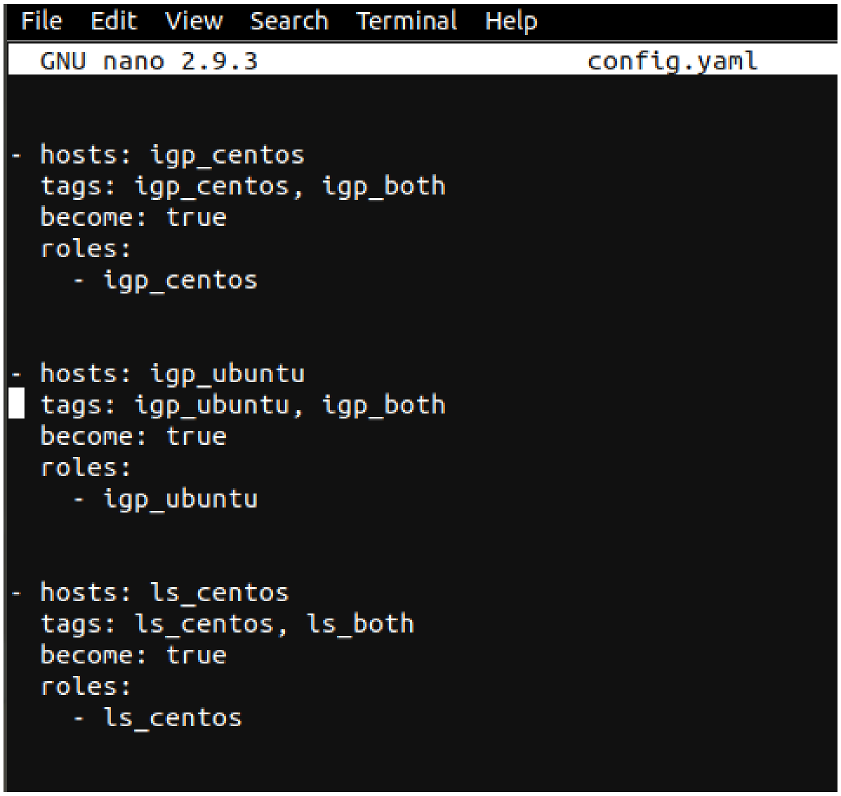
config.yaml



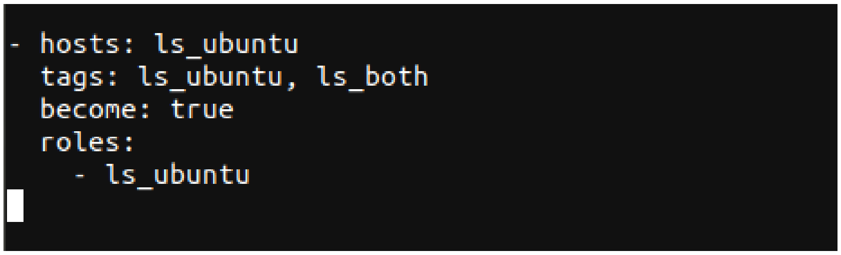
```
  GNU nano 2.9.3                          config.yaml

---

- hosts: all
  become: true
  pre_tasks:

    - name: dnf and epel installation
      dnf:
        name:
          - epel-release
          - dnf
      when: ansible_distribution == "CentOS"

    - name: dpkg in ubuntu
      shell: |
        dpkg --configure -a
      when: ansible_distribution == "Ubuntu"

    - name: install updates (CentOS)
      dnf:
        update_cache: yes
        update_only: yes
      when: ansible_distribution == "CentOS"
```



```
  GNU nano 2.9.3                          config.yaml

    - name: install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
      when: ansible_distribution == "Ubuntu"


- hosts: centos_elk
  become: true
  roles:
    - centos_elk


- hosts: ubuntu_elk
  become: true
  roles:
    - ubuntu_elk


- hosts: ubuntu_nagios
  become: true
  roles:
    - ubuntu_nagios
```



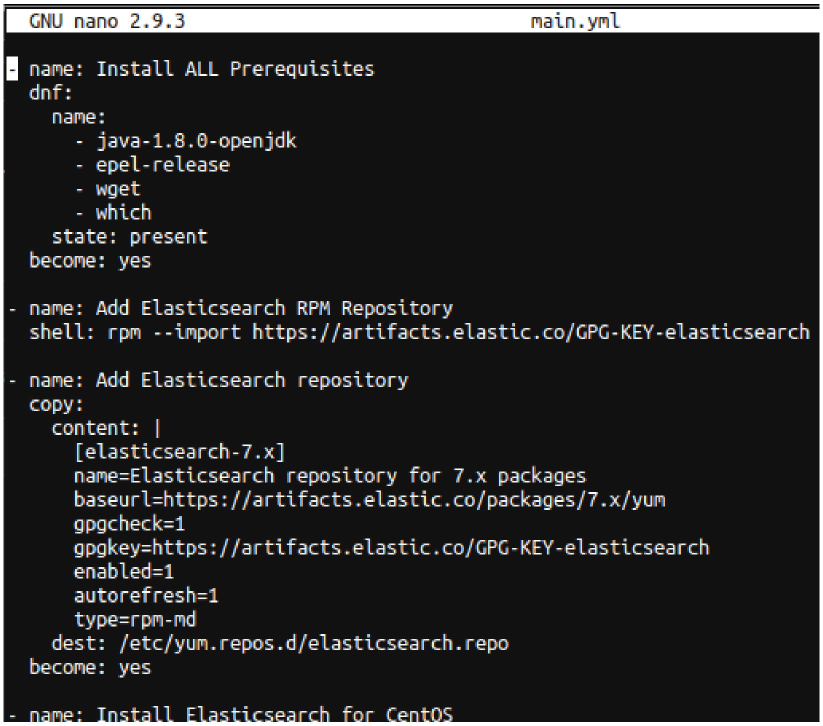
```
  GNU nano 2.9.3                               config.yaml


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


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


- hosts: ls_centos
  tags: ls_centos, ls_both
  become: true
  roles:
    - ls_centos
```



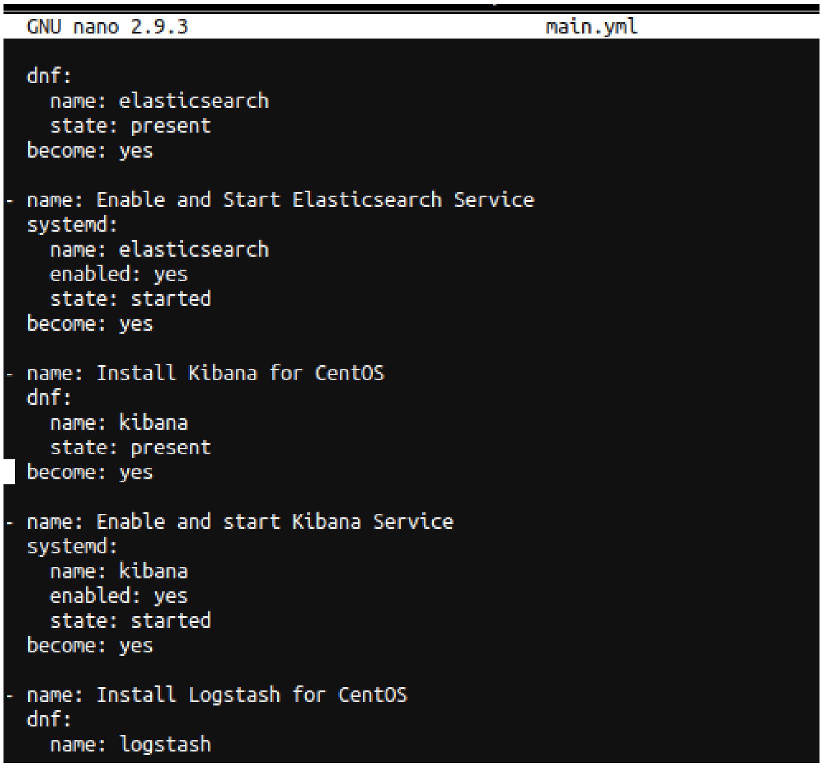
```
- hosts: ls_ubuntu
  tags: ls_ubuntu, ls_both
  become: true
  roles:
    - ls_ubuntu
```

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

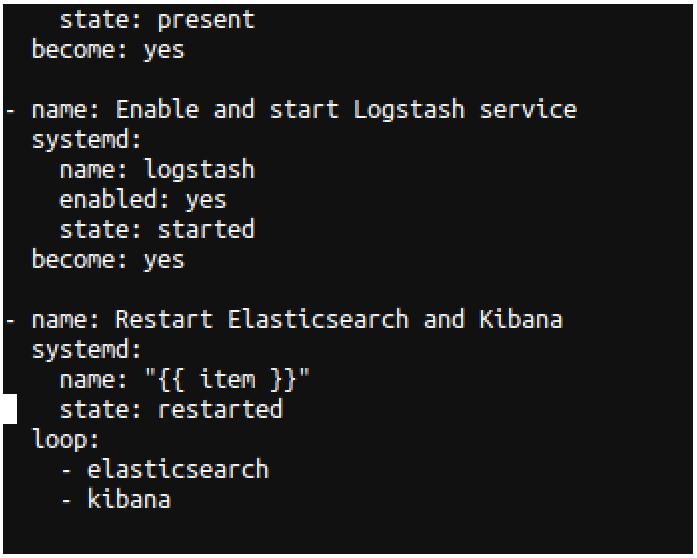
Elastic Stack for CentOS



```
GNU nano 2.9.3                                          main.yml

- name: Install ALL Prerequisites
  dnf:
    name:
      - java-1.8.0-openjdk
      - epel-release
      - wget
      - which
    state: present
  become: yes

- name: Add Elasticsearch RPM Repository
  shell: rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Add Elasticsearch repository
  copy:
    content: |
      [elasticsearch-7.x]
      name=Elasticsearch repository for 7.x packages
      baseurl=https://artifacts.elastic.co/packages/7.x/yum
      gpgcheck=1
      gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
      enabled=1
      autorefresh=1
      type=rpm-md
    dest: /etc/yum.repos.d/elasticsearch.repo
  become: yes

- name: Install Elasticsearch for CentOS
```

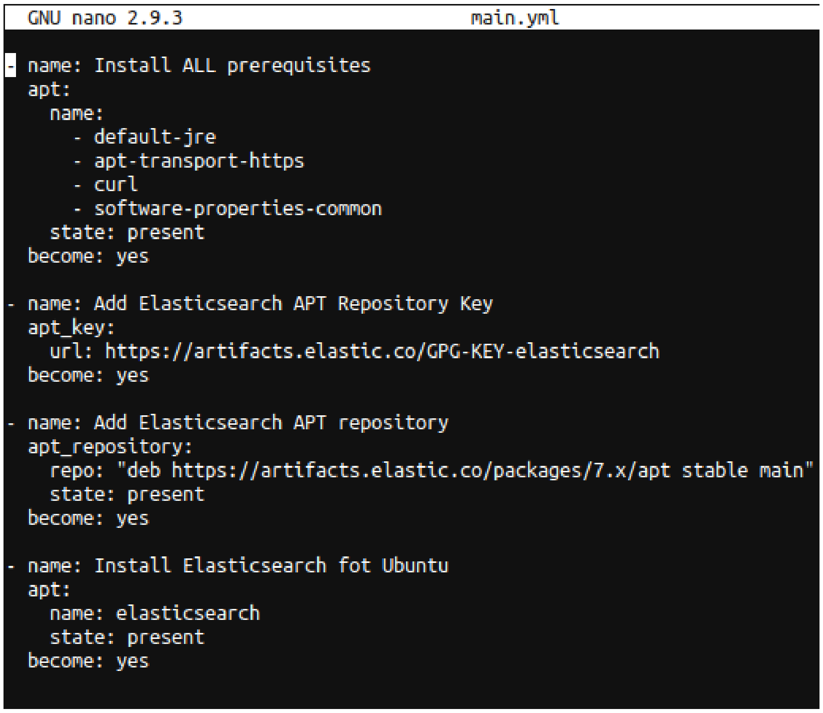


```
GNU nano 2.9.3                                          main.yml

  dnf:
    name: elasticsearch
    state: present
  become: yes

- name: Enable and Start Elasticsearch Service
  systemd:
    name: elasticsearch
    enabled: yes
    state: started
  become: yes

- name: Install Kibana for CentOS
  dnf:
    name: kibana
    state: present
  become: yes

- name: Enable and start Kibana Service
  systemd:
    name: kibana
    enabled: yes
    state: started
  become: yes

- name: Install Logstash for CentOS
  dnf:
    name: logstash
```

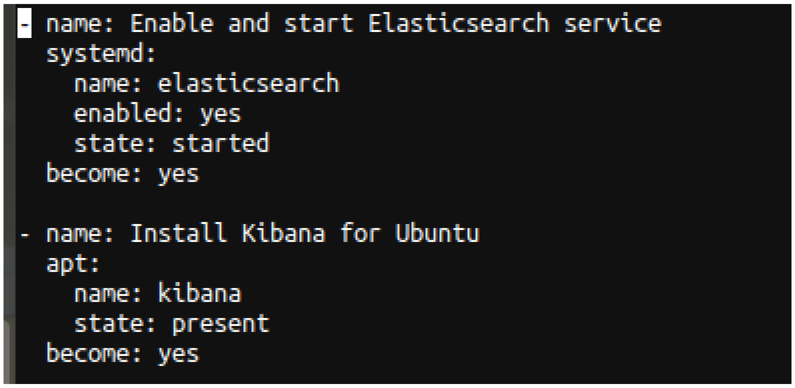


```
    state: present
  become: yes

- name: Enable and start Logstash service
  systemd:
    name: logstash
    enabled: yes
    state: started
  become: yes

- name: Restart Elasticsearch and Kibana
  systemd:
    name: "{{ item }}"
    state: restarted
  loop:
    - elasticsearch
    - kibana
```

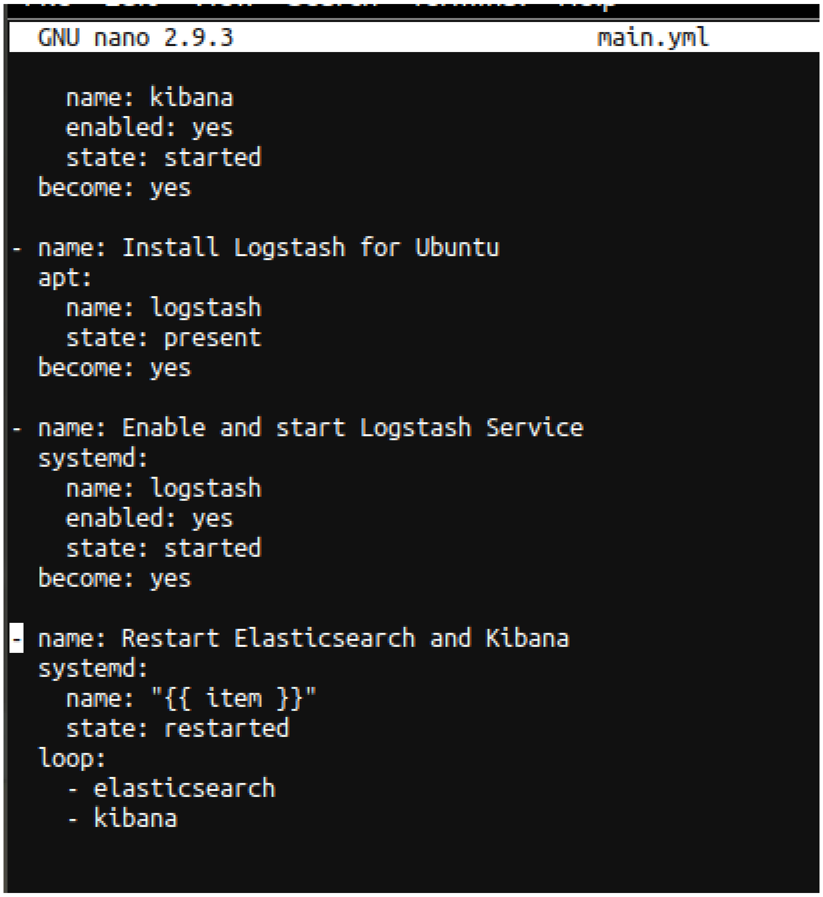
Elastic Stack for Ubuntu



```
  GNU nano 2.9.3                              main.yml

- name: Install ALL prerequisites
  apt:
    name:
      - default-jre
      - apt-transport-https
      - curl
      - software-properties-common
    state: present
  become: yes

- name: Add Elasticsearch APT Repository Key
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
  become: yes

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

- name: Install Elasticsearch fot Ubuntu
  apt:
    name: elasticsearch
    state: present
  become: yes
```

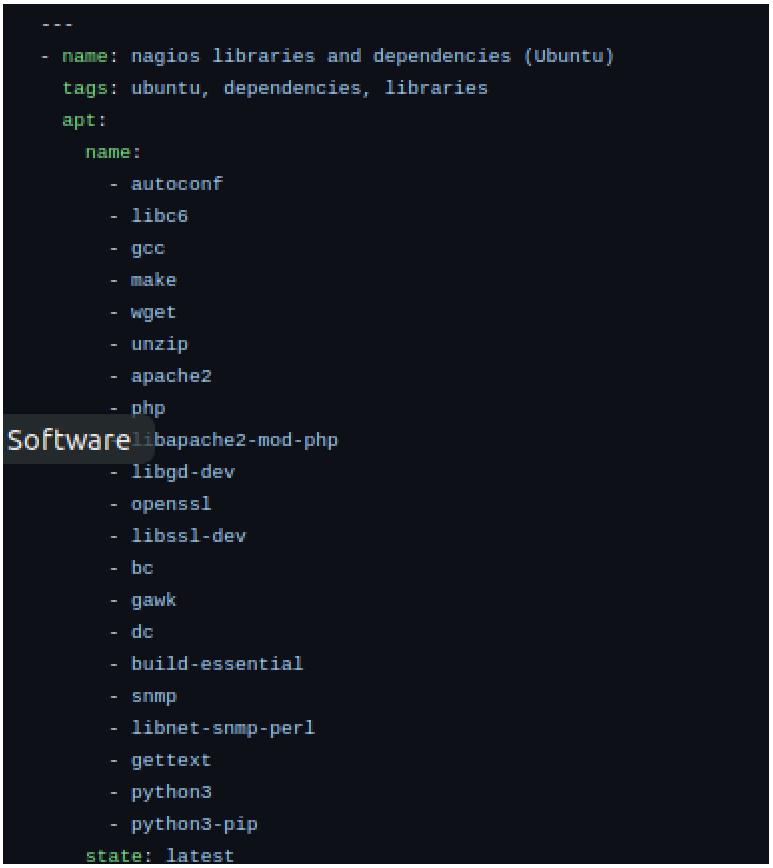


```
- name: Enable and start Elasticsearch service
  systemd:
    name: elasticsearch
    enabled: yes
    state: started
  become: yes

- name: Install Kibana for Ubuntu
  apt:
    name: kibana
    state: present
  become: yes
```

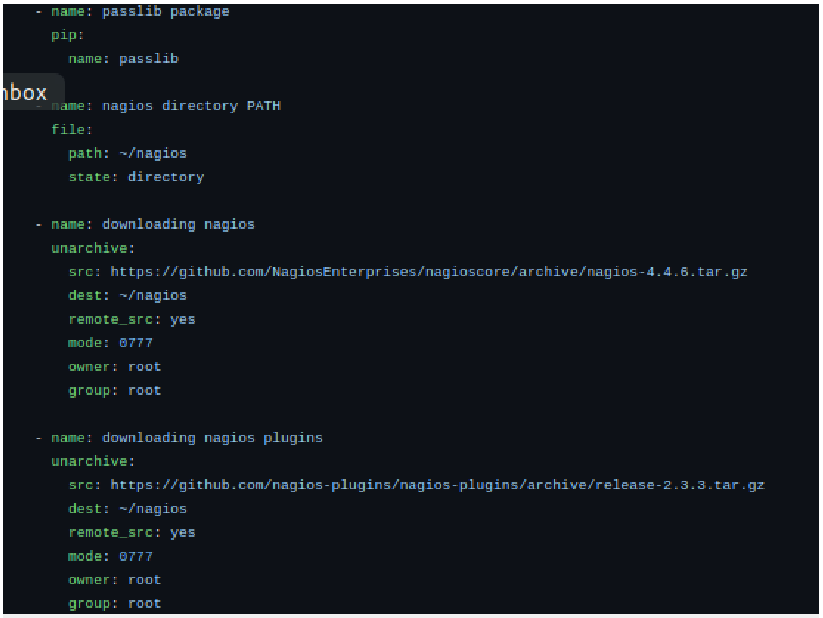


```
  GNU nano 2.9.3                              main.yml

    name: kibana
    enabled: yes
    state: started
  become: yes

- name: Install Logstash for Ubuntu
  apt:
    name: logstash
    state: present
  become: yes

- name: Enable and start Logstash Service
  systemd:
    name: logstash
    enabled: yes
    state: started
  become: yes

- name: Restart Elasticsearch and Kibana
  systemd:
    name: "{{ item }}"
    state: restarted
  loop:
    - elasticsearch
    - kibana
```

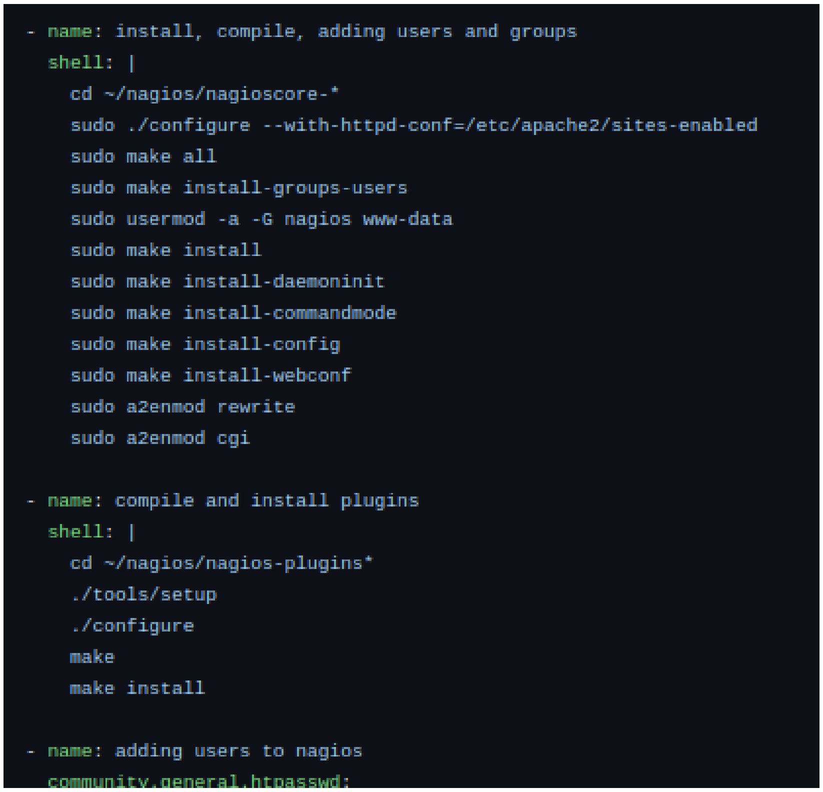
Nagios for Ubuntu



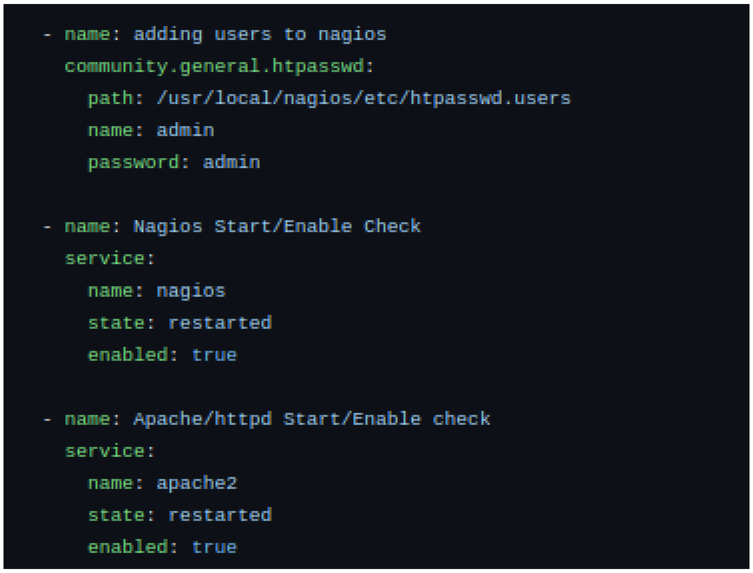
```
---
- name: nagios libraries and dependencies (Ubuntu)
  tags: ubuntu, dependencies, libraries
  apt:
    name:
      - autoconf
      - libc6
      - gcc
      - make
      - wget
      - unzip
      - apache2
      - php
      - libapache2-mod-php
      - libgd-dev
      - openssl
      - libssl-dev
      - bc
      - gawk
      - dc
      - build-essential
      - snmp
      - libnet-snmp-perl
      - gettext
      - python3
      - python3-pip
    state: latest
```



```
- name: passlib package
  pip:
    name: passlib

- name: nagios directory PATH
  file:
    path: ~/nagios
    state: directory

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

- name: downloading nagios plugins
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.3.tar.gz
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root
```



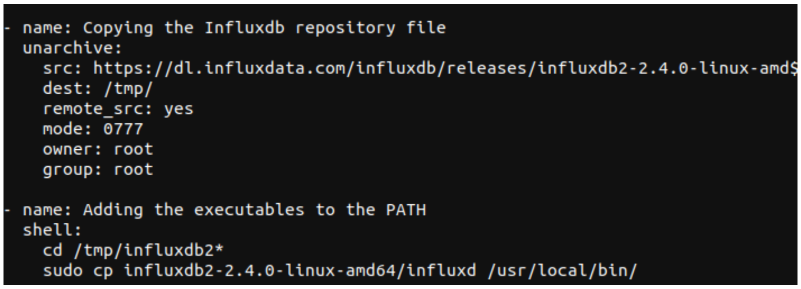
```
- name: install, compile, adding users and groups
  shell: |
    cd ~/nagios/nagioscore-*
    sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
    sudo make all
    sudo make install-groups-users
    sudo usermod -a -G nagios www-data
    sudo make install
    sudo make install-daemoninit
    sudo make install-commandmode
    sudo make install-config
    sudo make install-webconf
    sudo a2enmod rewrite
    sudo a2enmod cgi

- name: compile and install plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install

- name: adding users to nagios
  community.general.htpasswd:
```



```
- name: adding users to nagios
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin

- name: Nagios Start/Enable Check
  service:
    name: nagios
    state: restarted
    enabled: true

- name: Apache/httpd Start/Enable check
  service:
    name: apache2
    state: restarted
    enabled: true
```

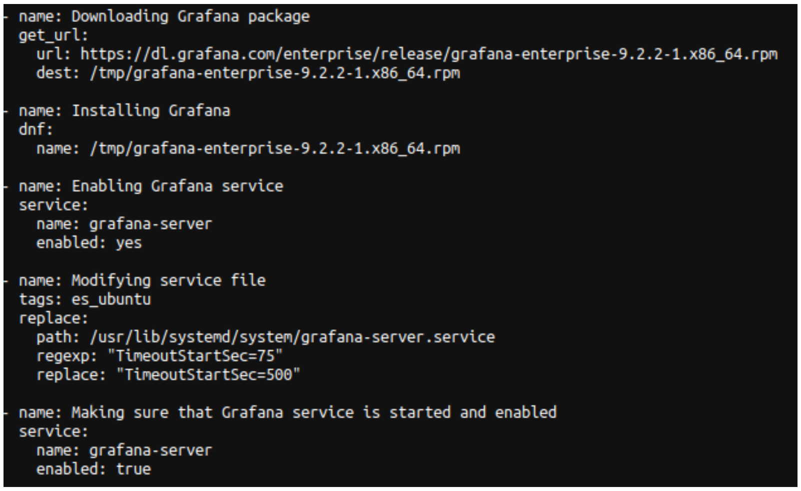
2.3. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)

Influxdb for CentOS



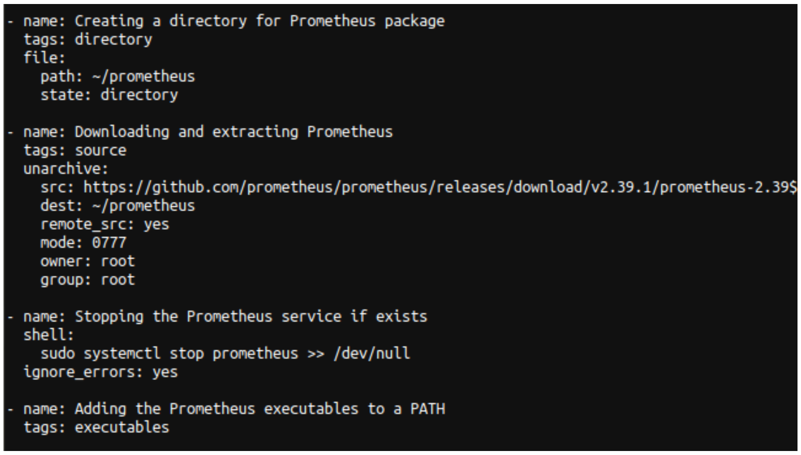
```
- name: Copying the Influxdb repository file
  unarchive:
    src: https://dl.influxdata.com/influxdb/releases/influxdb2-2.4.0-linux-amd$
    dest: /tmp/
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Adding the executables to the PATH
  shell:
    cd /tmp/influxdb2*
    sudo cp influxdb2-2.4.0-linux-amd64/influxd /usr/local/bin/
```

Grafana for CentOS

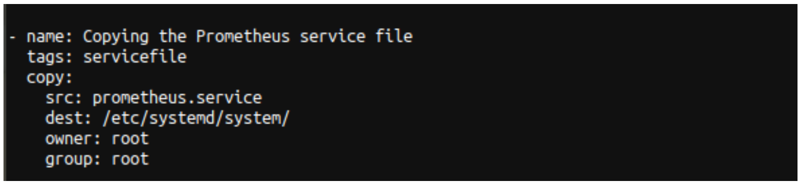


```
- name: Downloading Grafana package
  get_url:
    url: https://dl.grafana.com/enterprise/release/grafana-enterprise-9.2.2-1.x86_64.rpm
    dest: /tmp/grafana-enterprise-9.2.2-1.x86_64.rpm

- name: Installing Grafana
  dnf:
    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: Making sure that Grafana service is started and enabled
  service:
    name: grafana-server
    enabled: true
```

Prometheus for CentOS

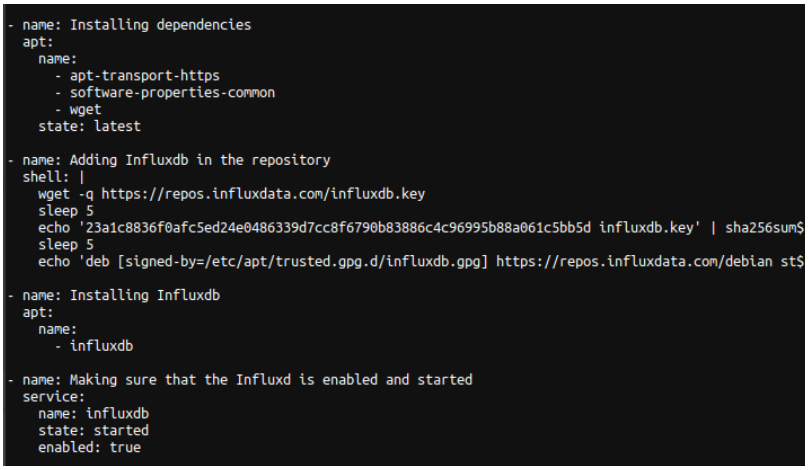


```
- name: Creating a directory for Prometheus package
  tags: directory
  file:
    path: ~/prometheus
    state: directory

- name: Downloading and extracting Prometheus
  tags: source
  unarchive:
    src: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39$
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root

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

- name: Adding the Prometheus executables to a PATH
  tags: executables
```



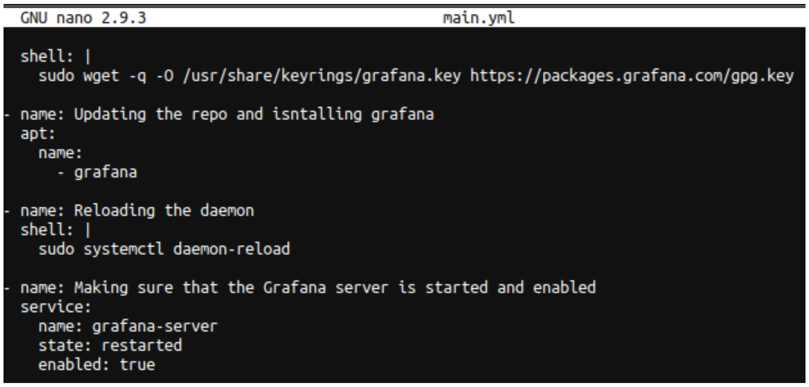
```
- name: Copying the Prometheus service file
  tags: servicefile
  copy:
    src: prometheus.service
    dest: /etc/systemd/system/
    owner: root
    group: root
```

Influxdb for Ubuntu



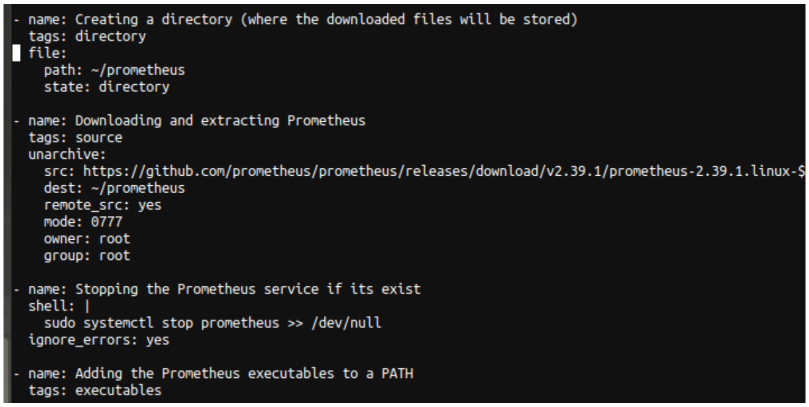
```
- name: Installing dependencies
  apt:
    name:
      - apt-transport-https
      - software-properties-common
      - wget
    state: latest

- name: Adding Influxdb in the repository
  shell: |
    wget -q https://repos.influxdata.com/influxdb.key
    sleep 5
    echo '23a1c8836f0afc5ed24e0486339d7cc8f6790b83886c4c96995b88a061c5bb5d influxdb.key' | sha256sum$
    sleep 5
    echo 'deb [signed-by=/etc/apt/trusted.gpg.d/influxdb.gpg] https://repos.influxdata.com/debian st$

- name: Installing Influxdb
  apt:
    name:
      - influxdb

- name: Making sure that the Influxd is enabled and started
  service:
    name: influxdb
    state: started
    enabled: true
```

Grafana for Ubuntu

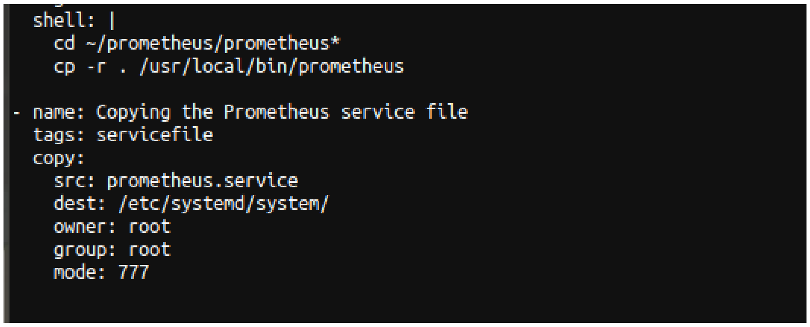


```
GNU nano 2.9.3                              main.yml

  shell: |
    sudo wget -q -O /usr/share/keyrings/grafana.key https://packages.grafana.com/gpg.key

- name: Updating the repo and isntalling grafana
  apt:
    name:
      - grafana

- name: Reloading the daemon
  shell: |
    sudo systemctl daemon-reload

- name: Making sure that the Grafana server is started and enabled
  service:
    name: grafana-server
    state: restarted
    enabled: true
```

Prometheus for Ubuntu



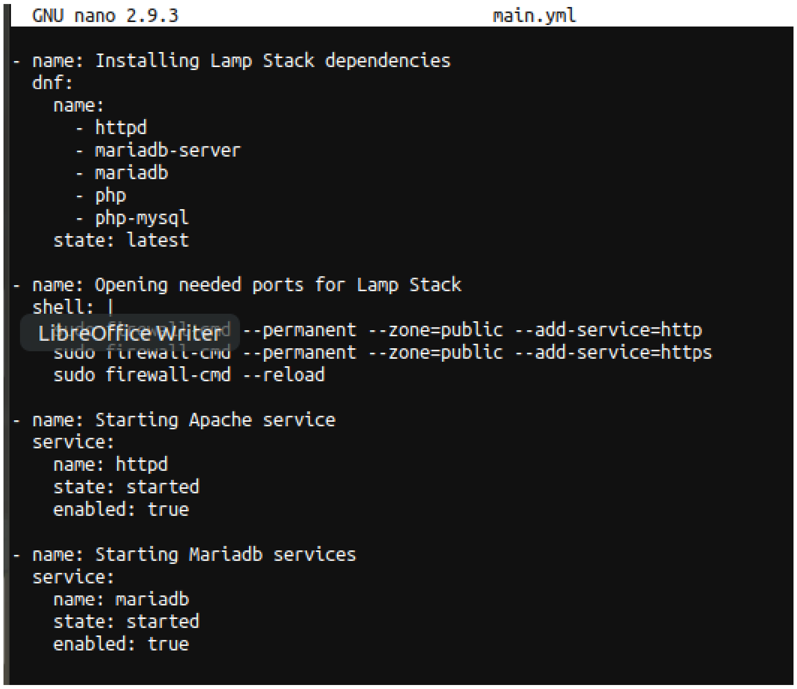
```
- name: Creating a directory (where the downloaded files will be stored)
  tags: directory
  file:
    path: ~/prometheus
    state: directory

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

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

- name: Adding the Prometheus executables to a PATH
  tags: executables
```



```
  shell: |
    cd ~/prometheus/prometheus*
    cp -r . /usr/local/bin/prometheus

- name: Copying the Prometheus service file
  tags: servicefile
  copy:
    src: prometheus.service
    dest: /etc/systemd/system/
    owner: root
    group: root
    mode: 777
```

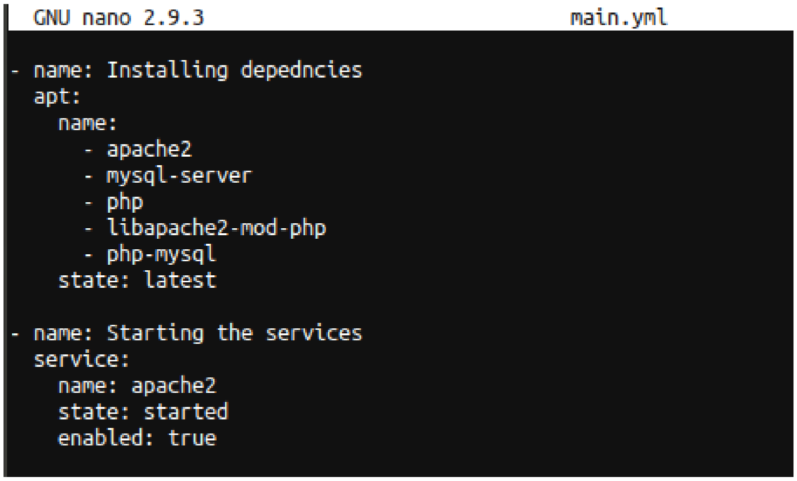
2.4. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)

Lamp Stack for CentOS



```
GNU nano 2.9.3                                  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
```

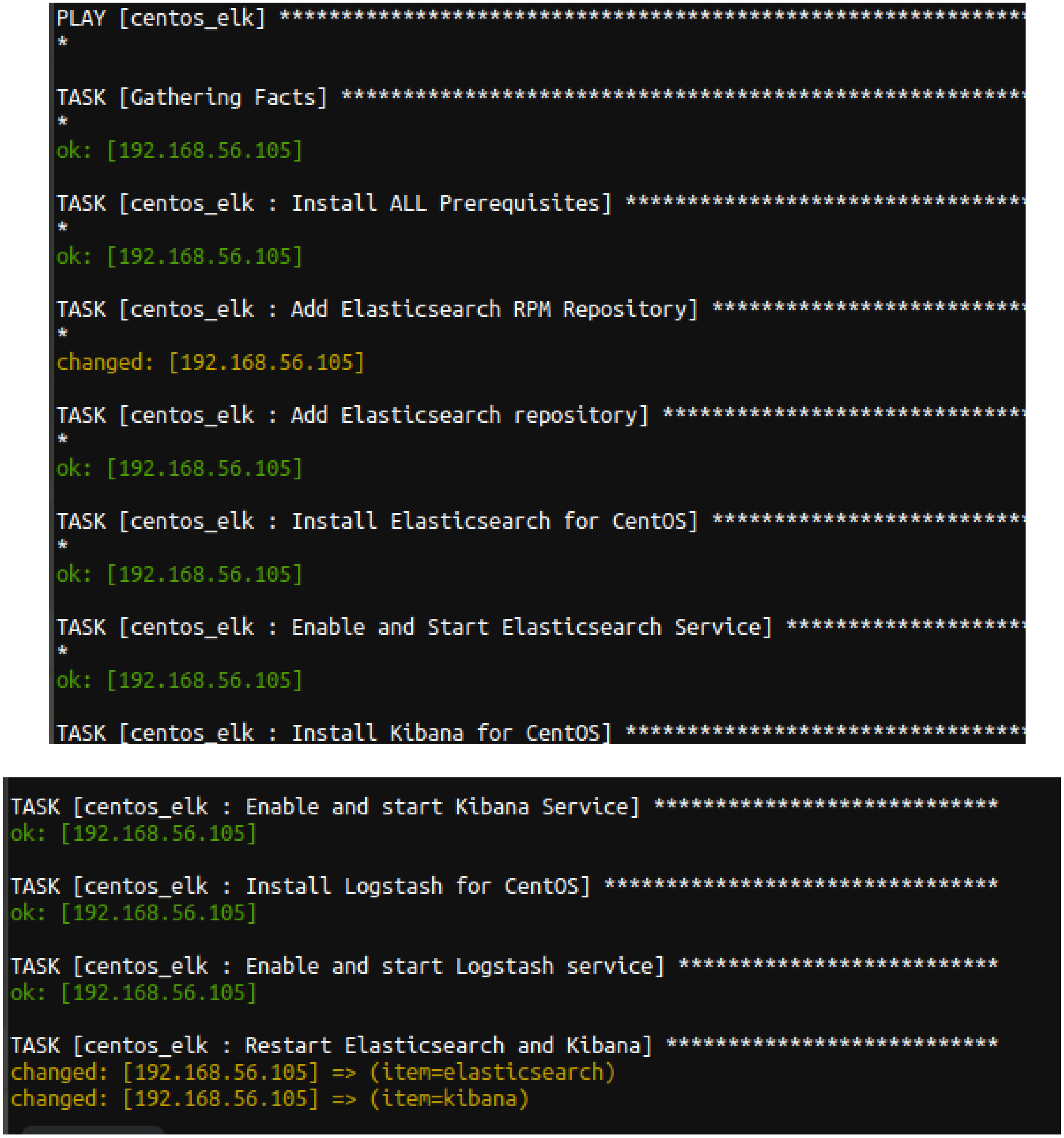
Lamp Stack for Ubuntu



```
GNU nano 2.9.3                                  main.yml

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

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

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.

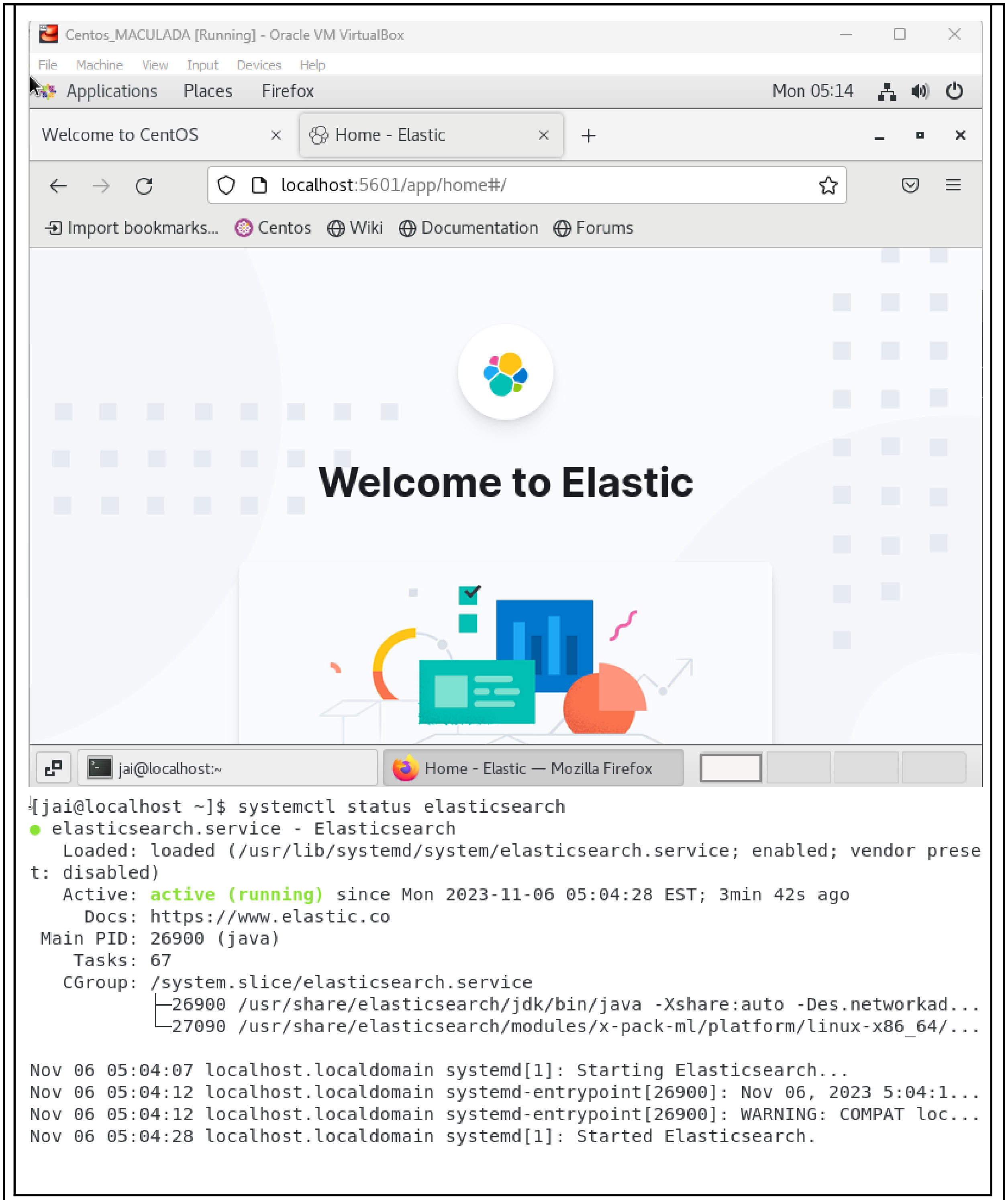
3. Output (screenshots and explanations)



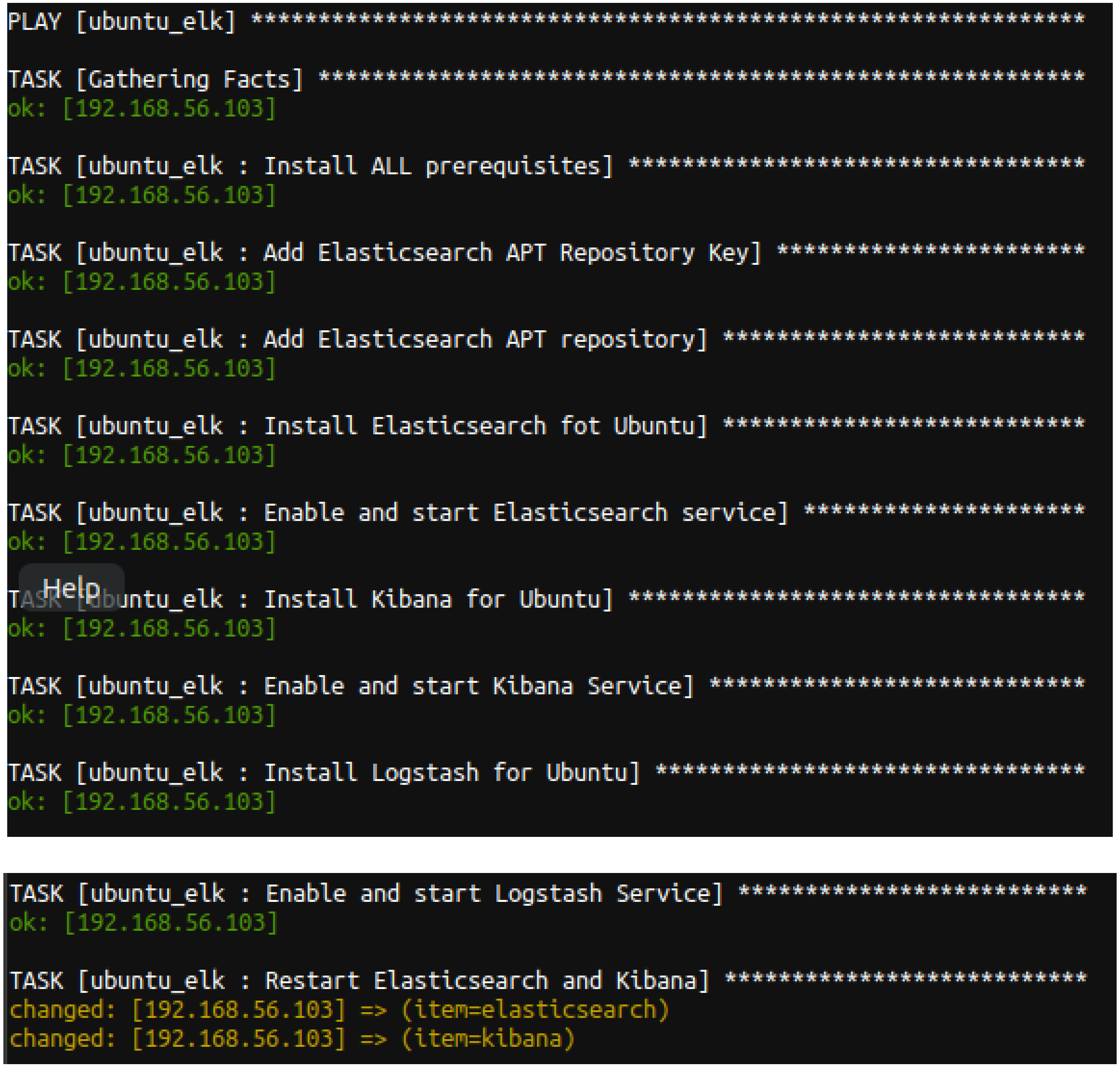
```
PLAY [centos_elk] ************************************************************
*

TASK [Gathering Facts] *******************************************************
*
ok: [192.168.56.105]

TASK [centos_elk : Install ALL Prerequisites] ********************************
*
ok: [192.168.56.105]

TASK [centos_elk : Add Elasticsearch RPM Repository] *************************
*
changed: [192.168.56.105]

TASK [centos_elk : Add Elasticsearch repository] *****************************
*
ok: [192.168.56.105]

TASK [centos_elk : Install Elasticsearch for CentOS] *************************
*
ok: [192.168.56.105]

TASK [centos_elk : Enable and Start Elasticsearch Service] *******************
*
ok: [192.168.56.105]

TASK [centos_elk : Install Kibana for CentOS] ********************************
```

```
TASK [centos_elk : Enable and start Kibana Service] ****************************
ok: [192.168.56.105]

TASK [centos_elk : Install Logstash for CentOS] ********************************
ok: [192.168.56.105]

TASK [centos_elk : Enable and start Logstash service] **************************
ok: [192.168.56.105]

TASK [centos_elk : Restart Elasticsearch and Kibana] ***************************
changed: [192.168.56.105] => (item=elasticsearch)
changed: [192.168.56.105] => (item=kibana)
```

CENTOS_ELK


Centos_MACULADA [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Firefox
Mon 05:14
Welcome to CentOS
Home - Elastic
localhost:5601/app/home#/
Import bookmarks... Centos Wiki Documentation Forums
Welcome to Elastic
jai@localhost:~
Home - Elastic — Mozilla Firefox

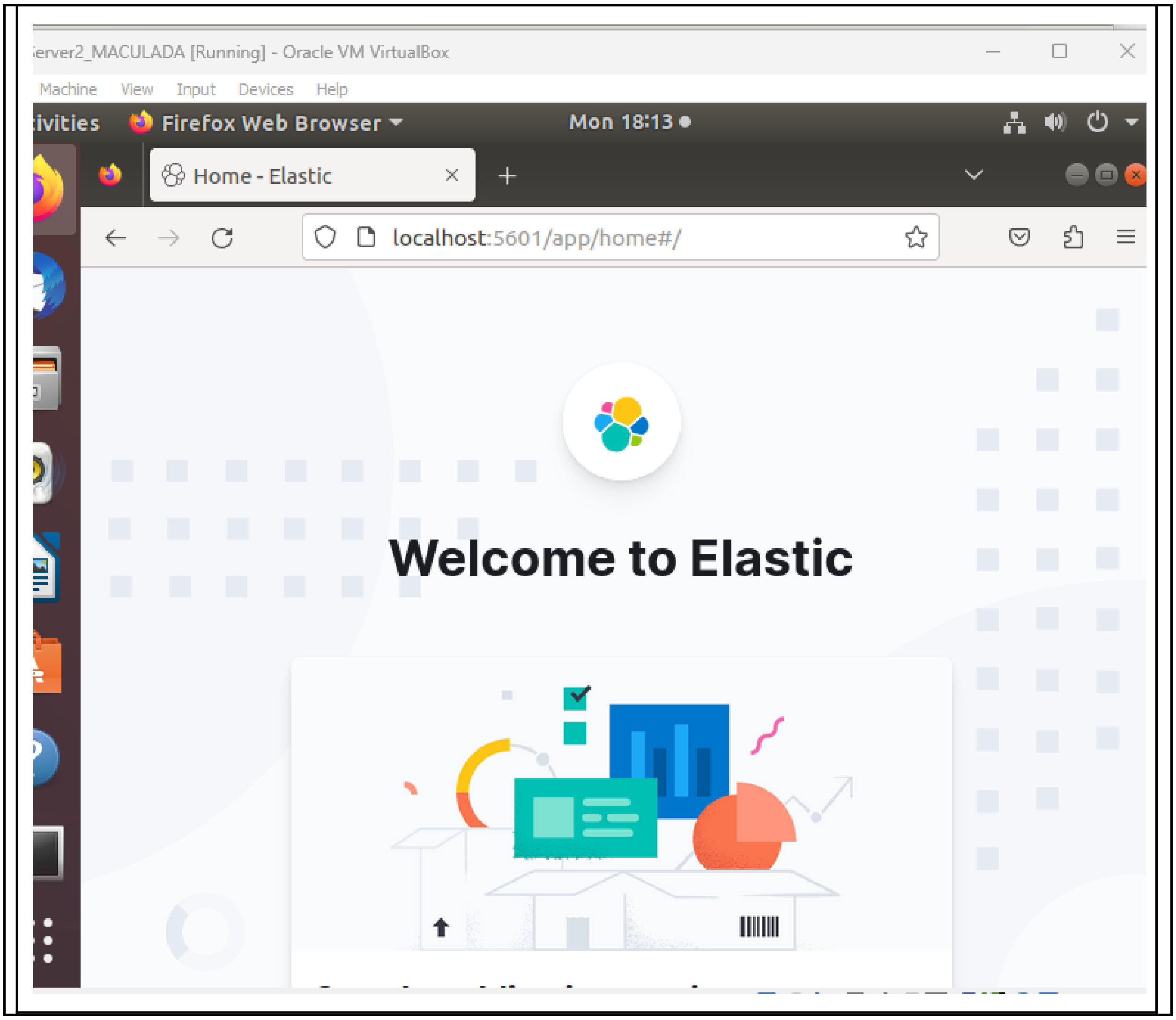
```
[jai@localhost ~]$ systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendor prese
t: disabled)
   Active: active (running) since Mon 2023-11-06 05:04:28 EST; 3min 42s ago
     Docs: https://www.elastic.co
 Main PID: 26900 (java)
    Tasks: 67
   CGroup: /system.slice/elasticsearch.service
           ├─26900 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkad...
           └─27090 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/...

Nov 06 05:04:07 localhost.localdomain systemd[1]: Starting Elasticsearch...
Nov 06 05:04:12 localhost.localdomain systemd-entrypoint[26900]: Nov 06, 2023 5:04:1...
Nov 06 05:04:12 localhost.localdomain systemd-entrypoint[26900]: WARNING: COMPAT loc...
Nov 06 05:04:28 localhost.localdomain systemd[1]: Started Elasticsearch.
```



```
PLAY [ubuntu_elk] **************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install ALL prerequisites] **********************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Add Elasticsearch APT Repository Key] ***********************
ok: [192.168.56.103]

TASK [ubuntu_elk : Add Elasticsearch APT repository] ***************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Elasticsearch fot Ubuntu] ***************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Enable and start Elasticsearch service] *********************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Kibana for Ubuntu] **********************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Enable and start Kibana Service] ****************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Install Logstash for Ubuntu] ********************************
ok: [192.168.56.103]
```

```
TASK [ubuntu_elk : Enable and start Logstash Service] **************************
ok: [192.168.56.103]

TASK [ubuntu_elk : Restart Elasticsearch and Kibana] ***************************
changed: [192.168.56.103] => (item=elasticsearch)
changed: [192.168.56.103] => (item=kibana)
```

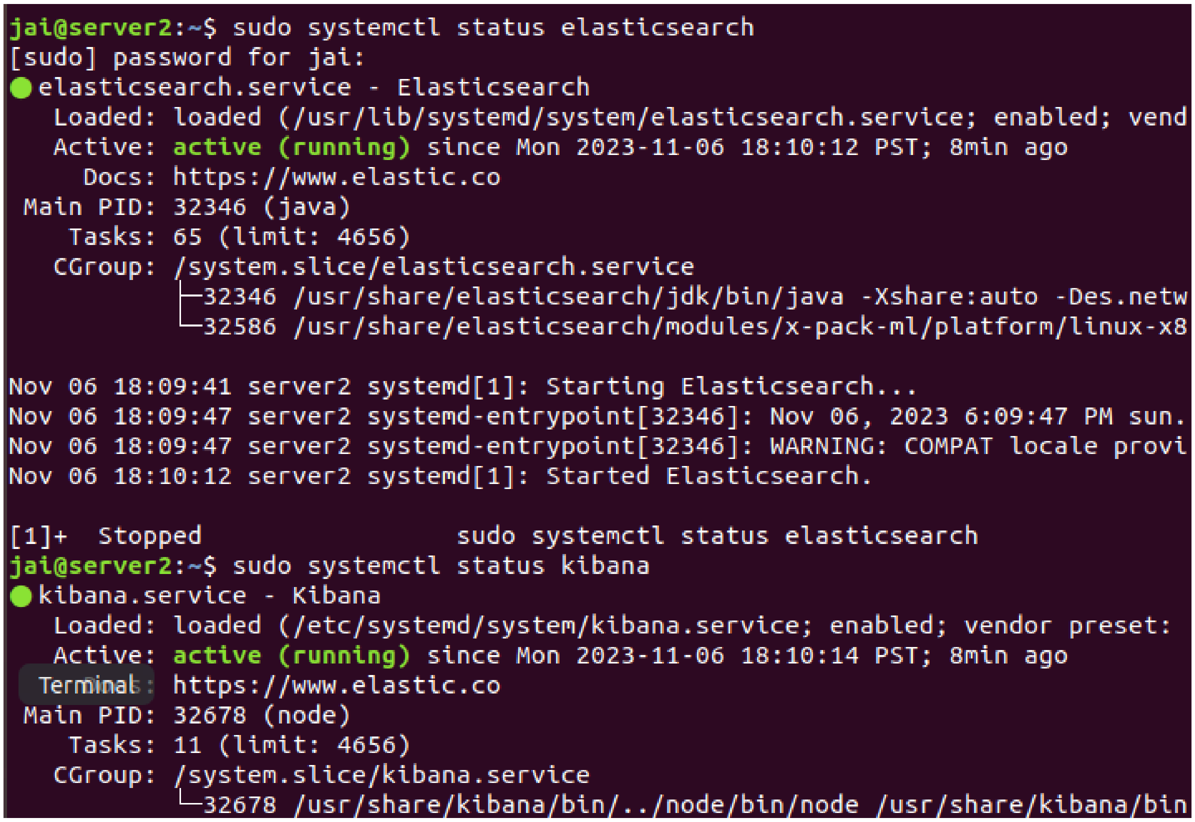
UBUNTU_ELK



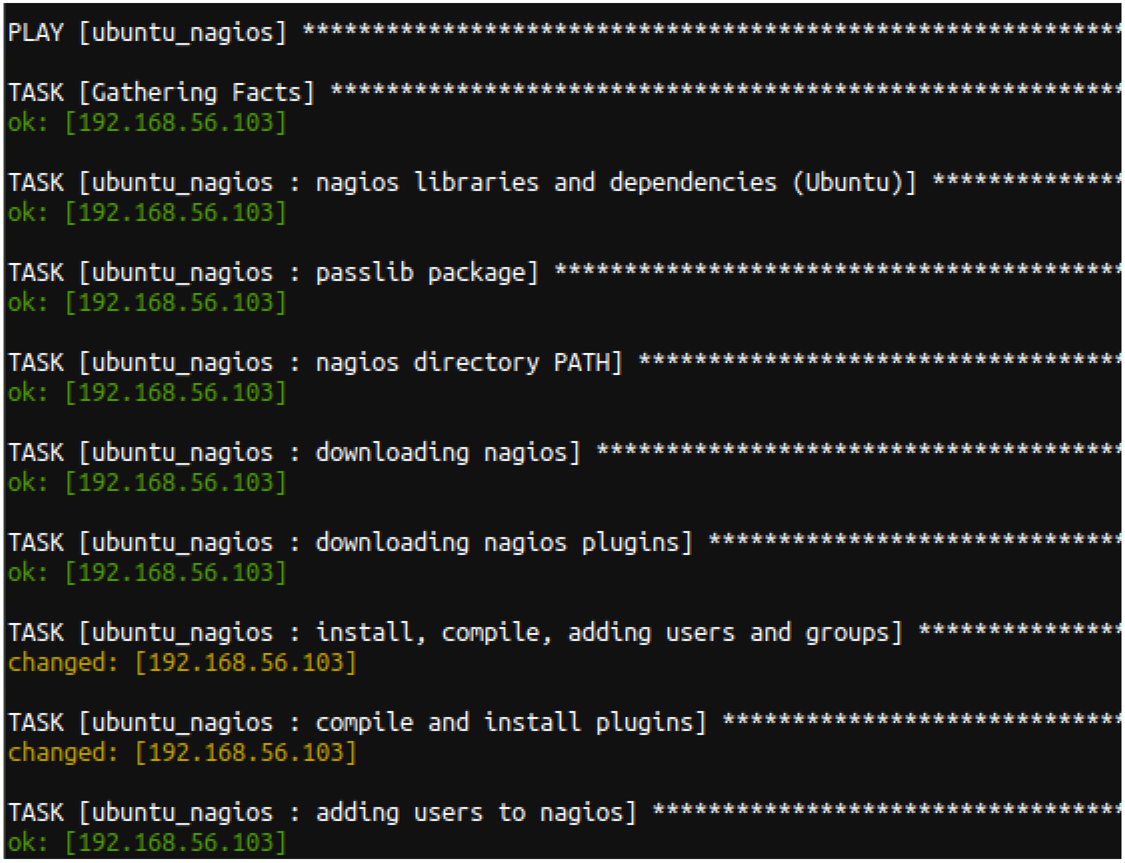
Server2_MACULADA [Running] - Oracle VM VirtualBox

localhost:5601/app/home#/

Welcome to Elastic

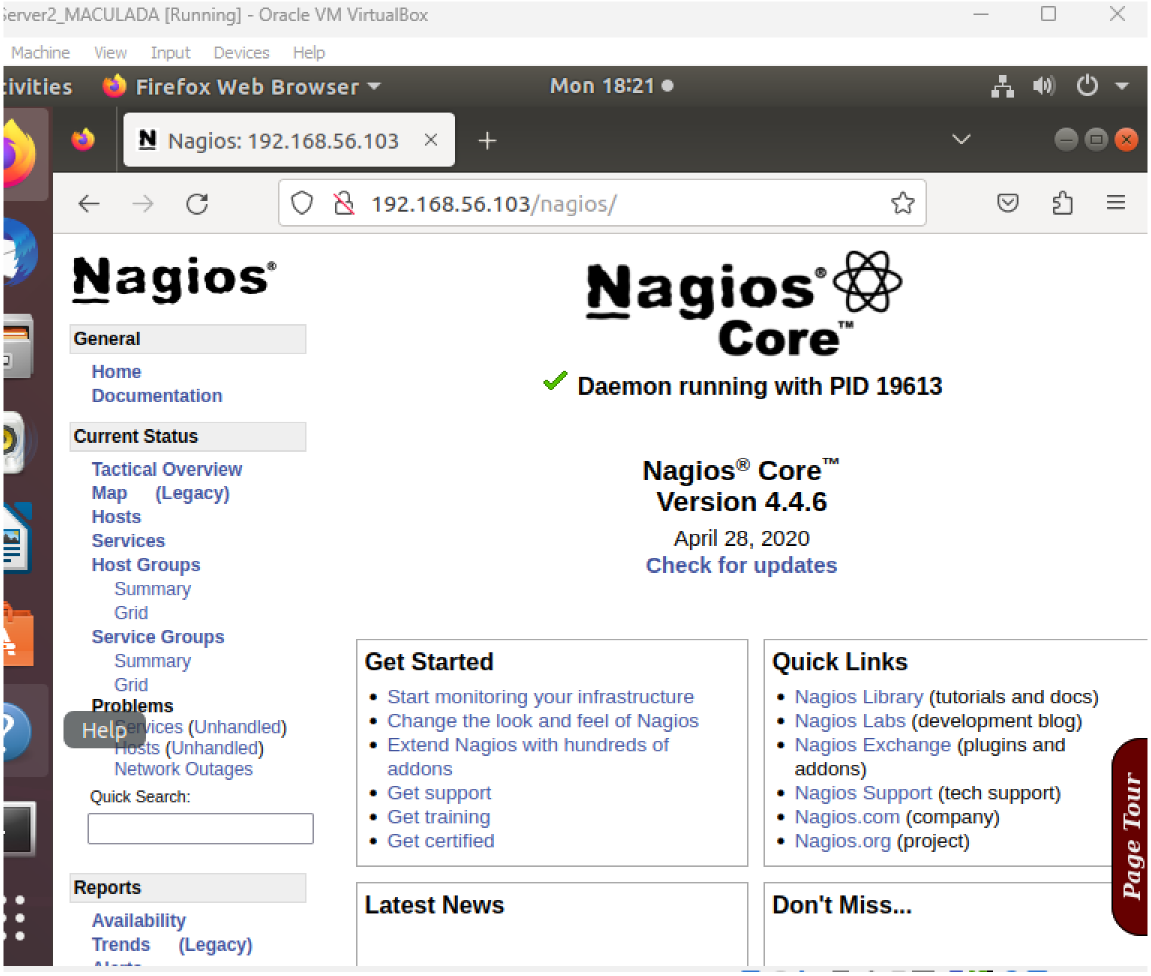


```
jai@server2:~$ sudo systemctl status elasticsearch
[sudo] password for jai:
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vend
   Active: active (running) since Mon 2023-11-06 18:10:12 PST; 8min ago
     Docs: https://www.elastic.co
 Main PID: 32346 (java)
    Tasks: 65 (limit: 4656)
   CGroup: /system.slice/elasticsearch.service
           ├─32346 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.netw
           └─32586 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x8

Nov 06 18:09:41 server2 systemd[1]: Starting Elasticsearch...
Nov 06 18:09:47 server2 systemd-entrypoint[32346]: Nov 06, 2023 6:09:47 PM sun.
Nov 06 18:09:47 server2 systemd-entrypoint[32346]: WARNING: COMPAT locale provi
Nov 06 18:10:12 server2 systemd[1]: Started Elasticsearch.

[1]+  Stopped                 sudo systemctl status elasticsearch
jai@server2:~$ sudo systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset:
   Active: active (running) since Mon 2023-11-06 18:10:14 PST; 8min ago
  Terminal: https://www.elastic.co
 Main PID: 32678 (node)
    Tasks: 11 (limit: 4656)
   CGroup: /system.slice/kibana.service
           └─32678 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin
```



```
PLAY [ubuntu_nagios] **********************************************************

TASK [Gathering Facts] ********************************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : nagios libraries and dependencies (Ubuntu)] *************
ok: [192.168.56.103]

TASK [ubuntu_nagios : passlib package] ****************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : nagios directory PATH] **********************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : downloading nagios] *************************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : downloading nagios plugins] *****************************
ok: [192.168.56.103]

TASK [ubuntu_nagios : install, compile, adding users and groups] **************
changed: [192.168.56.103]

TASK [ubuntu_nagios : compile and install plugins] ****************************
changed: [192.168.56.103]

TASK [ubuntu_nagios : adding users to nagios] *********************************
ok: [192.168.56.103]
```

UBUNTU_NAGIOS



✓ Daemon running with PID 19613

Nagios® Core™
Version 4.4.6
April 28, 2020
Check for updates

Get Started

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

Quick Links

- Nagios Library (tutorials and docs)
- Nagios Labs (development blog)
- Nagios Exchange (plugins and addons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

Latest News

Don't Miss...

IGP_CENTOS/UBUNTU

LS_CENTOS

GitHub link:
https://github.com/jaebieeee/CPE_MIDEXAM_MACULADA.git

Conclusions: (link your conclusion from the objective)

In performing this Midterm Exam, I utilized all my playbook for installing nagios and elastic stack. In conclusion, tackling the midterm exam was an insightful journey into the world of system administration and monitoring. Installing Elastic Stack on both Ubuntu and CentOS taught me the importance of centralized logging and real-time data analysis. Setting up Nagios exclusively on CentOS sharpened my skills in network monitoring and alerting. Deploying Grafana, InfluxDB, and Prometheus on both operating systems demonstrated the power of data visualization and performance monitoring. Lastly, configuring LAMP stacks on CentOS and Ubuntu highlighted the versatility of web servers. This experience has broadened my horizons and equipped me with invaluable skills for managing and optimizing diverse server environments.