Name: Andaya, Lyka C	**Date Performed: November 13, 2023**
Course/Section: CPE31S4	**Date Submitted: November 14, 2023**
Instructor: Dr.Taylar	**Semester and SY: 2023-2024**

Activity 11: Containerization

1. Objectives

Create a Dockerfile and form a workflow using Ansible as Infrastructure as Code (IaC) to enable Continuous Delivery process

2. Discussion

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

Source: https://docs.docker.com/get-started/overview/

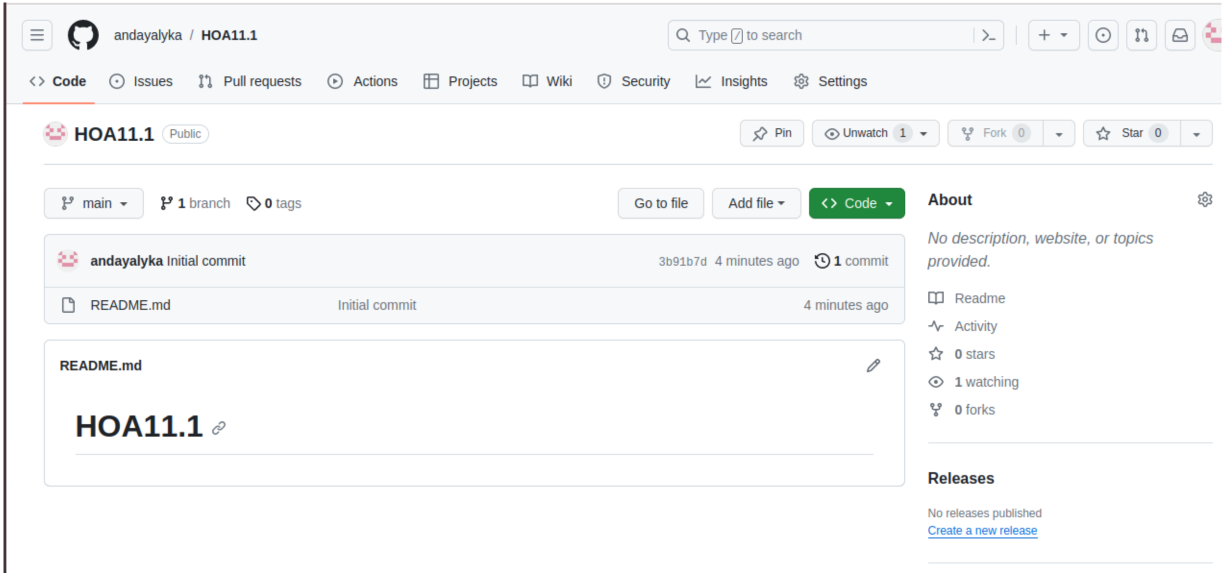
You may also check the difference between containers and virtual machines. Click the link given below.

Source: https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm

3. Tasks

1. Create a new repository for this activity.
2. Install Docker and enable the docker socket.
3. Add to Docker group to your current user.
4. Create a Dockerfile to install web and DB server.
5. Install and build the Dockerfile using Ansible.
6. Add, commit and push it to your repository.

4. Output (screenshots and explanations)



HOA11.1 Public

andayalyka Initial commit 3b91b7d 4 minutes ago 1 commit

README.md Initial commit 4 minutes ago

README.md

HOA11.1

About

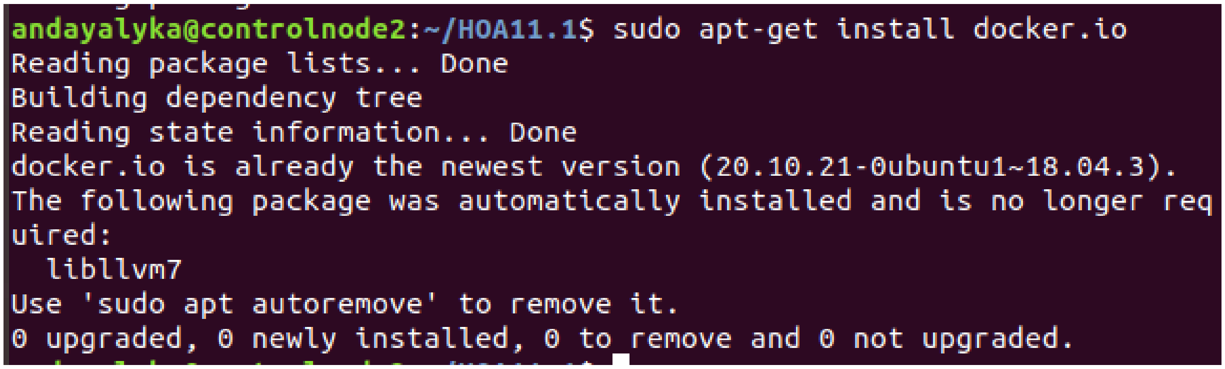
No description, website, or topics provided.

Readme
Activity
0 stars
1 watching
0 forks

Releases

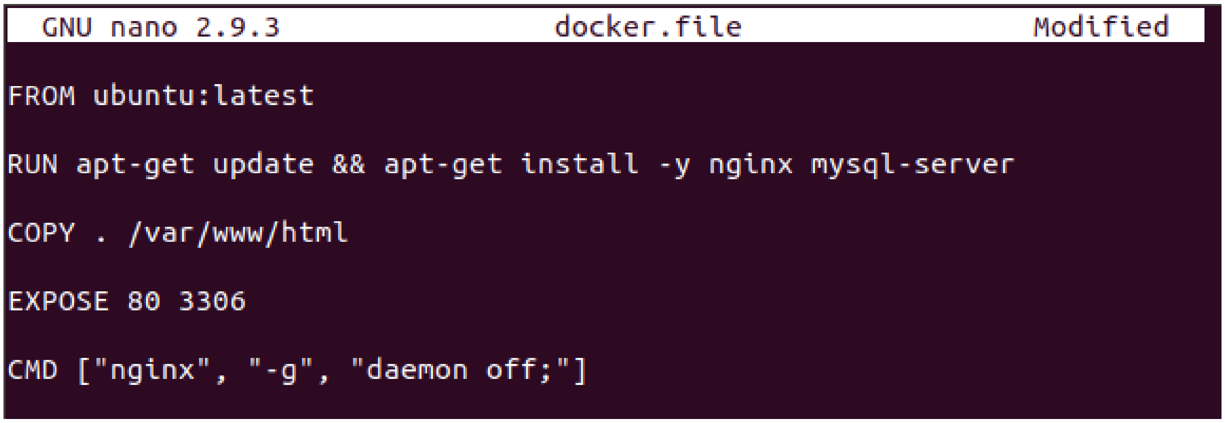
No releases published
Create a new release

Explanation: New created repository named as HOA11.1



```
andayalyka@controlnode2:~/HOA11.1$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
docker.io is already the newest version (20.10.21-0ubuntu1~18.04.3).
The following package was automatically installed and is no longer req
uired:
  libllvm7
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

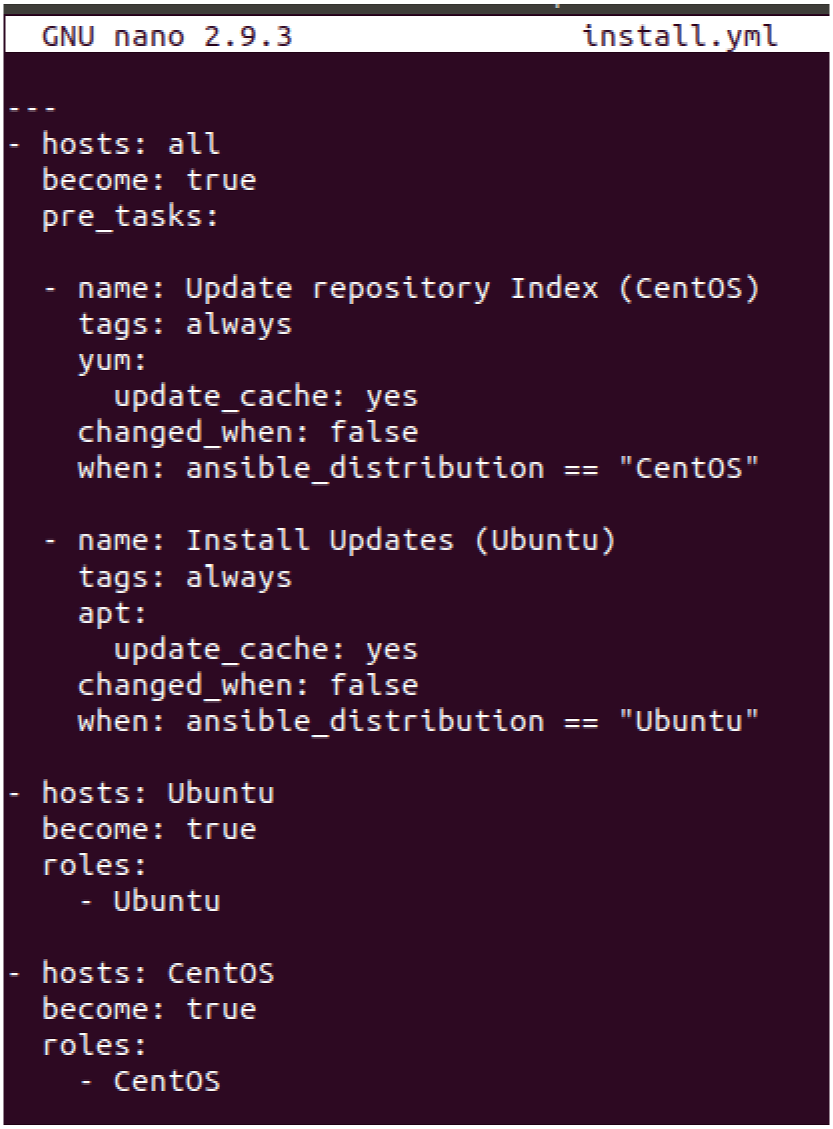
Explanation: The created repository has been cloned in the virtual machine



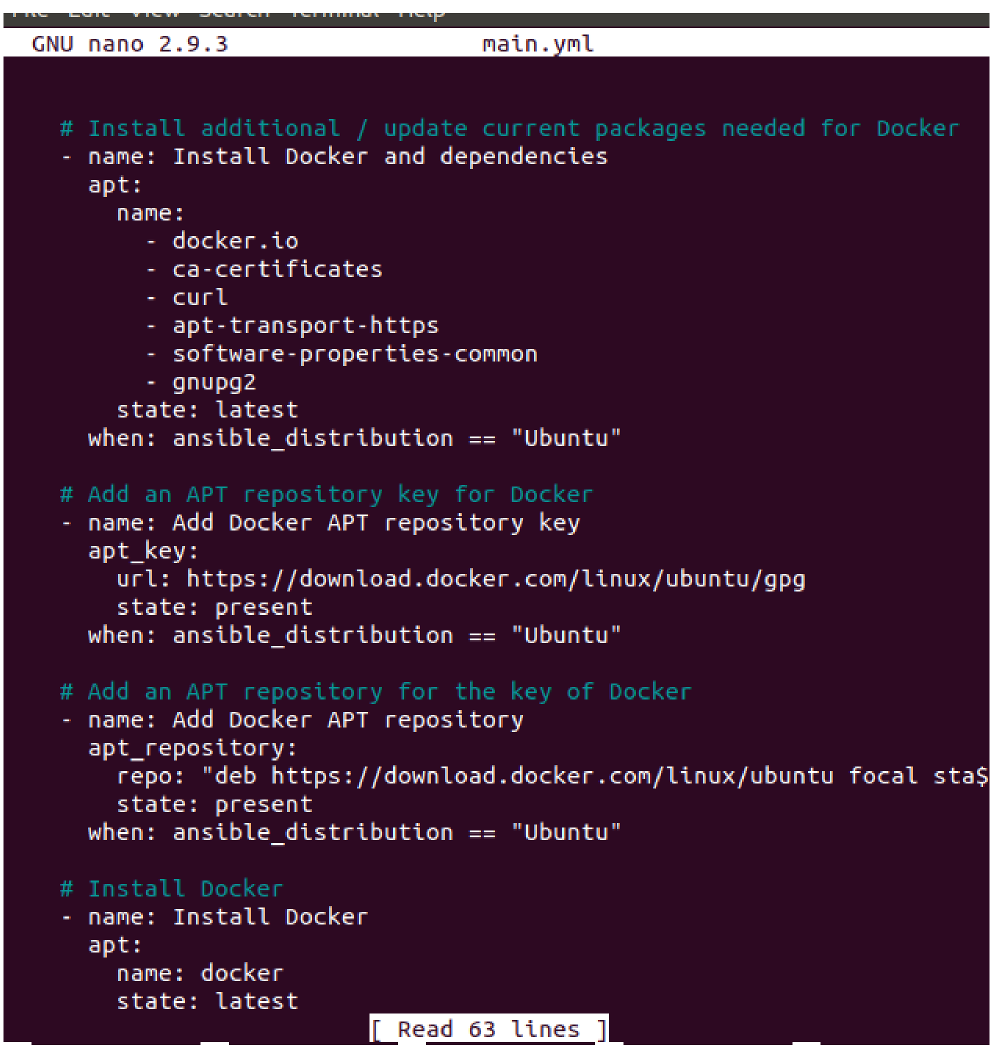
```
  GNU nano 2.9.3                 docker.file                      Modified

FROM ubuntu:latest

RUN apt-get update && apt-get install -y nginx mysql-server

COPY . /var/www/html

EXPOSE 80 3306

CMD ["nginx", "-g", "daemon off;"]
```

Explanation: Docker file is created to install the web and DB server

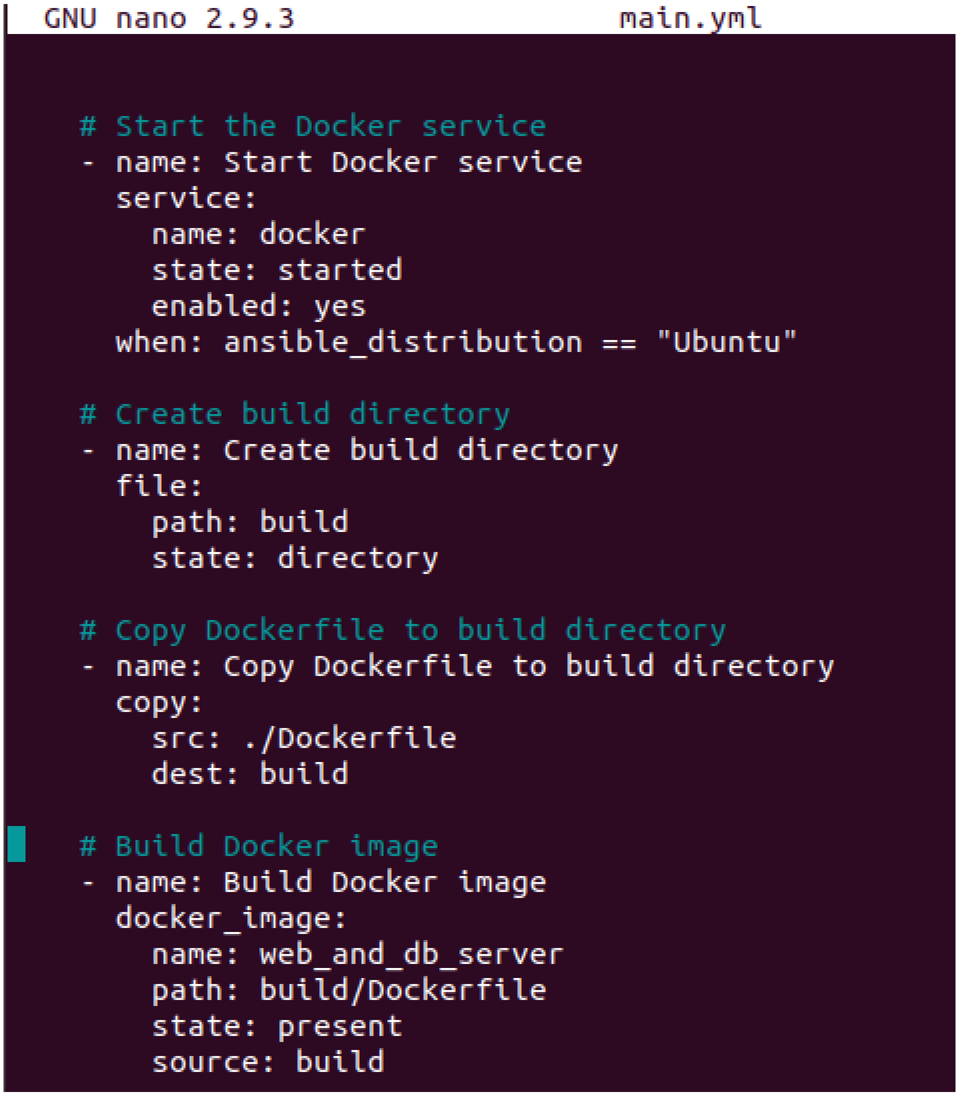
INPUT



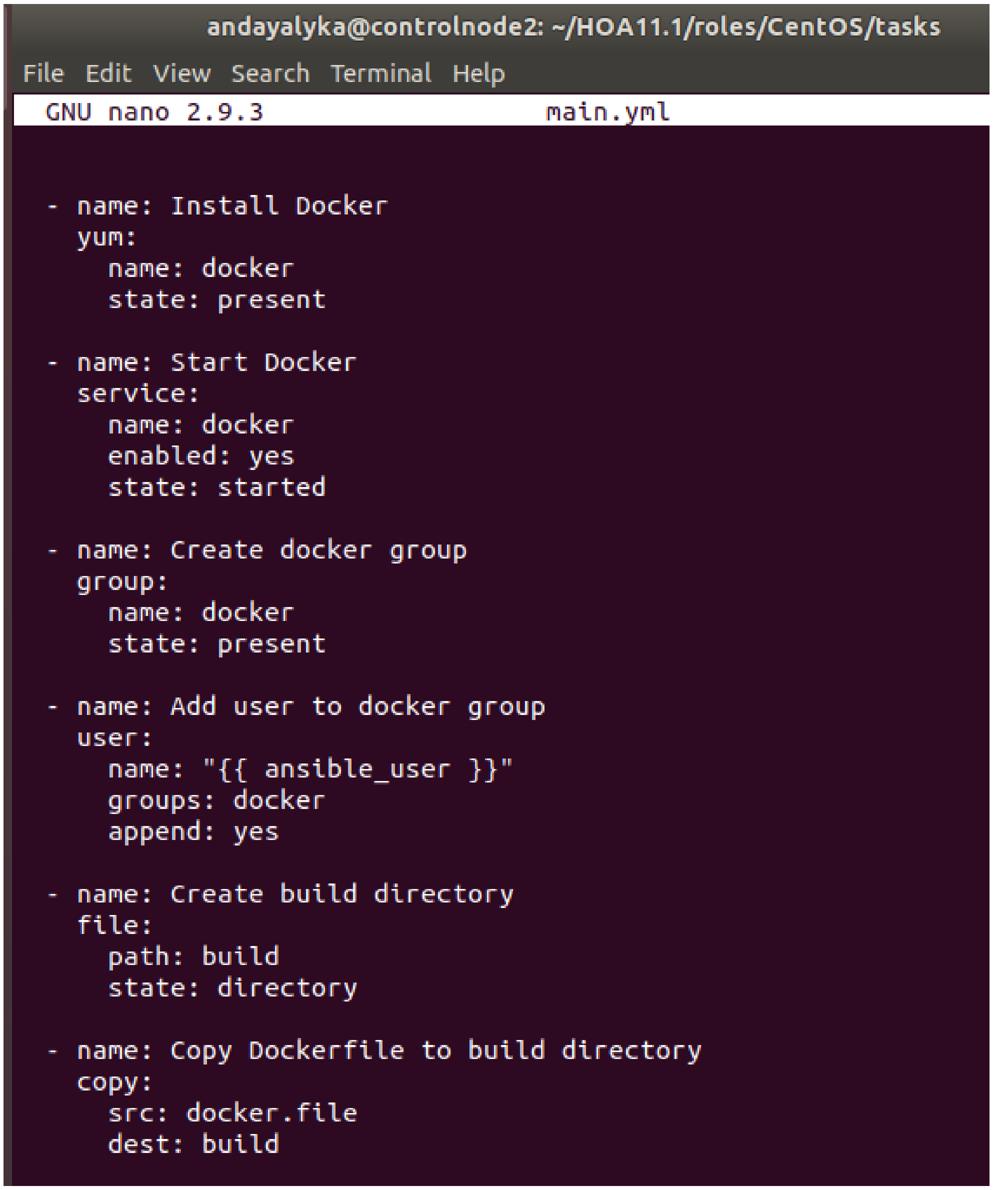
```
  GNU nano 2.9.3                 install.yml

---
- hosts: all
  become: true
  pre_tasks:

  - name: Update repository Index (CentOS)
    tags: always
    yum:
      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: Ubuntu
  become: true
  roles:
    - Ubuntu

- hosts: CentOS
  become: true
  roles:
    - CentOS
```



```
  GNU nano 2.9.3                    main.yml


    # Install additional / update current packages needed for Docker
    - name: Install Docker and dependencies
      apt:
        name:
          - docker.io
          - ca-certificates
          - curl
          - apt-transport-https
          - software-properties-common
          - gnupg2
        state: latest
      when: ansible_distribution == "Ubuntu"

    # Add an APT repository key for Docker
    - name: Add Docker APT repository key
      apt_key:
        url: https://download.docker.com/linux/ubuntu/gpg
        state: present
      when: ansible_distribution == "Ubuntu"

    # Add an APT repository for the key of Docker
    - name: Add Docker APT repository
      apt_repository:
        repo: "deb https://download.docker.com/linux/ubuntu focal sta$
        state: present
      when: ansible_distribution == "Ubuntu"

    # Install Docker
    - name: Install Docker
      apt:
        name: docker
        state: latest
                           [ Read 63 lines ]
```

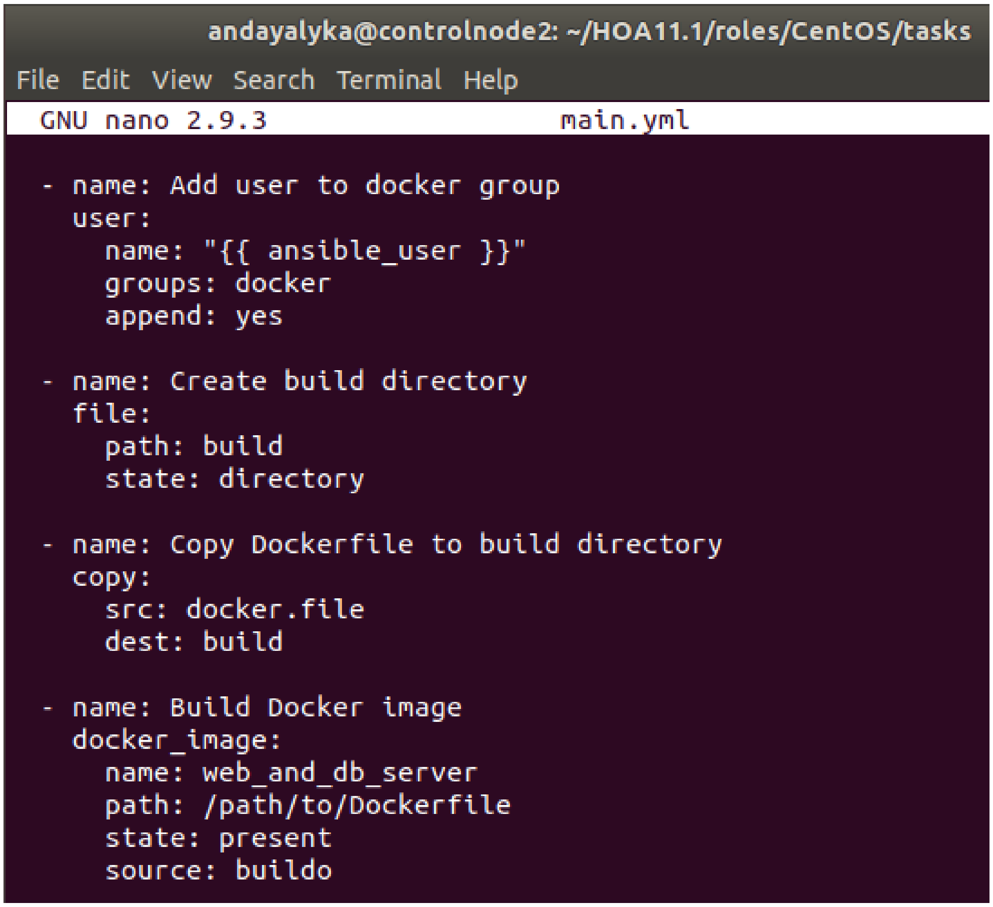


```
  GNU nano 2.9.3                    main.yml


    # Start the Docker service
    - name: Start Docker service
      service:
        name: docker
        state: started
        enabled: yes
      when: ansible_distribution == "Ubuntu"

    # Create build directory
    - name: Create build directory
      file:
        path: build
        state: directory

    # Copy Dockerfile to build directory
    - name: Copy Dockerfile to build directory
      copy:
        src: ./Dockerfile
        dest: build

    # Build Docker image
    - name: Build Docker image
      docker_image:
        name: web_and_db_server
        path: build/Dockerfile
        state: present
        source: build
```



andayalyka@controlnode2: ~/HOA11.1/roles/CentOS/tasks
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                    main.yml


- name: Install Docker
  yum:
    name: docker
    state: present

- name: Start Docker
  service:
    name: docker
    enabled: yes
    state: started

- name: Create docker group
  group:
    name: docker
    state: present

- name: Add user to docker group
  user:
    name: "{{ ansible_user }}"
    groups: docker
    append: yes

- name: Create build directory
  file:
    path: build
    state: directory

- name: Copy Dockerfile to build directory
  copy:
    src: docker.file
    dest: build
```

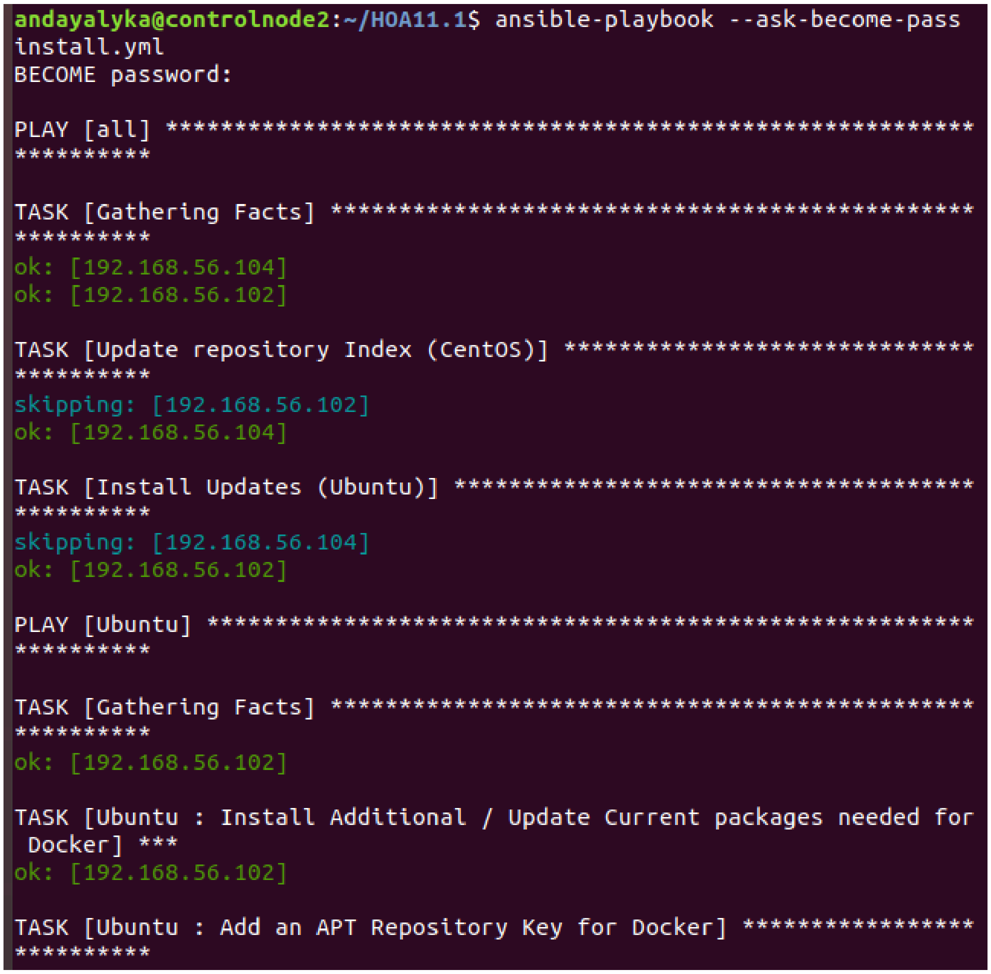


andayalyka@controlnode2: ~/HOA11.1/roles/CentOS/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

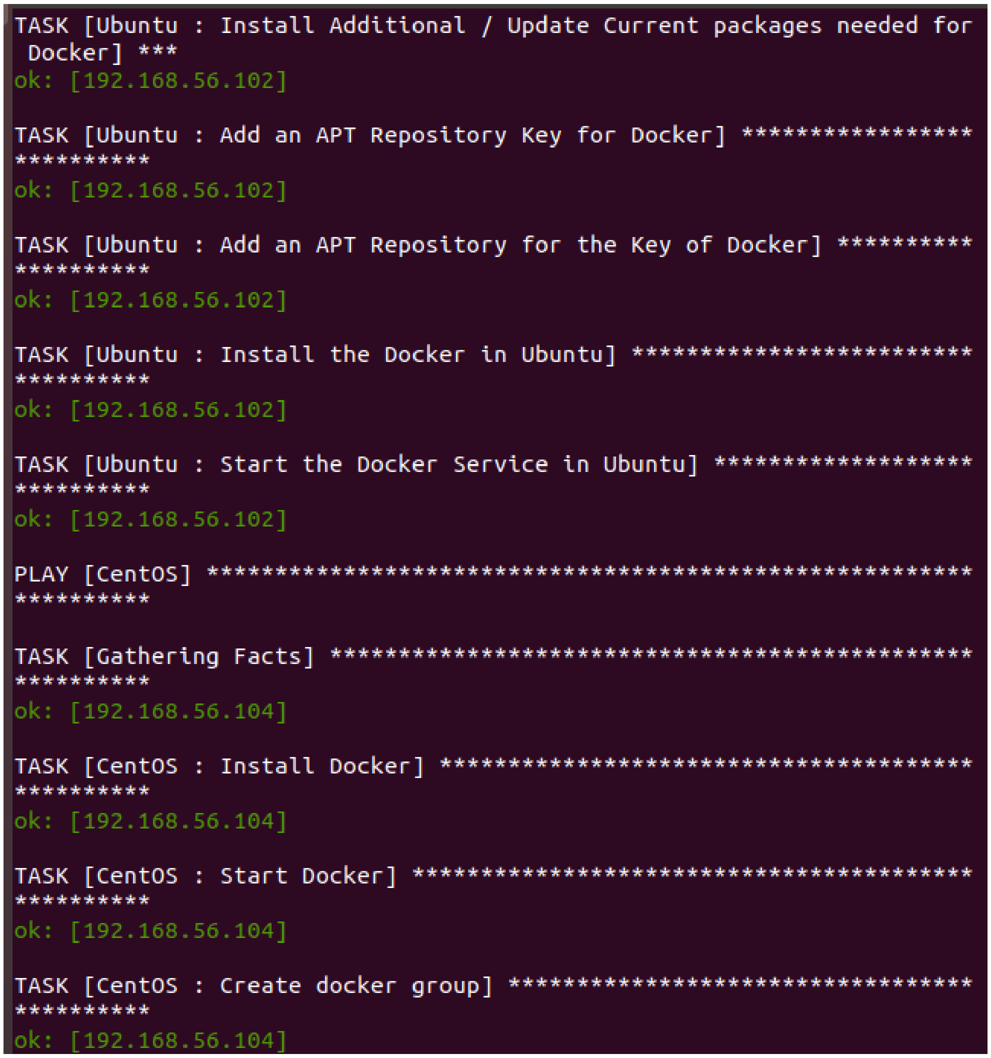
```
- name: Add user to docker group
  user:
    name: "{{ ansible_user }}"
    groups: docker
    append: yes

- name: Create build directory
  file:
    path: build
    state: directory

- name: Copy Dockerfile to build directory
  copy:
    src: docker.file
    dest: build

- name: Build Docker image
  docker_image:
    name: web_and_db_server
    path: /path/to/Dockerfile
    state: present
    source: buildo
```

Explanation: The docker will install in linux server and in centos

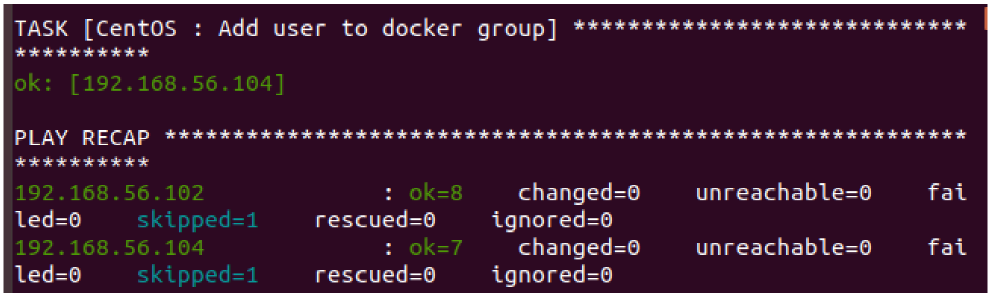
PROCESS



```
andayalyka@controlnode2:~/HOA11.1$ ansible-playbook --ask-become-pass
install.yml
BECOME password:

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

TASK [Gathering Facts] ***********************************************
**********
ok: [192.168.56.104]
ok: [192.168.56.102]

TASK [Update repository Index (CentOS)] ******************************
**********
skipping: [192.168.56.102]
ok: [192.168.56.104]

TASK [Install Updates (Ubuntu)] **************************************
**********
skipping: [192.168.56.104]
ok: [192.168.56.102]

PLAY [Ubuntu] ********************************************************
**********

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

TASK [Ubuntu : Install Additional / Update Current packages needed for
 Docker] ***
ok: [192.168.56.102]

TASK [Ubuntu : Add an APT Repository Key for Docker] *****************
**********
```



```
TASK [Ubuntu : Install Additional / Update Current packages needed for
 Docker] ***
ok: [192.168.56.102]

TASK [Ubuntu : Add an APT Repository Key for Docker] *****************
**********
ok: [192.168.56.102]

TASK [Ubuntu : Add an APT Repository for the Key of Docker] **********
**********
ok: [192.168.56.102]

TASK [Ubuntu : Install the Docker in Ubuntu] *************************
**********
ok: [192.168.56.102]

TASK [Ubuntu : Start the Docker Service in Ubuntu] *******************
**********
ok: [192.168.56.102]

PLAY [CentOS] ********************************************************
**********

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

TASK [CentOS : Install Docker] ***************************************
**********
ok: [192.168.56.104]

TASK [CentOS : Start Docker] *****************************************
**********
ok: [192.168.56.104]

TASK [CentOS : Create docker group] **********************************
**********
ok: [192.168.56.104]
```

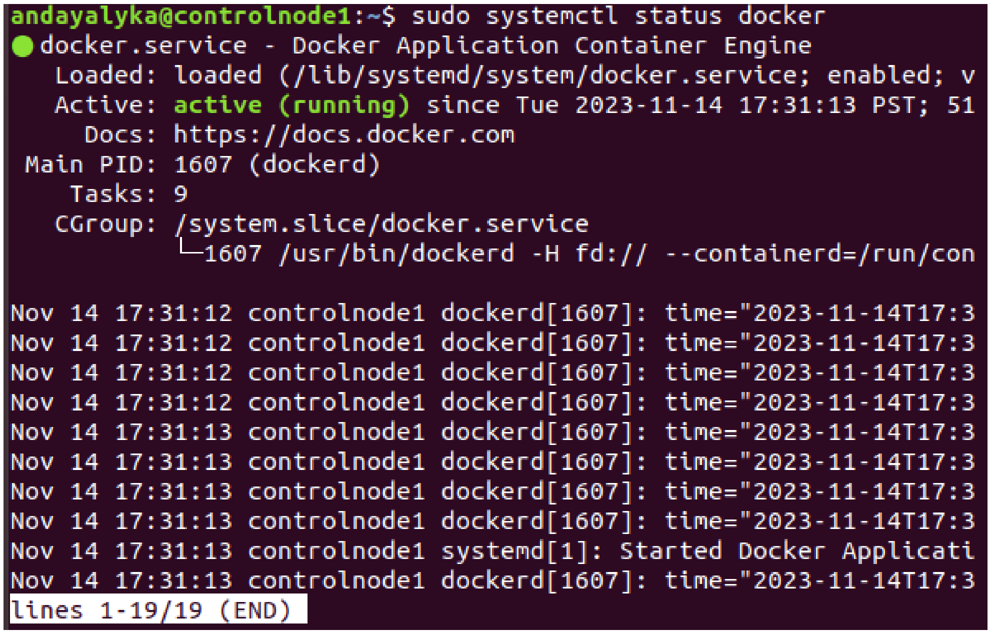


```
TASK [CentOS : Add user to docker group] *****************************
**********
ok: [192.168.56.104]

PLAY RECAP ***********************************************************
**********
192.168.56.102             : ok=8    changed=0    unreachable=0    fai
led=0    skipped=1    rescued=0    ignored=0
192.168.56.104             : ok=7    changed=0    unreachable=0    fai
led=0    skipped=1    rescued=0    ignored=0
```

Explanation: It executed the tasks that I created in the playbook

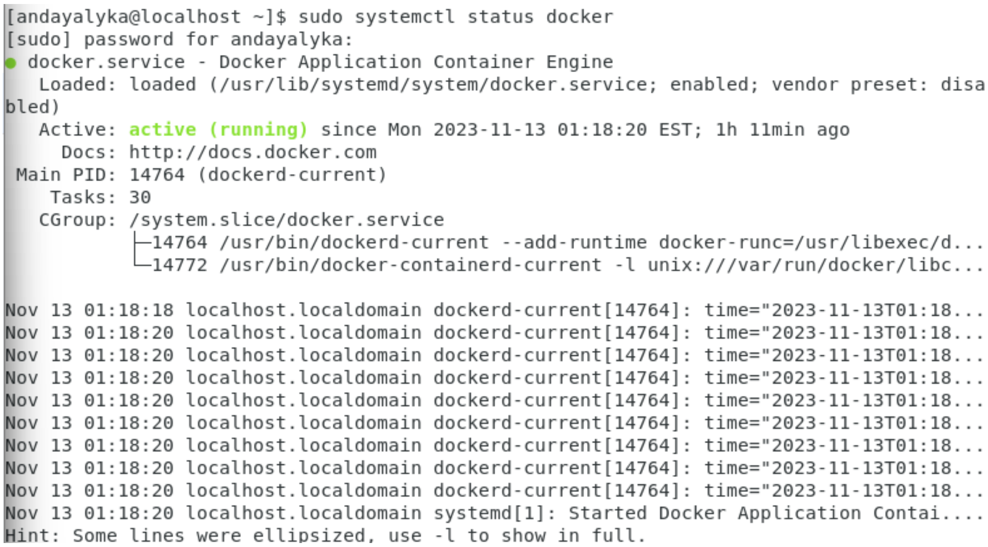
OUTPUT

Ubuntu



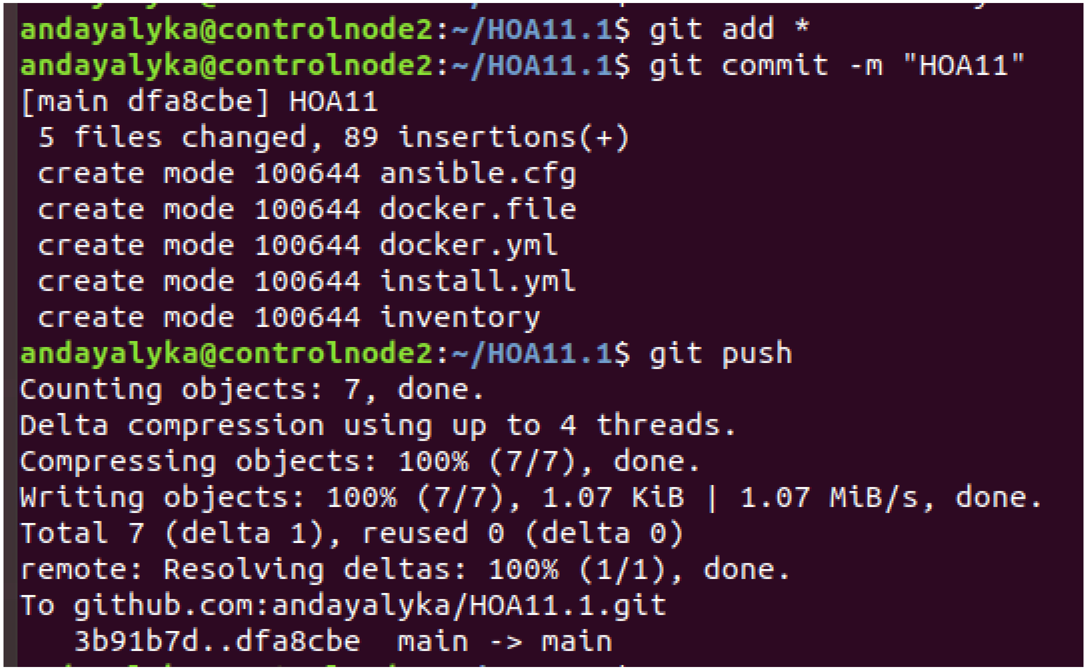
```
andayalyka@controlnode1:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; v
     Active: active (running) since Tue 2023-11-14 17:31:13 PST; 51
       Docs: https://docs.docker.com
   Main PID: 1607 (dockerd)
      Tasks: 9
     CGroup: /system.slice/docker.service
             └─1607 /usr/bin/dockerd -H fd:// --containerd=/run/con

Nov 14 17:31:12 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:12 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:12 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:12 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:13 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:13 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:13 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:13 controlnode1 dockerd[1607]: time="2023-11-14T17:3
Nov 14 17:31:13 controlnode1 systemd[1]: Started Docker Applicati
Nov 14 17:31:13 controlnode1 dockerd[1607]: time="2023-11-14T17:3
lines 1-19/19 (END)
```

CentOS



```
[andayalyka@localhost ~]$ sudo systemctl status docker
[sudo] password for andayalyka:
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; vendor preset: disa
bled)
   Active: active (running) since Mon 2023-11-13 01:18:20 EST; 1h 11min ago
     Docs: http://docs.docker.com
 Main PID: 14764 (dockerd-current)
    Tasks: 30
   CGroup: /system.slice/docker.service
           ├─14764 /usr/bin/dockerd-current --add-runtime docker-runc=/usr/libexec/d...
           └─14772 /usr/bin/docker-containerd-current -l unix:///var/run/docker/libc...

Nov 13 01:18:18 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain dockerd-current[14764]: time="2023-11-13T01:18...
Nov 13 01:18:20 localhost.localdomain systemd[1]: Started Docker Application Contai....
Hint: Some lines were ellipsized, use -l to show in full.
```

Explanation:



```
andayalyka@controlnode2:~/HOA11.1$ git add *
andayalyka@controlnode2:~/HOA11.1$ git commit -m "HOA11"
[main dfa8cbe] HOA11
 5 files changed, 89 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 docker.file
 create mode 100644 docker.yml
 create mode 100644 install.yml
 create mode 100644 inventory
andayalyka@controlnode2:~/HOA11.1$ git push
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.07 KiB | 1.07 MiB/s, done.
Total 7 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:andayalyka/HOA11.1.git
   3b91b7d..dfa8cbe  main -> main
```

Explanation: All of the created files has been push in the github and has a commit message “HOA11”

Reflections:

Answer the following:

1. What are the benefits of implementing containerizations?

The initial step involves comprehending the architecture and prerequisites of log monitoring systems. This encompasses the careful selection of suitable log monitoring tools tailored to the specific requirements of the enterprise. Subsequently, I will delve into Ansible, mastering the creation of playbooks, task definition, and role organization, establishing the groundwork for my automated process.

Conclusions:

Through the process of devising and structuring a workflow for log monitoring using Ansible, I will acquire the knowledge to streamline the setup, configuration, and administration of log monitoring tools in a corporate context. This involves grasping the fundamentals of Infrastructure as Code (IaC) and recognizing Ansible as a potent instrument for this purpose.

The initial step involves comprehending the architecture and prerequisites of log monitoring systems. This encompasses the careful selection of suitable log monitoring tools tailored to the specific requirements of the enterprise. Subsequently, I will delve into Ansible, mastering the creation of playbooks, task definition, and role organization, establishing the groundwork for my automated process.

This endeavor underscores the importance of adhering to best practices when deploying and configuring log monitoring solutions. I will address vital elements such as handling dependencies, safeguarding communications, and ensuring seamless integration with existing infrastructure.

Additionally, I will develop proficiency in the efficient management of logs, including strategies for data retention, analysis, and timely alerting. This involves establishing custom rules and triggers to promptly identify critical events.

Active participation in this endeavor will equip me with hands-on expertise in crafting and executing automated workflows with Ansible. This practical skill set is invaluable for optimizing log monitoring procedures, fortifying security measures, and guaranteeing the dependability of crucial systems within a corporate environment.