Name: Buduan, Christian Aaron C.	**Date Performed:** 28/10/2024
Course/Section: CpE31S2	**Date Submitted:** 04/11/2024
Instructor: Engr. Robin Valenzuela	**Semester and SY:** 1st sem 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 it 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)

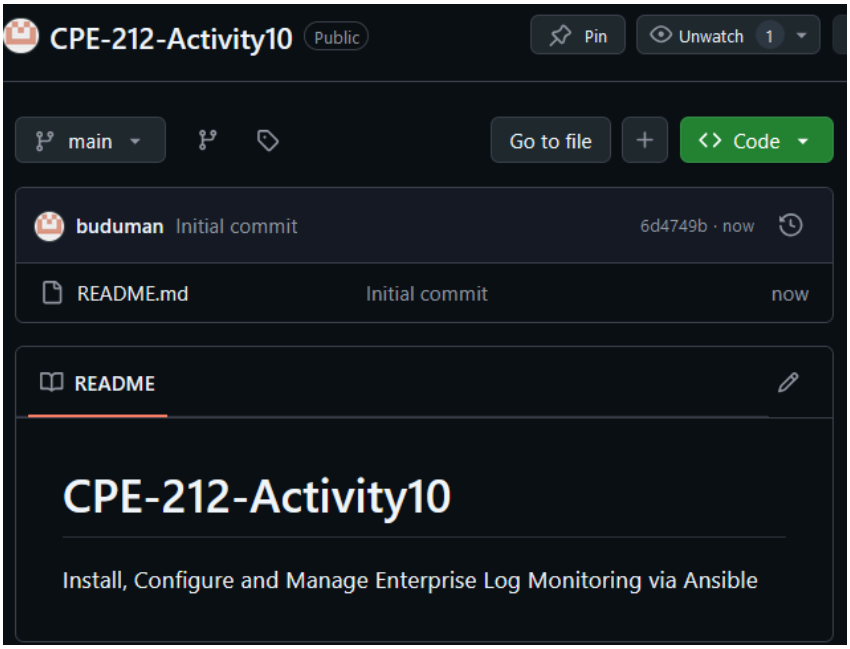

Figure 10.1 create a new github repository

```
qcacbuduan@Workstation:~$ git clone git@github.com:buduman/CPE-212-Activity10.git
Cloning into 'CPE-212-Activity10'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
qcacbuduan@Workstation:~$ ls
Buduan_PrelimExam   CPE-212-Activity6  CPE212_Buduan  examples.desktop  Templates
CPE-212-Activity10  CPE-212-Activity7  Desktop        Music             Videos
CPE-212-Activity4   CPE-212-Activity8  Documents      Pictures
CPE-212-Activity5   CPE-212-Activity9  Downloads      Public
```

Figure 10.2 clone the newly created repository into your manage node using ssh code

```
qcacbuduan@Workstation:~$ cp -r *9/roles *10
qcacbuduan@Workstation:~$ cp *9/inventory *10
qcacbuduan@Workstation:~$ cp *9/ansible.cfg *10
```

Figure 10.3 Copy the files from the previous activity to the new repository and make changes in main.yml under the tasks file

```
qcacbuduan@Workstation:~/CPE-212-Activity10$ nano act10.yml
qcacbuduan@Workstation:~/CPE-212-Activity10$ cat act10.yml
---

- hosts: all
  become: true
  pre_tasks:

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

- hosts: all
  become: true
  roles:
    -   base

- hosts: web_servers
  become: true
  roles:
    -   web_servers

- hosts: db_servers
  become: true
  roles:
    -   db_servers
```

Figure 10.4 create a new file act10.yml

```
qcacbuduan@Workstation:~/CPE-212-Activity10$ ls
act10.yml  ansible.cfg  inventory  README.md  roles
```

Figure 10.5 contents for this activity

```
qcacbuduan@Workstation:~/CPE-212-Activity10$ cat roles/web*/tasks/main.yml
---
 - name: Add GPG key for ElasticSearch (Ubuntu)
   tags: ubuntu
   apt_key:
     url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
     state: present
   when: ansible_distribution == "Ubuntu"

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

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

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

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

```
- name: Install ElasticSearch, Kibana, & LogStash
  tags: ubuntu
  package:
    name:
      - elasticsearch
      - kibana
      - logstash
    state: latest

- name: Enable ElasticSearch, Kibana, & LogStash Service
  vars:
    elastic_services:
      - elasticsearch
      - kibana
      - logstash
  service:
     name: "{{ item }}"
     enabled: yes
     state: started
  loop: "{{ elastic_services }}"
```

Figure 10.6-7 In installing the three programs In the web_servers role, which consist of an Ubuntu server, I made a playbook that consists of doing tasks such as adding a GPG key for elasticsearch, then allow the following ports to the firewall so that it would be installed properly, adding elasticsearch to the repository which is important in installing elasticsearch, make initial configurations, and finally install and enable the three programs.

```
qcacbuduan@Workstation:~/CPE-212-Activity10$ cat roles/db*/tasks/main.yml
---
 - name: Allow Port 9200 through Firewall (CentOS)
   firewalld:
     zone: public
     port: 9200/tcp
     permanent: yes
     state: enabled
     immediate: yes
   when: ansible_distribution == "CentOS"

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

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

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

```
- name: Install ElasticSearch, Kibana, & LogStash
  tags: ubuntu
  package:
    name:
      - elasticsearch
      - kibana
      - logstash
    state: latest

- name: Enable ElasticSearch, Kibana, & LogStash Service
  vars:
    elastic_services:
      - elasticsearch
      - kibana
      - logstash
  service:
     name: "{{ item }}"
     enabled: yes
     state: started
  loop: "{{ elastic_services }}"
```

Figure 10.8-9 In the db_servers where it consist of CentOS server, we start the play by enabling the port through the firewall, installing elastic search to the yum repository as well as add the GPG key, and have the same configurations as the play in web_servers. Finally, It installs the three packages and enables them.

we can now execute act10.yml using the command *ansible-playbook –ask-become-pass*

```
PLAY [web_servers] *************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [server1]

TASK [web_servers : Add GPG key for ElasticSearch (Ubuntu)] ********************************
ok: [server1]

TASK [web_servers : Allow Port 9200 through Firewall (Ubuntu)] *****************************
[WARNING]: The value 9200 (type int) in a string field was converted to u'9200' (type string). If this
does not look like what you expect, quote the entire value to ensure it does not change.
ok: [server1]

TASK [web_servers : Add ElasticSearch to repository (Ubuntu)] ******************************
ok: [server1]

TASK [web_servers : Configure ElasticSearch] ***********************************************
ok: [server1]

TASK [web_servers : Install ElasticSearch, Kibana, & LogStash] *****************************
ok: [server1]

TASK [web_servers : Enable ElasticSearch, Kibana, & LogStash Service] **********************
ok: [server1] => (item=elasticsearch)
ok: [server1] => (item=kibana)
ok: [server1] => (item=logstash)
```

FIgure 10.10 web_servers play

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

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

TASK [db_servers : Allow Port 9200 through Firewall (CentOS)] ******
ok: [centosbuduan]

TASK [db_servers : Install ElasticSearch to repository (CentOS)] ***
ok: [centosbuduan]

TASK [db_servers : Configure ElasticSearch] ************************
ok: [centosbuduan]

TASK [db_servers : Install ElasticSearch, Kibana, & LogStash] ******
ok: [centosbuduan]

TASK [db_servers : Enable ElasticSearch, Kibana, & LogStash Service]
ok: [centosbuduan] => (item=elasticsearch)
ok: [centosbuduan] => (item=kibana)
ok: [centosbuduan] => (item=logstash)
```

Figure 10.11 db_servers play

You can verify if the install was successful and if they are working properly by using *systemctl status command*

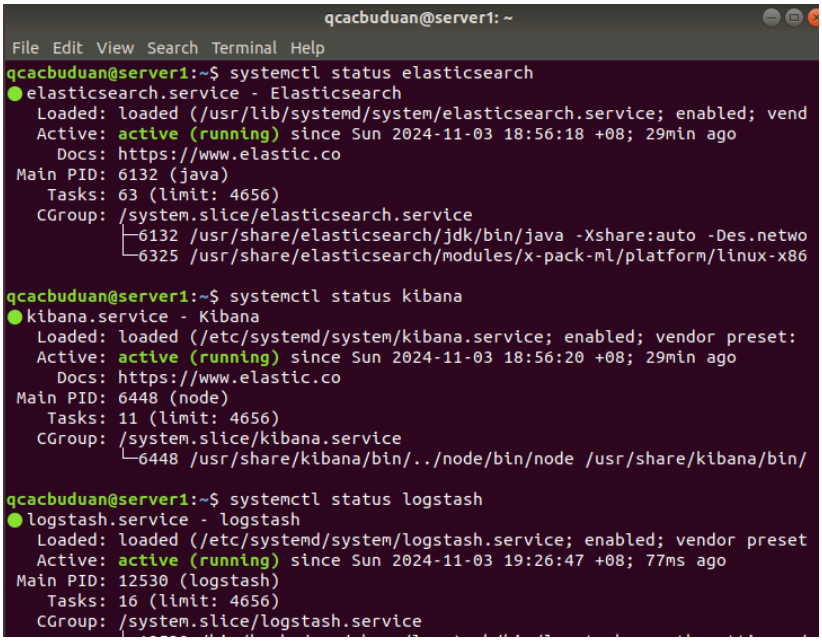

Figure 10.12 Verifying ElasticSearch, Kibana, & logstash status in Ubuntu

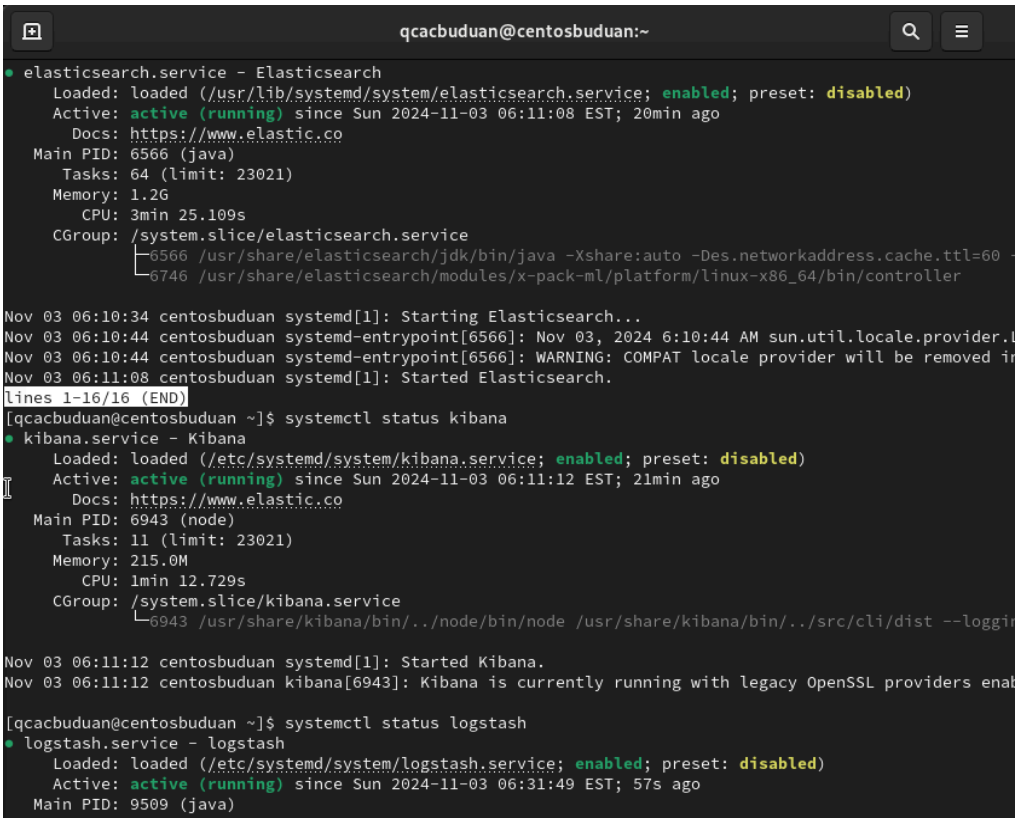

Figure 10.13 Verifying ElasticSearch, Kibana, & logstash status in CentOS

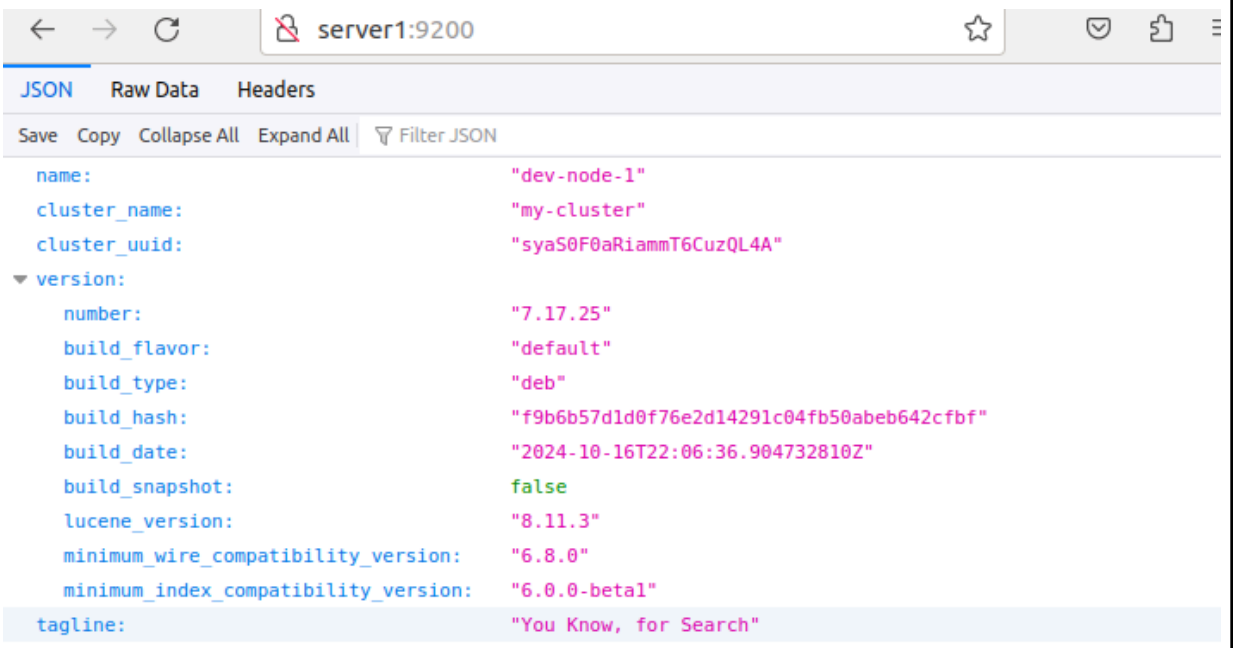

Figure 10.14 You can also verify the if they are working properly on Ubuntu in your browser by typing *[hostname]:9200* in the search bar

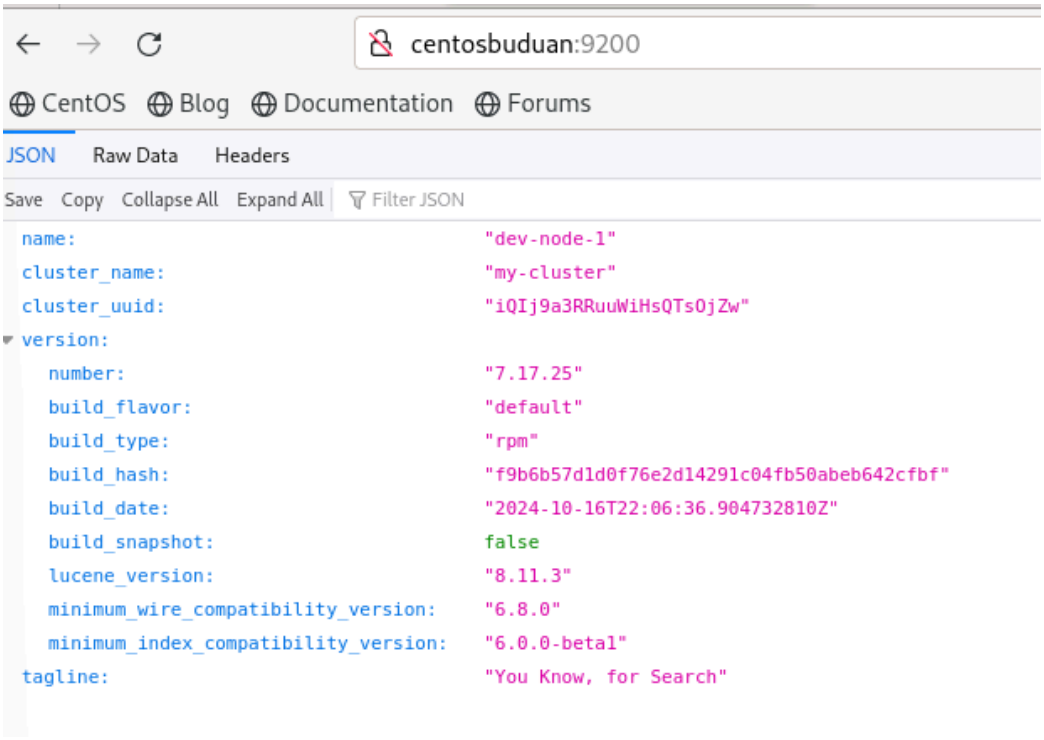

Figure 10.15 The same also applies in CentOS

```
qcacbuduan@Workstation:~/CPE-212-Activity10$ git add --all
qcacbuduan@Workstation:~/CPE-212-Activity10$ git commit -m "act10 done"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
qcacbuduan@Workstation:~/CPE-212-Activity10$ git push origin main
Everything up-to-date
qcacbuduan@Workstation:~/CPE-212-Activity10$
```

Figure 10.16 after successfully installing the required packages, you can save everything to your github repository.

My Github Repository: https://github.com/buduman/CPE-212-Activity10.git

Reflections:

Answer the following:

1. What are the benefits of having a log monitoring tool?
   - Some of the benefits you can get from having a log monitoring tool when it comes to Automating server management is that it allows real-time tracking of server activities, detecting issues quicker, apart from that, It can send you alerts, ensuring quick response in solving the issues. With the use of a monitoring tool, you can have access to the log data which can provide you with more clues for diagnosing problems and issues, as well as detect security breaches, enhancing the security of your servers.

Conclusions:

- In this activity, I was able to learn the importance of a log monitoring tool such as ElasticSearch along with Kibana and LogStash and how it benefits in managing servers. 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. which helps us detect and diagnose any issues that occur in any of our managed nodes. When installing ElasticSearch, apart from having to open ports through the firewall, adding the GPG key is also needed to be able to install ElasticSearch and enable the service without any issues.