Name: Juanson, Aliya Dane P.	**Date Performed:** October 30, 2024
Course/Section: CPE 31S2	**Date Submitted:** November 04, 2024
Instructor: Sir Robin Valuenza	**Semester and SY:** 2024-2025

Activity 10: Install, Configure, and Manage Log Monitoring tools

1. Objectives

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

2. Discussion

Log monitoring software scans and monitors log files generated by servers, applications, and networks. By detecting and alerting users to patterns in these log files, log monitoring software helps solve performance and security issues. System administrators use log monitoring software to detect common important events indicated by log files.

Log monitoring software helps maintain IT infrastructure performance and pinpoints issues to prevent downtime and mitigate risks. These tools will often integrate with IT alerting software, log analysis software, and other IT issue resolution products to more aptly flesh out the IT infrastructure maintenance ecosystem.

To qualify for inclusion in the Log Monitoring category, a product must:

- Monitor the log files generated by servers, applications, or networks
- Alert users when important events are detected
- Provide reporting capabilities for log files

Elastic Stack

ELK suite stands for Elasticsearch, Kibana, Beats, and Logstash (also known as the ELK Stack). Source: https://www.elastic.co/elastic-stack

The Elastic Stack is a group of open source products from Elastic designed to help users take data from any type of source and in any format, and search, analyze and visualize that data in real time. The product group was formerly known as the ELK Stack for the core products in the group -- Elasticsearch, Logstash and Kibana -- but has been rebranded as the Elastic Stack. A fourth product, Beats, was subsequently added to the stack. The Elastic Stack can be deployed on premises or made available as software as a service (SaaS). Elasticsearch supports Amazon Web Services (AWS), Google Cloud Platform and Microsoft Azure.

GrayLog

Graylog is a powerful platform that allows for easy log management of both structured and unstructured data along with debugging applications.

It is based on Elasticsearch, MongoDB, and Scala. Graylog has a main server, which receives data from its clients installed on different servers, and a web interface, which visualizes the data and allows to work with logs aggregated by the main server.

We use Graylog primarily as the stash for the logs of the web applications we build. However, it is also effective when working with raw strings (i.e. syslog): the tool parses it into the structured data we need. It also allows advanced custom search in the logs using structured queries. In other words, when integrated properly with a web app, Graylog helps engineers to analyze the system behavior on almost per code line basis.

Source: https://www.graylog.org/products/open-source

3. Tasks

1. Create a playbook that:
    a. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash)
2. Apply the concept of creating roles.
3. Describe how you did step 1. (Provide screenshots and explanations in your report. Make your report detailed such that it will look like a manual.)
4. Show an output of the installed Elastic Stack for both Ubuntu and CentOS.
5. Make sure to create a new repository in GitHub for this activity.

4. Output (screenshots and explanations)

Create a repository and git clone to your local machine.

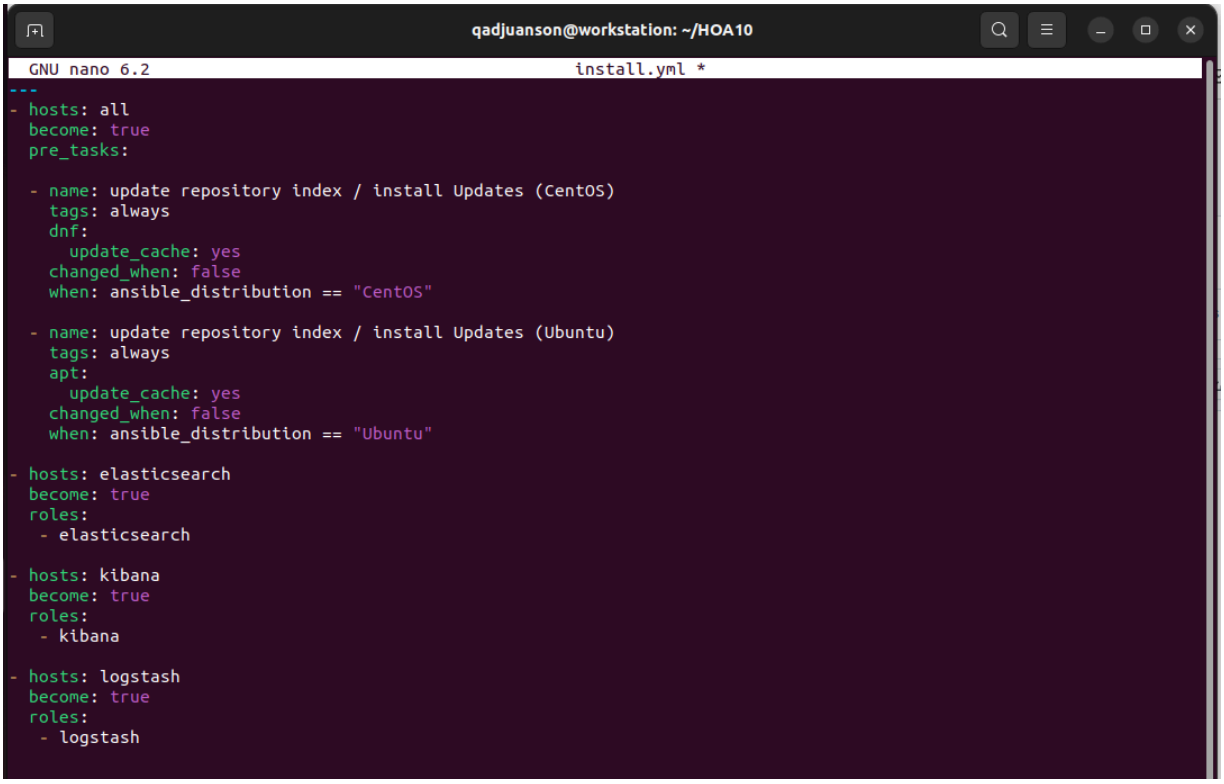
```
qadjuanson@workstation:~$ git clone git@github.com:qadjuanson/HOA10.git
Cloning into 'HOA10'...
warning: You appear to have cloned an empty repository.
qadjuanson@workstation:~$
```

Inside the HOA10, create a file named ansible.cg and inventory.

```
qadjuanson@workstation:~/HOA10$ sudo nano ansible.cfg
qadjuanson@workstation:~/HOA10$ sudo nano inventory
qadjuanson@workstation:~/HOA10$ ls
ansible.cfg  inventory
```

```
qadjuanson@workstation:~/HOA10$ cat ansible.cfg
[defaults]
inventory = inventory
remote_user = qadjuanson
host_key_checking = True
qadjuanson@workstation:~/HOA10$ cat inventory
[elasticsearch]
192.168.56.104 ansible_user=qadjuanson
[kibana]
192.168.56.102
[logstash]
192.168.56.101
```

Create a playbook named install.yml inside the HOA10.



Create a directory name roles inside the HOA10. Inside the roles create a directory named elasticsearch. Inside the directory named elasticsearch create a directory named tasks.

```
qadjuanson@workstation:~/HOA10$ mkdir roles
qadjuanson@workstation:~/HOA10$ cd roles
qadjuanson@workstation:~/HOA10/roles$ mkdir elasticsearch
qadjuanson@workstation:~/HOA10/roles$ cd elasticsearch
qadjuanson@workstation:~/HOA10/roles/elasticsearch$ mdkir tasks
```

```
qadjuanson@workstation:~/HOA10/roles/elasticsearch$ mkdir tasks
```

Inside the ~/HOA10/roles/elasticsearch/tasks, create a playbook named elasticsearch.yml.

qadjuanson@workstation: ~/HOA10/roles/elasticsearch/tasks

```
 GNU nano 6.2                                elasticsearch.yml *
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
```

Inside the ~/HOA10/roles/elasticsearch/tasks, create a playbook named main.yml.

qadjuanson@workstation: ~/HOA10/roles/elasticsearch/tasks

```
 GNU nano 6.2                                main.yml *
---
- name: Install Java
  yum:
    name: java-11-openjdk
    state: present
  when: ansible_distribution == "CentOS"

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

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

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

- name: Start Elastic Search
```

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

- name: Allow port 9200 through the firewall
  command: firewall-cmd --zone=public --add-port=9200/tcp --permanent
  register: firewall_result
  ignore_errors: true
```

Inside the directory named roles, create a directory named kibana and inside of if create a directory called tasks.

```
qadjuanson@workstation:~/HOA10/roles$ mkdir kibana
qadjuanson@workstation:~/HOA10/roles$ cd kibana
qadjuanson@workstation:~/HOA10/roles/kibana$ mkdir tasks
```

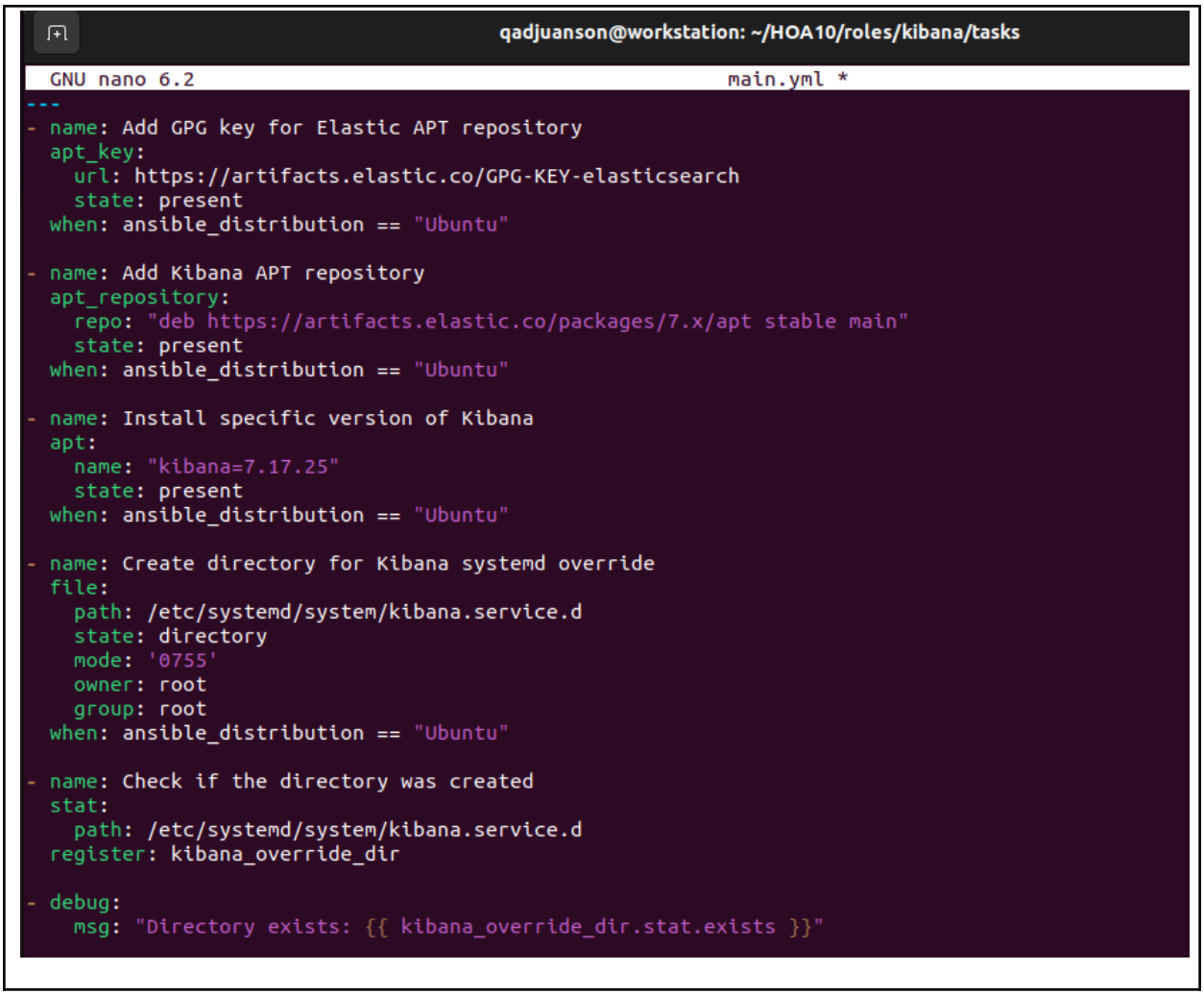
Inside the ~/HOA10/roles/kibana/tasks, create a playbook named kibana.yml.

qadjuanson@workstation: ~/HOA10/roles/kibana/tasks

GNU nano 6.2 kibana.yml *

```
# 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: "192.168.56.102"

# Set the public base URL for Kibana
server.publicBaseUrl: "http://192.168.56.102:5601"

# Elasticsearch server URL
elasticsearch.hosts: ["http://192.168.56.104:9200"]
```

Inside the ~/HOA10/roles/kibana/tasks, create a playbook named main.yml.


qadjuanson@workstation: ~/HOA10/roles/kibana/tasks
GNU nano 6.2 main.yml *

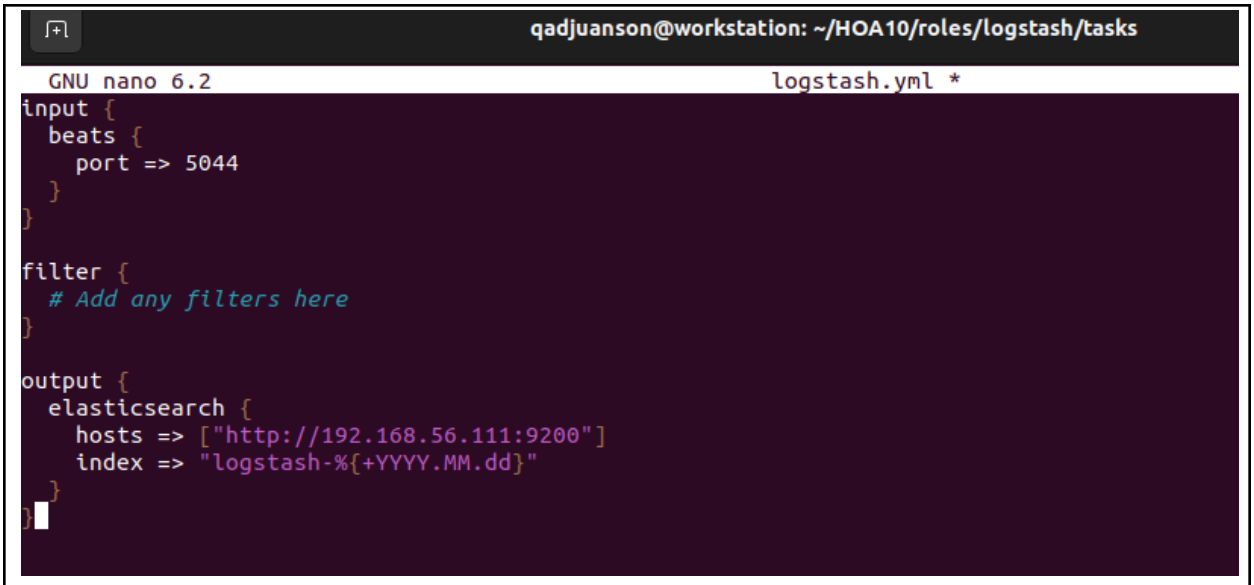
- name: Add GPG key for Elastic APT repository
apt_key:
url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
state: present
when: ansible_distribution == "Ubuntu"
- name: Add Kibana APT repository
apt_repository:
repo: "deb https://artifacts.elastic.co/packages/7.x/apt stable main"
state: present
when: ansible_distribution == "Ubuntu"
- name: Install specific version of Kibana
apt:
name: "kibana=7.17.25"
state: present
when: ansible_distribution == "Ubuntu"
- name: Create directory for Kibana systemd override
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
stat:
path: /etc/systemd/system/kibana.service.d
register: kibana_override_dir
- debug:
msg: "Directory exists: {{ kibana_override_dir.stat.exists }}"

```
  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)
  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
  template:
    src: kibana.yml
    dest: /etc/kibana/kibana.yml
  when: ansible_distribution == "Ubuntu"

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

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

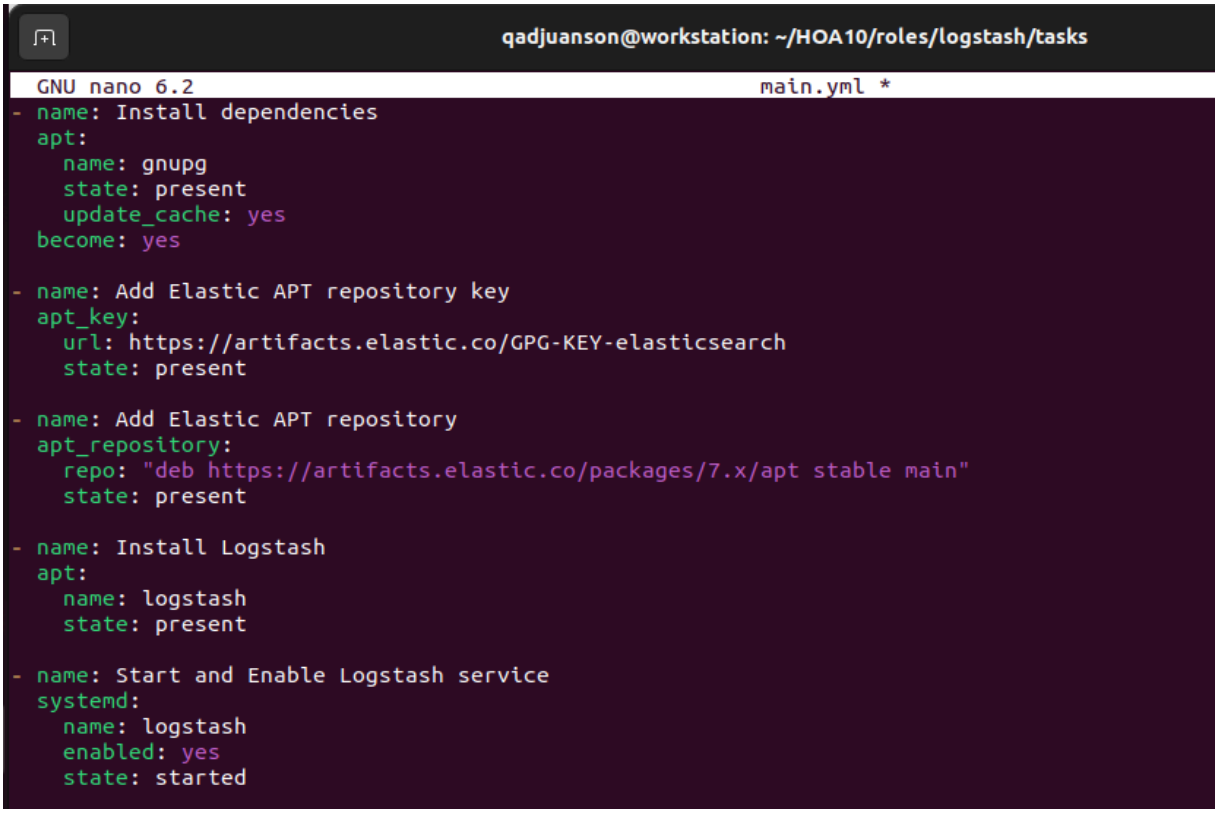
Inside the directory named roles, create a directory named logstash and inside of if create a directory called tasks.

```
qadjuanson@workstation:~/HOA10/roles$ mkdir logstash
qadjuanson@workstation:~/HOA10/roles$ cd logstash
qadjuanson@workstation:~/HOA10/roles/logstash$ mkdir tasks
```

Inside the ~/HOA10/roles/logstash/tasks, create a playbook named logstash.yml.



Inside the ~/HOA10/roles/logstash/tasks, create a playbook named main.yml.



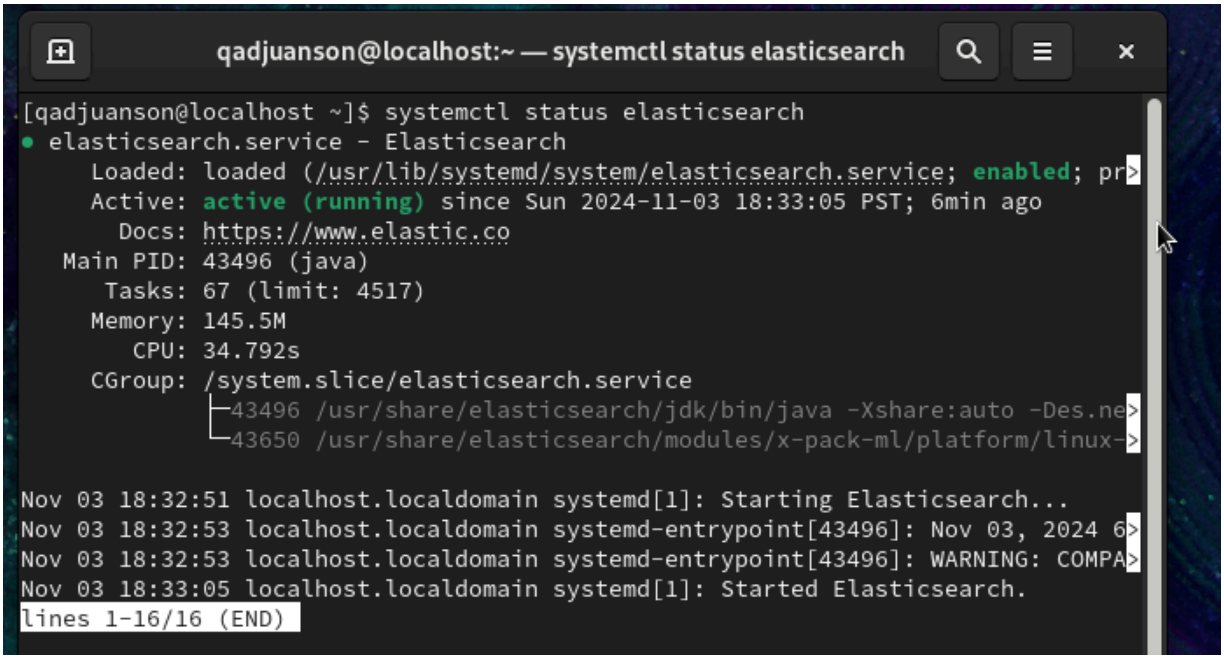
Run the playbook, *ansible-playbook install.yml --ask-become-pass.*

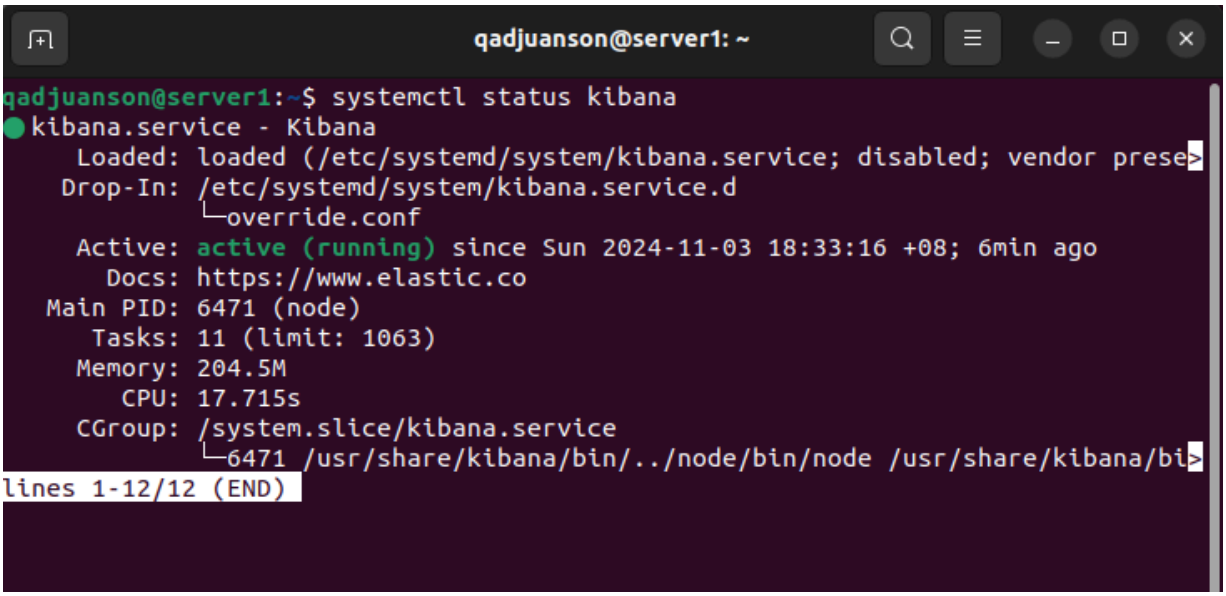
```
qadjuanson@workstation:~/HOA10$ ansible-playbook install.yml --ask-become-pass
BECOME password:

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

TASK [Gathering Facts] *********************************************************************************************
ok: [192.168.56.101]
ok: [192.168.56.102]
[DEPRECATION WARNING]: Distribution centos 9 on host 192.168.56.104 should use /usr/libexec/platform-python, but is
using /usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to
using the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.10/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [192.168.56.104]

TASK [update repository index / install Updates (CentOS)] **********************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.101]
ok: [192.168.56.104]

TASK [update repository index / install Updates (Ubuntu)] **********************************************************
skipping: [192.168.56.104]
ok: [192.168.56.101]
ok: [192.168.56.102]

PLAY [elasticsearch] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Install Java] ********************************************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Install EPEL repository] *********************************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Install Elastic Search YUM repository] *******************************************************
ok: [192.168.56.104]
```

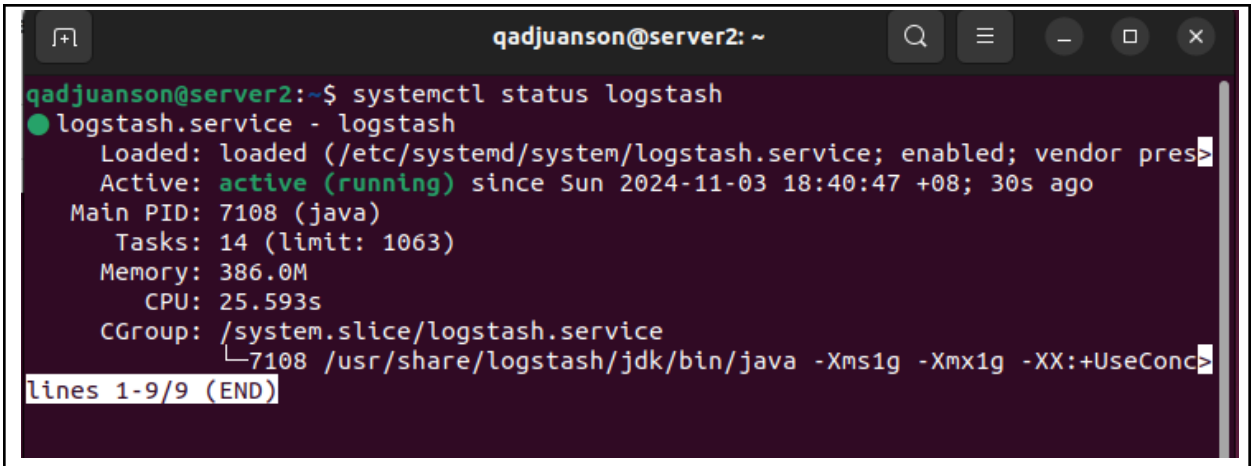
```
qadjuanson@workstation: ~/HOA10
TASK [elasticsearch : Install Elastic Search YUM repository] *******************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Install Elastic Search] **********************************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Configure Elastic Search] ********************************************************************
ok: [192.168.56.104]

TASK [elasticsearch : Start Elastic Search] ************************************************************************
changed: [192.168.56.104]

TASK [elasticsearch : Allow port 9200 through the firewall] ********************************************************
changed: [192.168.56.104]

PLAY [kibana] ******************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [192.168.56.102]

TASK [kibana : Add GPG key for Elastic APT repository] *************************************************************
ok: [192.168.56.102]

TASK [kibana : Add Kibana APT repository] **************************************************************************
ok: [192.168.56.102]

TASK [kibana : Install specific version of Kibana] *****************************************************************
ok: [192.168.56.102]

TASK [kibana : Create directory for Kibana systemd override] *******************************************************
ok: [192.168.56.102]

TASK [kibana : Check if the directory was created] *****************************************************************
ok: [192.168.56.102]

TASK [kibana : debug] **********************************************************************************************
ok: [192.168.56.102] => {
    "msg": "Directory exists: True"
}
```

```
TASK [kibana : Add GPG key for Elastic APT repository] ************************************************************
ok: [192.168.56.102]

TASK [kibana : Add Kibana APT repository] *************************************************************************
ok: [192.168.56.102]

TASK [kibana : Install specific version of Kibana] ****************************************************************
ok: [192.168.56.102]

TASK [kibana : Create directory for Kibana systemd override] ******************************************************
ok: [192.168.56.102]

TASK [kibana : Check if the directory was created] ****************************************************************
ok: [192.168.56.102]

TASK [kibana : debug] *********************************************************************************************
ok: [192.168.56.102] => {
    "msg": "Directory exists: True"
}

TASK [kibana : Create Kibana service override configuration] ******************************************************
changed: [192.168.56.102]

TASK [kibana : Configure Kibana (Setting OpenSSL Legacy Provider)] ************************************************
ok: [192.168.56.102]

TASK [kibana : Configure Kibana] **********************************************************************************
changed: [192.168.56.102]

TASK [kibana : Reload systemd] ************************************************************************************
changed: [192.168.56.102]

TASK [kibana : Enable Kibana service] *****************************************************************************
changed: [192.168.56.102]

PLAY [logstash] ***************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************
ok: [192.168.56.101]
```

```
changed: [192.168.56.102]

TASK [kibana : Configure Kibana (Setting OpenSSL Legacy Provider)] ************************************************
ok: [192.168.56.102]

TASK [kibana : Configure Kibana] **********************************************************************************
changed: [192.168.56.102]

TASK [kibana : Reload systemd] ************************************************************************************
changed: [192.168.56.102]

TASK [kibana : Enable Kibana service] *****************************************************************************
changed: [192.168.56.102]

PLAY [logstash] ***************************************************************************************************

TASK [Gathering Facts] ********************************************************************************************
ok: [192.168.56.101]

TASK [logstash : Install dependencies] ****************************************************************************
ok: [192.168.56.101]

TASK [logstash : Add Elastic APT repository key] ******************************************************************
changed: [192.168.56.101]

TASK [logstash : Add Elastic APT repository] **********************************************************************
changed: [192.168.56.101]

TASK [logstash : Install Logstash] ********************************************************************************
changed: [192.168.56.101]

TASK [logstash : Start and Enable Logstash service] ***************************************************************
changed: [192.168.56.101]

PLAY RECAP ********************************************************************************************************
192.168.56.101             : ok=8    changed=4    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
192.168.56.102             : ok=14   changed=4    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
192.168.56.104             : ok=10   changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```


qadjuanson@localhost:~ — systemctl status elasticsearch
[qadjuanson@localhost ~]$ systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; pr>
Active: active (running) since Sun 2024-11-03 18:33:05 PST; 6min ago
Docs: https://www.elastic.co
Main PID: 43496 (java)
Tasks: 67 (limit: 4517)
Memory: 145.5M
CPU: 34.792s
CGroup: /system.slice/elasticsearch.service
├─43496 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.ne>
└─43650 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux->
Nov 03 18:32:51 localhost.localdomain systemd[1]: Starting Elasticsearch...
Nov 03 18:32:53 localhost.localdomain systemd-entrypoint[43496]: Nov 03, 2024 6>
Nov 03 18:32:53 localhost.localdomain systemd-entrypoint[43496]: WARNING: COMPA>
Nov 03 18:33:05 localhost.localdomain systemd[1]: Started Elasticsearch.
lines 1-16/16 (END)


qadjuanson@server1: ~
qadjuanson@server1:~$ systemctl status kibana
● kibana.service - Kibana
Loaded: loaded (/etc/systemd/system/kibana.service; disabled; vendor prese>
Drop-In: /etc/systemd/system/kibana.service.d
└─override.conf
Active: active (running) since Sun 2024-11-03 18:33:16 +08; 6min ago
Docs: https://www.elastic.co
Main PID: 6471 (node)
Tasks: 11 (limit: 1063)
Memory: 204.5M
CPU: 17.715s
CGroup: /system.slice/kibana.service
└─6471 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bi>
lines 1-12/12 (END)



Reflections:

Answer the following:

1. What are the benefits of having a log monitoring tool?
    - Log monitoring tools are helpful for keeping systems secure, reliable, and efficient. They continuously watch for suspicious activity, like hacking attempts, and can alert your team immediately if something unusual happens. This helps protect your data and respond quickly to threats. When things go wrong, log monitoring makes it much easier to figure out what happened by collecting detailed information about errors and issues. This means you can troubleshoot problems faster and reduce downtime, which keeps everything running smoothly for users.

Conclusions:

In this activity, using Ansible to set up and manage log monitoring makes things easier, faster, and more reliable. This process organizes everything into a clear set of steps that can be repeated anytime, which is helpful if you need to add more servers or make changes.