Jenkins: Lab 2
Name: Omar Khaled Gamaleldin

Part 1) Docker Agent

The Docker agent container runing

Running the Build for the agent 1:

Part 2) Cloud Agent
2.1: Terraform Provisioning

```
resource "aws_instance" "jenkins_agent" {
  ami = data.aws_ami.amazon_linux2023.id
  associate_public_ip_address = true
  instance_type = "t3.micro"
  key_name = aws_key_pair.lab_key.key_name
  vpc_security_group_ids = [aws_security_group.sg-jenkins.id]
  subnet_id = aws_subnet.jenkins_public_subnet.id
  tags = {
    "Name" = "jenkins-agent-instance"
  }
}
```

The instance provisioning in the main.tf.
VPC, subnet, and other resources are present in the rest of the main.tf terraform file.

2.2: Ansible Configuration

Ansible Playbook

```yaml
- name: Configure Jenkins Agent
  hosts: jenkins_agents
  become: true

  vars:
    jenkins_user: jenkins

  tasks:
    - name: Initial Update Packages
      ansible.builtin.dnf:
        name: "*"
        state: latest

    - name: Install Required Packages
      ansible.builtin.dnf:
        name:
          - java-21-amazon-corretto
          - docker
          - git
        state: present

    - name: Start Docker and enable it
      ansible.builtin.service:
        name: docker
        state: started
        enabled: true

# To seperate from the root user
    - name: Create Jenkins User
      ansible.builtin.user:
        name: "{{ jenkins_user }}"
        append: true
        groups: docker
        shell: /bin/bash

    - name: Allow jenkins user passwordless sudo
      ansible.builtin.copy:
        dest: /etc/sudoers.d/jenkins
        content: "jenkins ALL=(ALL) NOPASSWD:ALL\n"
        mode: '0440'
    - name: Add ssh-key for Jenkins user to use by Jenkins Master
      ansible.posix.authorized_key:
        user: "{{ jenkins_user }}"
        state: present
        key: "{{ lookup('file', './controller_id_rsa.pub') }}"
```

Nodes:

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time	
🖥	Built-In Node	Linux (amd64)	In sync	36.81 GiB	5.72 GiB	36.81 GiB	0ms	⚙
🖥	docker-agent	Linux (amd64)	In sync	36.81 GiB	5.72 GiB	36.81 GiB	43ms	⚙
🖥	ec2-agent	Linux (amd64)	⚠ 15 sec ahead	N/A	⚠ 0 B	N/A	21926ms	⚙
	last checked	**3 min 16 sec**	**3 min 15 sec**	**3 min 4 sec**	**3 min 10 sec**	**3 min 4 sec**	**3 min 12 sec**	

Nodes + New Node Configure Monitors ↻

Part 3) Jenkinsfile to build and deploy sample app once in docker-agent and once in ec2-agent

```
pipeline {
    agent none

    stages {
        stage('Build on Docker Agent') {
            agent { label 'docker-agent' }
            steps {
                echo "Building app on Docker agent"
                sh 'whoami'
                sh 'hostname'

                // Example: run a build inside docker
                sh 'docker run --rm -v $PWD:/app -w /app openjdk:17 javac HelloWorld.java'
                sh 'docker run --rm -v $PWD:/app -w /app openjdk:17 java HelloWorld'
            }
        }

        stage('Build on EC2 Agent') {
            agent { label 'ec2-agent' }
            steps {
                echo "Building app on EC2 agent"
                sh 'whoami'
                sh 'hostname'

                // Simple local build (no docker, runs directly on EC2)
                sh 'echo "public class HelloWorld { public static void main(String[] args) { System.out.println(\\"Hello from EC2 agent!\\"); } }" > HelloWorld.java'
                sh 'javac HelloWorld.java'
                sh 'java HelloWorld'
            }
        }
    }
}
```

Docker Stage:

```
Started by user Omar Khaled
[Pipeline] Start of Pipeline
[Pipeline] stage
[Pipeline] { (Build on Docker Agent)
[Pipeline] node
Running on docker-agent in /home/jenkins/workspace/multi-agent pipeline
[Pipeline] {
[Pipeline] echo
Building app on Docker agent
[Pipeline] sh
+ whoami
jenkins
[Pipeline] sh
+ hostname
0c289a7c33e7
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
```

EC2 Stage

```
[Pipeline] stage
[Pipeline] { (Build on EC2 Agent)
[Pipeline] node
Still waiting to schedule task
'ec2-agent' is offline
Running on ec2-agent in /home/jenkins/agent/workspace/multi-agent pipeline
[Pipeline] {
[Pipeline] echo
Building app on EC2 agent
[Pipeline] sh
+ whoami
jenkins
[Pipeline] sh
+ hostname
ip-10-0-0-244.ec2.internal
[Pipeline] sh
+ echo 'public class HelloWorld { public static void main(String[] args) { System.out.println("Hello from EC2 agent!"); } }'
[Pipeline] sh
+ javac HelloWorld.java
[Pipeline] sh
+ java HelloWorld
Hello from EC2 agent!
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Part 4) Multi-Branch Build

main branch:


```
Cloning repository https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git
 > git init /var/lib/jenkins/workspace/Multi-Bracnch-Build_main # timeout=10
Fetching upstream changes from https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git
 > git --version # timeout=10
 > git --version # 'git version 2.51.0'
using GIT_ASKPASS to set credentials Github Token for repo code checkout
 > git fetch --no-tags --force --progress -- https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git +refs/heads/main:refs/remotes/origin/main # timeout=10
 > git config remote.origin.url https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git # timeout=10
 > git config --add remote.origin.fetch +refs/heads/main:refs/remotes/origin/main # timeout=10
Avoid second fetch
Checking out Revision a8f75d91c8813e64c1468914824ef4d173c409a9 (main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f a8f75d91c8813e64c1468914824ef4d173c409a9 # timeout=10
Commit message: "The Jenkinsfile on Main Branch"
 > git rev-list --no-walk a8f75d91c8813e64c1468914824ef4d173c409a9 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Print Branch)
[Pipeline] echo
Running on branch: main
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline

GitHub has been notified of this commit's build result

Finished: SUCCESS
```

dev branch:


```
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git # timeout=10
Fetching without tags
Fetching upstream changes from https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git
 > git --version # timeout=10
 > git --version # 'git version 2.51.0'
using GIT_ASKPASS to set credentials Github Token for repo code checkout
 > git fetch --no-tags --force --progress -- https://github.com/omarkhaled-19/Jenkins-lab2-multibranch-pipeline.git +refs/heads/dev:refs/remotes/origin/dev # timeout=10
Checking out Revision 469d6aef1558e6c405f6943ae0b63dc7bf31be0e (dev)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 469d6aef1558e6c405f6943ae0b63dc7bf31be0e # timeout=10
Commit message: "Second change"
 > git rev-list --no-walk ae02d91fd5937657db1f63c7ce5ac3785af8b1d4 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Print Branch)
[Pipeline] echo
Running on branch dev after change again: dev
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline

GitHub has been notified of this commit's build result

Finished: SUCCESS
```

Slack Notification

jenkins_builds

You created this channel today. This is the very beginning of the **# jenkins_builds** channel.

[Add description]   [Add people to channel]   [Send emails to channel]

Today ⌄

Omar Khaled 20:34
joined #jenkins_builds.

Omar Khaled 20:36
added an integration to this channel: jenkins

jenkins APP 20:44
Slack/Jenkins plugin: you're all set on http://localhost:8080/

jenkins APP 21:23
Slack/Jenkins plugin: you're all set on http://localhost:8080/

Job 'Multi-Bracnch-Build/main [6]' succeeded on built-in

Message #jenkins_builds

Files

main

Go to file

Jenkinsfile
README.md

Jenkins-lab2-multibranch-pipeline / Jenkinsfile

omarkhaled-19 Again ✓

Code   Blame   19 lines (18 loc) · 457 Bytes

```
1    pipeline {
2      agent any
3      stages {
4        stage('Print Branch') {
5          steps {
6            echo "Running on branch: ${env.BRANCH_NAME}"
7          }
8        }
9      }
10       post {
11         success {
12           slackSend(message: "Job '${env.JOB_NAME} [${env.BUILD_NUMBER}]' succeeded on ${env.NODE_NAME}")
13         }
14         failure {
15           slackSend(message: "Job '${env.JOB_NAME} [${env.BUILD_NUMBER}]' failed. Check console output: ${env.BUILD_URL}")
16         }
17       }
18     }
```