Name: Bernardo, Christian Emmanuel	**Date Performed: 02/04/2024**
Course/Section: CPE 232-CPE31S1	**Date Submitted: 02/04/2024**
Instructor: Dr. Jonathan Taylar	**Semester and SY: 2nd, 2023-2024**

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.

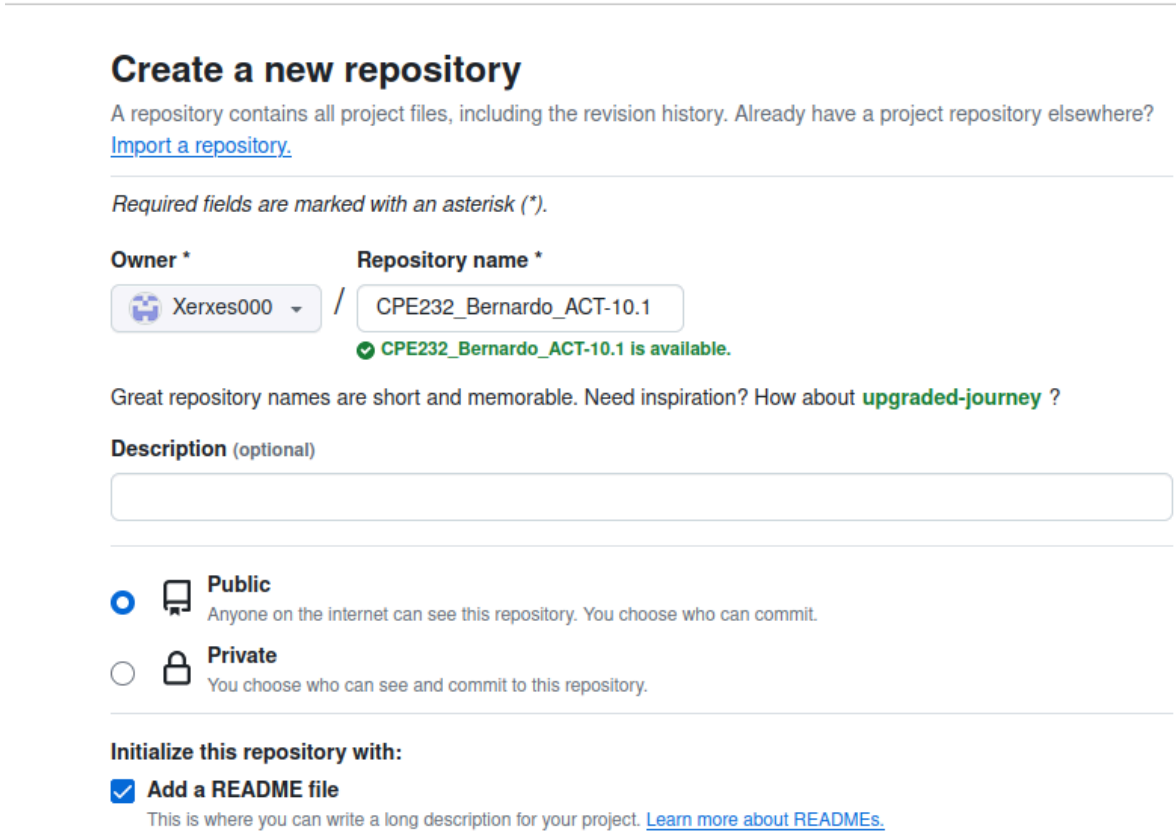
Source: https://www.graylog.org/products/open-source

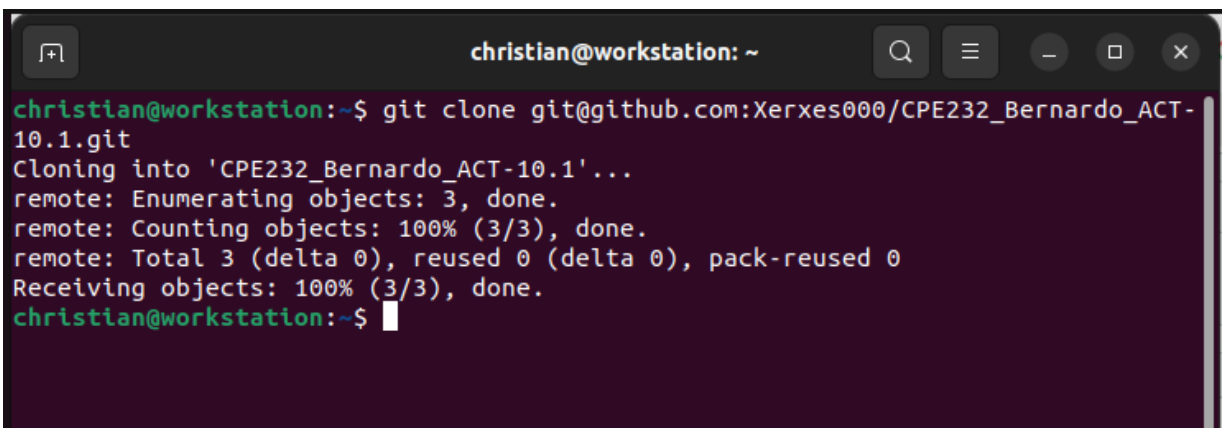
3. Tasks

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)

step one: Setting up the repository for this activity





these two steps are making the repository and cloning the repository in the workstation

```
christian@workstation: ~/CPE232_Bernardo_ACT-10.1
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ansible.cfg
[sudo] password for christian:
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ansible.cfg
[defaults]

inventory = inventory
host_key_checking = false

deprecation_warnings = false

remote_user = christian
private_key_files = ~/.ssh/id_rsa
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the ansible.cfg for the activity

step two: making the inventory file

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano inventory
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat inventory
[UbuntuServer]
192.168.56.109

[CentOSServer]
192.168.56.113
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ tree
.
├── ansible.cfg
├── inventory
└── README.md

0 directories, 3 files
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the inventory file

step three: making the directories

```
christian@workstation: ~/CPE232_Bernardo_ACT-10.1
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ mkdir roles
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cd roles
christian@workstation:~/CPE232_Bernardo_ACT-10.1/roles$ mkdir CentOS Ubuntu
christian@workstation:~/CPE232_Bernardo_ACT-10.1/roles$ mkdir ./CentOS/tasks
christian@workstation:~/CPE232_Bernardo_ACT-10.1/roles$ mkdir ./Ubuntu/tasks
christian@workstation:~/CPE232_Bernardo_ACT-10.1/roles$ cd ..
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ tree
.
├── ansible.cfg
├── inventory
├── README.md
└── roles
    ├── CentOS
    │   └── tasks
    └── Ubuntu
        └── tasks

5 directories, 3 files
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the newly made directories including: roles, CentOS, Ubuntu and tasks for each

step four: making elst.yml it stands for installation of Elastic Stack

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano elst.yml
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat elst.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: CentOSServer
  become: true
  roles:
    - CentOS

- hosts: UbuntuServer
  become: true
  roles:
    - Ubuntu
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

step five: the main.yml files for Ubuntu and CentOS

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/tasks/main.yml
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/tasks/main.yml
# Elastic Search Setup

- name: Termporarily setting the SELINUX of CentOS remote server to permissive
  selinux:
    policy: targeted
    state: permissive
  when: ansible_os_family == 'RedHat'

- name: Updating sysctl for max_map_count
  sysctl:
    name: vm.max_map_count
    value: "262144"
    sysctl_set: yes

- name: Adding the user 'elasticsearch'
  user:
    name: elasticsearch
    comment: elasticsearch user

- name: Creating directory for the downloaded files
  file:
    path: /data
    state: directory
    mode: 0777
```

```
- name: Extracting elasticsearch
  unarchive:
    src: /data/elasticsearch-6.8.15.tar.gz
    dest: /data/
    remote_src: yes
    creates: /data/elasticsearch-6.8.15/config/elasticsearch.yml

- name: Inserting the Elastic Search systemd service unit file
  template:
    src: elasticsearch.service.j2
    dest: /etc/systemd/system/elasticsearch.service
    mode: 0644

- name: Inserting the Elastic Search configuration template
  template:
    src: elasticsearch.yml.j2
    dest: /data/elasticsearch-6.8.15/config/elasticsearch.yml
    mode: 0660

- file:
    path: /data/elasticsearch-6.8.15
    owner: elasticsearch
    group: elasticsearch
    recurse: yes

- name: Daemon Reload
  systemd:
    daemon_reload: yes
```

```
  - name: Starting the Elastic Search service
    service:
      name: elasticsearch
      state: started
      enabled: yes

#Kibana Installation and Configuration

  - name: Creating directory for downloaded files
    file:
      path: /data
      state: directory
      mode: 0777

  - name: Installing Kibana tar
    get_url:
      url: https://artifacts.elastic.co/downloads/kibana/kibana-6.8.15-linux-x86_64.tar.gz
      dest: /data/kibana-6.8.15-linux-x86_64.tar.gz
      mode: 0755

  - name: Extracting Kibana
    unarchive:
      src: /data/kibana-6.8.15-linux-x86_64.tar.gz
      dest: /data/
      remote_src: yes
      creates: /data/kibana-6.8.15-linux-x86_64/config/kibana.yml
```

```
  - name: Inserting the Kibana systemd service unit file
    template:
      src: kibana.service.j2
      dest: /etc/systemd/system/kibana.service
      mode: 0644

  - name: Inserting the update of configration template for Kibana
    template:
      src: kibana.yml.j2
      dest: /data/kibana-6.8.15-linux-x86_64/config/kibana.yml
      mode: 0660

  - name: Daemon Reload
    systemd:
      daemon_reload: yes

  - name: Starting the Kibana service
    service:
      name: kibana
      state: started
      enabled: yes

# Logstash Setup
  - name: Creating directory for downloaded files
    file:
      path: /data
      state: directory
      mode: "u=rwx,g=rwx,o=rwx"
```

```
- name: Installing logstash tar ball
  get_url:
    url: https://artifacts.elastic.co/downloads/logstash/logstash-6.8.15.tar.gz
    dest: /data/logstash-6.8.15.tar.gz
    mode: 0755

- name: Extracting logstash
  unarchive:
    src: /data/logstash-6.8.15.tar.gz
    dest: /data/
    remote_src: yes
    creates: /data/logstash-6.8.15/conf.d/inputs.conf

- name: Inserting the Logstash systemd service unit file
  template:
    src: logstash.service.j2
    dest: /etc/systemd/system/logstash.service
    mode: 0644

- name: Script of logstash for starting/stopping
  template:
    src: start.sh.j2
    dest: /data/logstash-6.8.15/start.sh
    mode: 0754
```

```
- name: Creating /data/logstash-6.8.15/conf.d directory
  file:
    path: /data/logstash-6.8.15/conf.d
    state: directory
    mode: 0777

- name: Updating the configuration default of logstash
  template:
    src: inputs.conf.j2
    dest: /data/logstash-6.8.15/conf.d/inputs.conf
    mode: 0660

- name: Daemon Reload
  systemd:
    daemon_reload: yes

- name: Starting the Logstash service
  service:
    name: logstash
    state: started
    enabled: yes
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the CentOS main.yml

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/tasks/main.yml
# Elastic Search Setup

  - name: Termporarily setting the SELINUX of Ubuntu remote server to permissive
    selinux:
      policy: targeted
      state: permissive
    when: ansible_os_family == 'Ubuntu'

  - name: Updating sysctl for max_map_count
    sysctl:
      name: vm.max_map_count
      value: "262144"
      sysctl_set: yes

  - name: Adding the user 'elasticsearch'
    user:
      name: elasticsearch
      comment: elasticsearch user

  - name: Creating directory for the downloaded files
    file:
      path: /data
      state: directory
      mode: 0777

  - name: Downloading elasticsearch tar ball
    get_url:
      url: https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.8.15.tar.gz
      dest: /data/elasticsearch-6.8.15.tar.gz
      mode: 0755
```

```
  - name: Extracting elasticsearch
    unarchive:
      src: /data/elasticsearch-6.8.15.tar.gz
      dest: /data/
      remote_src: yes
      creates: /data/elasticsearch-6.8.15/config/elasticsearch.yml

  - name: Inserting the Elastic Search systemd service unit file
    template:
      src: elasticsearch.service.j2
      dest: /etc/systemd/system/elasticsearch.service
      mode: 0644

  - name: Inserting the Elastic Search configuration template
    template:
      src: elasticsearch.yml.j2
      dest: /data/elasticsearch-6.8.15/config/elasticsearch.yml
      mode: 0660

  - file:
      path: /data/elasticsearch-6.8.15
      owner: elasticsearch
      group: elasticsearch
      recurse: yes

  - name: Daemon Reload
    systemd:
      daemon_reload: yes
```

```
  - name: Starting the Elastic Search service
    service:
      name: elasticsearch
      state: started
      enabled: yes

#Kibana Installation and Configuration

  - name: Creating directory for downloaded files
    file:
      path: /data
      state: directory
      mode: 0777

  - name: Installing Kibana tar
    get_url:
      url: https://artifacts.elastic.co/downloads/kibana/kibana-6.8.15-linux-x86_64.tar.gz
      dest: /data/kibana-6.8.15-linux-x86_64.tar.gz
      mode: 0755

  - name: Extracting Kibana
    unarchive:
      src: /data/kibana-6.8.15-linux-x86_64.tar.gz
      dest: /data/
      remote_src: yes
      creates: /data/kibana-6.8.15-linux-x86_64/config/kibana.yml
```

```
  - name: Inserting the Kibana systemd service unit file
    template:
      src: kibana.service.j2
      dest: /etc/systemd/system/kibana.service
      mode: 0644

  - name: Inserting the update of configration template for Kibana
    template:
      src: kibana.yml.j2
      dest: /data/kibana-6.8.15-linux-x86_64/config/kibana.yml
      mode: 0660

  - name: Daemon Reload
    systemd:
      daemon_reload: yes

  - name: Starting the Kibana service
    service:
      name: kibana
      state: started
      enabled: yes

# Logstash Setup
  - name: Creating directory for downloaded files
    file:
      path: /data
      state: directory
      mode: "u=rwx,g=rwx,o=rwx"
```

```
- name: Installing logstash tar ball
  get_url:
    url: https://artifacts.elastic.co/downloads/logstash/logstash-6.8.15.tar.gz
    dest: /data/logstash-6.8.15.tar.gz
    mode: 0755

- name: Extracting logstash
  unarchive:
    src: /data/logstash-6.8.15.tar.gz
    dest: /data/
    remote_src: yes
    creates: /data/logstash-6.8.15/conf.d/inputs.conf

- name: Inserting the Logstash systemd service unit file
  template:
    src: logstash.service.j2
    dest: /etc/systemd/system/logstash.service
    mode: 0644

- name: Script of logstash for starting/stopping
  template:
    src: start.sh.j2
    dest: /data/logstash-6.8.15/start.sh
    mode: 0754
```

```
- name: Creating /data/logstash-6.8.15/conf.d directory
  file:
    path: /data/logstash-6.8.15/conf.d
    state: directory
    mode: 0777

- name: Updating the configuration default of logstash
  template:
    src: inputs.conf.j2
    dest: /data/logstash-6.8.15/conf.d/inputs.conf
    mode: 0660

- name: Daemon Reload
  systemd:
    daemon_reload: yes

- name: Starting the Logstash service
  service:
    name: logstash
    state: started
    enabled: yes
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the Ubuntu main.yml

step six: Including the required dependencies for Elastic Stack (Elastic Search, Kibana, Logstash)

this is the CentOS dependencies

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/elasticsearch.service.j2
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/elasticsearch.service.j2
[Unit]
Description=Elasticsearch service
After=network.target

[Service]
Type=simple
LimitNOFILE=65536
LimitMEMLOCK=infinity
User=elasticsearch
Group=elasticsearch
ExecStart=/data/elasticsearch-6.8.15/bin/elasticsearch

[Install]
WantedBy=multi-user.target
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/elasticsearch.yml.j2
http.port: 9200
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/inputs.conf.j2
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/inputs.conf.j2
input {
  beats {
    port => 5044
  }
}

filter {
  mutate {
    remove_field => [ '[host]' ]
  }
}

filter {
  mutate {
    convert => { "[system][process][cpu][total][norm][pct]" => "float" }
    convert => { "[system][diskio][iostat][request][avg_size]" => "float" }
    convert => { "[system][process][memory][rss][pct]" => "float" }
    convert => { "[system][process][cpu][total][pct]" => "float" }
    convert => { "[system][diskio][iostat][queue][avg_size]" => "float" }
    convert => { "[system][core][steal][pct]" => "float" }
    convert => { "[system][cpu][steal][pct]" => "float" }
    convert => { "[system][diskio][iostat][service_time]" => "float" }
  }
}

output {
  elasticsearch {
    hosts => "localhost:9200"
    index => 'logstash-daily-%{+YYYY.MM.dd}'
  }
}

christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/kibana.service.j2
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/kibana.service.j2
[Unit]
Description=Kibana service
After=network.target

[Service]
Type=simple
ExecStart=/data/kibana-6.8.15-linux-x86_64/bin/kibana

[Install]
WantedBy=multi-user.target
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/kibana.yml.j2
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat  ./roles/CentOS/kibana.yml.j2
server.port: 5601
server.host: 0.0.0.0
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ sudo nano ./roles/CentOS/logstash.service.j2
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/logstash.service.j2
[Unit]
Description=Logstash service
After=network.target

[Service]
Type=simple
ExecStart=/data/logstash-6.8.15/start.sh

[Install]
WantedBy=multi-user.target
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

this is the Ubuntu dependencies

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/elasticsearch.service.j2
[Unit]
Description=Elasticsearch service
After=network.target

[Service]
Type=simple
LimitNOFILE=65536
LimitMEMLOCK=infinity
User=elasticsearch
Group=elasticsearch
ExecStart=/data/elasticsearch-6.8.15/bin/elasticsearch

[Install]
WantedBy=multi-user.target
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/elasticsearch.yml.j2
http.port: 9200
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/inputs.conf.j2
input {
  beats {
    port => 5044
  }
}

filter {
  mutate {
    remove_field => [ '[host]' ]
  }
}

filter {
  mutate {
    convert => { "[system][process][cpu][total][norm][pct]" => "float" }
    convert => { "[system][diskio][iostat][request][avg_size]" => "float" }
    convert => { "[system][process][memory][rss][pct]" => "float" }
    convert => { "[system][process][cpu][total][pct]" => "float" }
    convert => { "[system][diskio][iostat][queue][avg_size]" => "float" }
    convert => { "[system][core][steal][pct]" => "float" }
    convert => { "[system][cpu][steal][pct]" => "float" }
    convert => { "[system][diskio][iostat][service_time]" => "float" }
  }
}

output {
  elasticsearch {
    hosts => "localhost:9200"
    index => 'logstash-daily-%{+YYYY.MM.dd}'
  }
}
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/kibana.service.j2
[Unit]
Description=Kibana service
After=network.target

[Service]
Type=simple
ExecStart=/data/kibana-6.8.15-linux-x86_64/bin/kibana

[Install]
WantedBy=multi-user.target
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat  ./roles/Ubuntu/kibana.yml.j2
server.port: 5601
server.host: 0.0.0.0
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/Ubuntu/start.sh.j2
#!/bin/bash

/data/logstash-6.8.15/bin/logstash -f /data/logstash-6.8.15/conf.d/inputs.conf
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ cat ./roles/CentOS/start.sh.j2
#!/bin/bash

/data/logstash-6.8.15/bin/logstash -f /data/logstash-6.8.15/conf.d/inputs.conf
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

step seven: executing the playbook

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ ansible-playbook --ask-become-pass elst.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.56.113]
fatal: [192.168.56.109]: UNREACHABLE! => {"changed": false, "msg": "Failed to connect to the host via ssh: ssh:
 connect to host 192.168.56.109 port 22: No route to host", "unreachable": true}

TASK [update repository index (CentOS)] ****************************************************************
ok: [192.168.56.113]

TASK [install updates (Ubuntu)] ************************************************************************
skipping: [192.168.56.113]

PLAY [CentOSServer] ************************************************************************************

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

TASK [CentOS : Termporarily setting the SELINUX of CentOS remote server to permissive] ***********************
changed: [192.168.56.113]

TASK [CentOS : Updating sysctl for max_map_count] ******************************************************
changed: [192.168.56.113]

TASK [CentOS : Adding the user 'elasticsearch'] ********************************************************
changed: [192.168.56.113]

TASK [CentOS : Creating directory for the downloaded files] ********************************************
changed: [192.168.56.113]

TASK [CentOS : Downloading elasticsearch tar ball] *****************************************************
changed: [192.168.56.113]
```

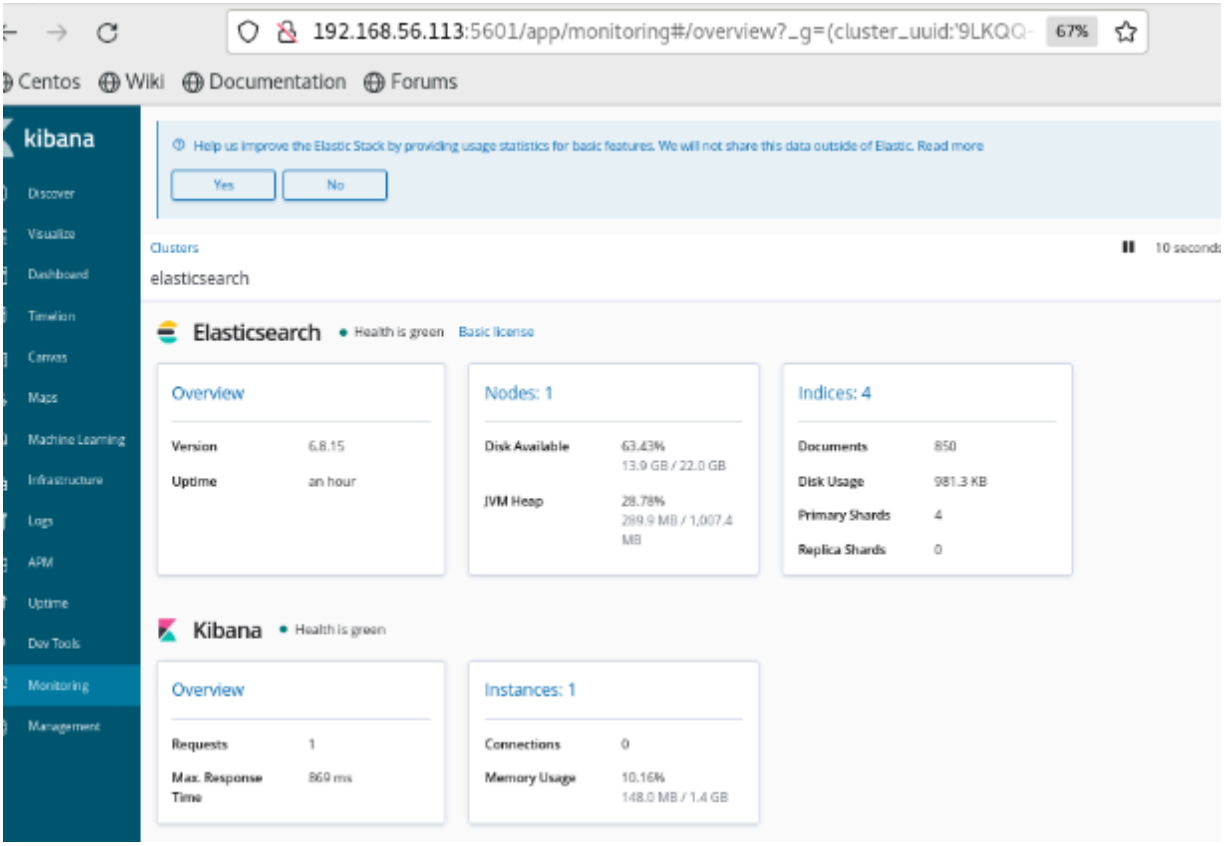
```
TASK [CentOS : Extracting elasticsearch] ***********************************************************************
changed: [192.168.56.113]

TASK [CentOS : Inserting the Elastic Search systemd service unit file] ****************************************
changed: [192.168.56.113]

TASK [CentOS : Inserting the Elastic Search configuration template] *******************************************
changed: [192.168.56.113]

TASK [CentOS : file] *******************************************************************************************
changed: [192.168.56.113]

TASK [CentOS : Daemon Reload] ******************************************************************************
ok: [192.168.56.113]

TASK [CentOS : Starting the Elastic Search service] ********************************************************
changed: [192.168.56.113]

TASK [CentOS : Creating directory for downloaded files] ****************************************************
ok: [192.168.56.113]

TASK [CentOS : Installing Kibana tar] ***********************************************************************
changed: [192.168.56.113]

TASK [CentOS : Extracting Kibana] ***************************************************************************
changed: [192.168.56.113]

TASK [CentOS : Inserting the Kibana systemd service unit file] *********************************************
changed: [192.168.56.113]

TASK [CentOS : Inserting the update of configration template for Kibana] ************************************
changed: [192.168.56.113]

TASK [CentOS : Daemon Reload] ******************************************************************************
ok: [192.168.56.113]
```

```
TASK [CentOS : Starting the Kibana service] ********************************************************************
changed: [192.168.56.113]

TASK [CentOS : Creating directory for downloaded files] ****************************************************
ok: [192.168.56.113]

TASK [CentOS : Installing logstash tar ball] ***********************************************************************
changed: [192.168.56.113]

TASK [CentOS : Extracting logstash] ********************************************************************************
changed: [192.168.56.113]

TASK [CentOS : Inserting the Logstash systemd service unit file] ***************************************************
changed: [192.168.56.113]

TASK [CentOS : Script of logstash for starting/stopping] ***********************************************************
changed: [192.168.56.113]
```

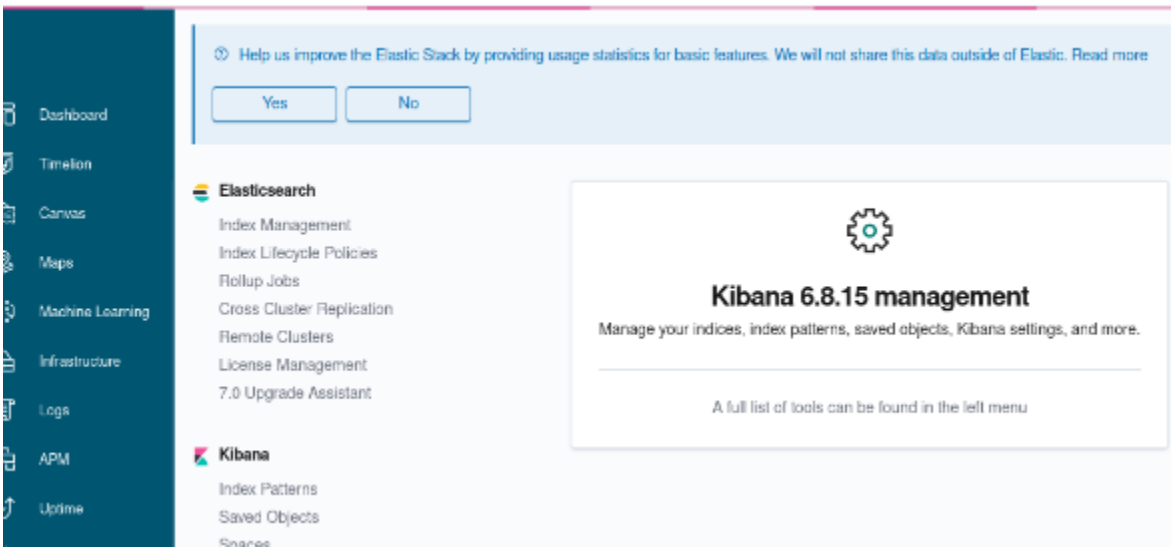
```
TASK [CentOS : Creating /data/logstash-6.8.15/conf.d directory] ********************************************
changed: [192.168.56.113]

TASK [CentOS : Updating the configuration default of logstash] *********************************************
changed: [192.168.56.113]

TASK [CentOS : Daemon Reload] ******************************************************************************
ok: [192.168.56.113]

TASK [CentOS : Starting the Logstash service] **************************************************************
changed: [192.168.56.113]

PLAY [UbuntuServer] ****************************************************************************************

PLAY RECAP *************************************************************************************************
192.168.56.109             : ok=0    changed=0    unreachable=1    failed=0    skipped=0    rescued=0    ignore
d=0
192.168.56.113             : ok=30   changed=22   unreachable=0    failed=0    skipped=1    rescued=0    ignore
d=0

christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```

step eight: testing the software

this is the CentOS testing



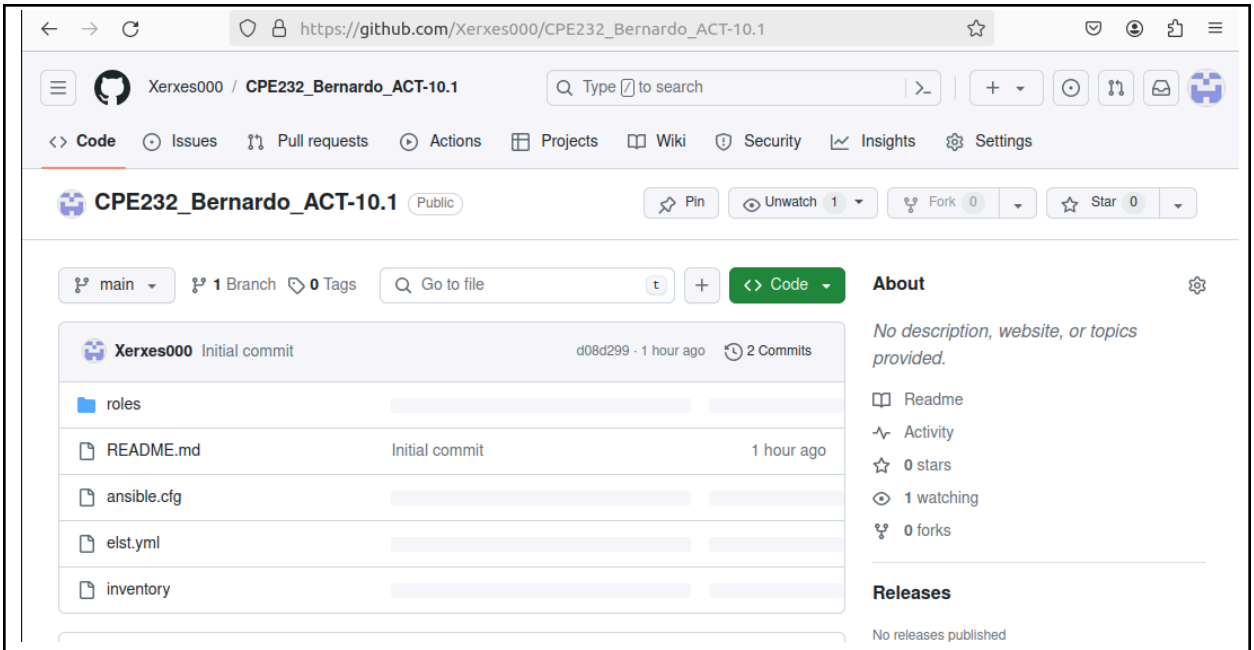
this is the Ubuntu Testing



step nine: git pushing the files to the repositories

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible.cfg
        elst.yml
        inventory
        roles/

nothing added to commit but untracked files present (use "git add" to track)
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ git add *
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ git commit -m "update"
[main 85de48b] update
 19 files changed, 509 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 elst.yml
 create mode 100644 inventory
 create mode 100644 roles/CentOS/elasticsearch.service.j2
 create mode 100644 roles/CentOS/elasticsearch.yml.j2
 create mode 100644 roles/CentOS/inputs.conf.j2
 create mode 100644 roles/CentOS/kibana.service.j2
 create mode 100644 roles/CentOS/kibana.yml.j2
 create mode 100644 roles/CentOS/logstash.service.j2
 create mode 100644 roles/CentOS/start.sh.j2
 create mode 100644 roles/CentOS/tasks/main.yml
 create mode 100644 roles/Ubuntu/elasticsearch.service.j2
 create mode 100644 roles/Ubuntu/elasticsearch.yml.j2
 create mode 100644 roles/Ubuntu/inputs.conf.j2
 create mode 100644 roles/Ubuntu/kibana.service.j2
 create mode 100644 roles/Ubuntu/kibana.yml.j2
 create mode 100644 roles/Ubuntu/logstash.service.j2
 create mode 100644 roles/Ubuntu/start.sh.j2
 create mode 100644 roles/Ubuntu/tasks/main.yml
```

```
christian@workstation:~/CPE232_Bernardo_ACT-10.1$ git push origin main
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Compressing objects: 100% (16/16), done.
Writing objects: 100% (20/20), 3.07 KiB | 629.00 KiB/s, done.
Total 20 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
To github.com:Xerxes000/CPE232_Bernardo_ACT-10.1.git
   d08d299..85de48b  main -> main
christian@workstation:~/CPE232_Bernardo_ACT-10.1$
```



- **Github Link: https://github.com/Xerxes000/CPE232_Bernardo_ACT-10.1.git**

Reflections:

Answer the following:

1. What are the benefits of having log monitoring tool?
   - Workload logs are gathered and centralized logging capabilities are provided by the log monitoring tool. Logs from complicated cloud environments can be combined into a searchable index by using certain log monitoring tools, like the ELK stack. User and network administrators can carry out root cause analysis and enhance security with these features.

Conclusions:

- To summarize, this exercise offered significant understanding into the playbook-based installation and setup of the Elastic Stack, which consists of Kibana, Logstash, and Elastic Search. My practical experience in managing these crucial components came from implementing roles to consolidate playbooks. Creating useful Ansible playbooks that can be used for a variety of tasks was another new skill I picked up.I also gained a deeper understanding of playbook design—more specifically, log monitoring tool design—thanks to this exercise. I now recognize the convenience with which playbooks make the

installation, configuration, and use of these tools possible. And finally, I realized how much benefit network administrators, users, and companies can receive from log monitoring tools. This increases overall system reliability by improving security and facilitating effective root cause analysis.