**Project 2**

**Hangout Point.**

**Objective:** Set up the Jenkins server in master or slave architecture. Use the Jenkins plugins to perform the computation part on the Docker containers. Create Jenkins pipeline script. Use the GIT web hook to schedule the job on check-in or poll SCM. Build an image using the artifacts and deploy them on containers. Remove the container stack after completing the job.

**Tools required:** EC2, Jenkins, Docker, Ansible.

**STEP 1: CREATE AN EC2 INSTANCE IN THE AWS CONSOLE**

**1.1 Graphical user interface, application, Word

Description automatically generated**

**1.2. Install all necessary software:**

-AWS has a feature called EC2 User Data. We will install Jenkins, Docker, and Ansible.

A screenshot of a computer

Description automatically generated with medium confidence

-Now upload it in the EC2 User Data

Graphical user interface, text, application

Description automatically generated

**STEP 2: SET UP AND RUN JENKINS**

**2.1**

-Now SSH into the master server and start Jenkins.

Text

Description automatically generated

**2.2**

-Visit the Jenkins console by getting the IP address of your EC2 and put the port 8080 with it. If everything goes right, the console will open with this.

Graphical user interface, text

Description automatically generated

**2.3**

-Now go to the path indicated in the console and use your temporary password and choose the plugins you want to install.

Graphical user interface, table

Description automatically generated

**2.4**

-Then afterwards Jenkins is finished.

Graphical user interface, text, application

Description automatically generated

**2.5**

-Now let’s install required Docker and Ansible plugins so we can integrate it with our Jenkins pipeline later.

**Docker:**

Graphical user interface, text, application

Description automatically generated

**Ansible:**

**Graphical user interface, application

Description automatically generated**

**Git:**

**Application

Description automatically generated with low confidence**

**STEP 3: INSTALL AND RUN DOCKER**

-Install docker by typing **yum install docker-ce docker-ce-cli containerd.io**

Text

Description automatically generated

-Now run Docker by typing **systemctl start docker**

A picture containing text

Description automatically generated

**STEP 4: INSTALL ANSIBLE**

-Now install Ansible by typing the command:

subscription-manager repos --enable rhel-7-server-ansible-2.6-rpms

yum install -y ansible

yum install -y python

yum install -y libselinux-python

yum install -y sshpass

**Text

Description automatically generated**

**STEP 5: ENABLE PASSWORDLESS SSH FOR ANSIBLE AND JENKINS**

5.1 SSH to the Master server and type ssh-keygen

Text

Description automatically generated

5.2 Copy the id\_rsa to the /home/root/.ssh/id\_rsa

Text

Description automatically generated

5.3 Go to the Slave server and paste it the same folder in /home/root

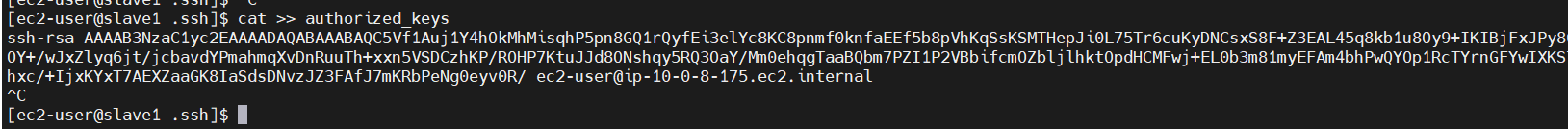
5.4 Change the configuration in /etc/ssh/sshd\_config

Text

Description automatically generated with low confidence

Then restart the sshd service by typing systemctl restart sshd

5.5 Now paste the id\_rsa.pub and paste it in the /home/ec2-user/authorized\_keys



5.6 Now try to SSH

Text

Description automatically generated

**STEP 7: CREATE AN ANSIBLE PLAYBOOK**

7.1 Create an inventory file first. This will determine the hosts of our Ansible playbook. Put the name of the slave server, IP address of the slave server, and default user.

**Text

Description automatically generated**

7.2 Test the Ansible playbook. Type **ansible <name\_of\_server> -m ping -i inventory.txt**

**Text

Description automatically generated**

**STEP 8: CREATE A DOCKER IMAGE**

8.1 Create a Dockerfile

Text

Description automatically generated

8.2 Run the command **docker build Dockerfile**

**STEP 9: CREATE A JENKINS JOB AND ASSEMBLE PIPELINE**