DevOps Project Documentation

Table of Contents

1. Introduction

This documentation outlines the setup and configuration of a DevOps project using Jenkins, Git, Docker, and AWS. The project involves building, tagging, and pushing Docker images automatically using Jenkins and Ansible.

2. Infrastructure Setup

AWS EC2 Instances

Create three public AWS EC2 instances named Jenkins, Docker, and Ansible.

Instance Naming

- Jenkins Instance: `jenkins-instance`
- Docker Instance: `docker-instance`
- Ansible Instance: `ansible-instance`

Connecting Instances

Ensure that the instances can communicate with each other. Use SSH for connecting to each instance.

3. Dependency Installation

Jenkins Instance

Connect to the Jenkins instance and install the necessary dependencies:

```
bash
sudo apt update
sudo apt install openjdk-8-jdk -y
sudo apt install git -y
```

Docker Instance

Connect to the Docker instance and install Docker:

```
bash
sudo apt update
sudo apt install docker.io -y
sudo usermod -aG docker ${USER}
sudo systemctl restart docker
```

Ansible Instance

Connect to the Ansible instance and install Ansible:

```
bash
sudo apt update
sudo apt install ansible -y
```

4. Jenkins Installation

Install Jenkins on the Jenkins instance:

```
bash
wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key
add -
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ >
/etc/apt/sources.list.d/jenkins.list'
sudo apt update
sudo apt install jenkins -y
sudo systemctl start jenkins
sudo systemctl enable jenkins
```

Configuring Jenkins

- Access Jenkins on your browser by navigating to `http://<jenkins-instance-public-ip>:8080`.
- Follow the on-screen instructions to complete the setup.
- Install necessary plugins, including the Git plugin.

Connecting Git Repository

- In Jenkins, navigate to "Manage Jenkins" > "Manage Plugins" > "Available" and install the Git plugin.
- Create a new Jenkins job and configure it to connect to your Git repository.

5. Automation Steps

Jenkins Configuration

In the Jenkins job configuration, add the following build steps:

1. Execute Shell:

```
   bash
1. docker image build -t myimage:v1
2. docker image tag myimage:v1 mitul006/myimage:v1
3. docker image push mitul006/myimage:v1
4.
```

Ansible Configuration

In the Ansible instance, create an Ansible playbook (`deploy.yml`) with the following content:

```
yaml
---
- hosts: docker-instance
  tasks:
    - name: Build and Tag Docker Image
      shell: |
        docker image build -t ..$JOB_NAME..$BUILD_ID
        docker image tag ..$JOB_NAME:.$BUILD_ID
mitul006/..$JOB_NAME:.$BUILD_ID
        docker image push mitul006/..$JOB_NAME:.$BUILD_ID
        docker image tag ..$JOB_NAME:.$BUILD_ID mitul006/..$JOB_NAME:latest
        docker image push mitul006/..$JOB_NAME:latest
        docker image rmi ..$JOB_NAME:.$BUILD_ID
mitul006/..$JOB_NAME:.$BUILD_ID mitul006/..$JOB_NAME:latest
      environment:
        JOB_NAME: "..$JOB_NAME"
        BUILD_ID: ".$BUILD_ID"
```

Docker Instance Configuration

1. Create a `docker-compose.yml` file on the Docker instance:

```
yaml
---
version: '3'
services:
  docker-host:
    tasks:
      - name: stop container
        shell: docker container stop cloudknowledge
```

```
      - name: remove container
        shell: docker container rm cloudknowledge
      - name: remove docker image
        shell: docker image rmi mitul006/cloudknowledge
      - name: create new container
        shell: docker container run -itd --name cloudknowledge-container -p
9000:80 mitul006/cloudknowledge
```

2. Run the Ansible playbook:

```
bash
ansible-playbook -i localhost, -e "JOB_NAME=example_job BUILD_ID=1"
deploy.yml
```

6. Conclusion

Your DevOps project is now set up and configured. Jenkins is connected to your Git repository, and the automation steps for building and pushing Docker images are defined using Ansible. Monitor Jenkins for successful builds and deployments. Adjust configurations as needed for your specific project requirements.

This documentation provides a foundation for your DevOps workflow. Customize it based on your project's unique needs and scale as required.