JENKINS INSTALLATION ON UBUNTU USING AWS

STEP 1: UBUNTU 20.04LT

T2.MICRO

SG: SSH ANYWHERE

ALL TRAFFIC ANYWHERE

KP NEW

LAUNCH

STEP2: copy public ip

Download putty & puttygen

Open puttygen--🡪 load pem key

Save

Open putty: host ip: public ip paste

Ssh-auth-upload ppk key

Login as ubuntu enter

STEP3: sudo apt-get update

STEP4 : sudo apt-cache search openjdk

STEP5: sudo apt-get install openjdk-8-jdk

click Y

STEP 5 java -version (check java install or not)

STEP 6: OPEN BROWSER TYPE JENKINS DOWNLOAD

<https://pkg.jenkins.io/debian-stable/> (UBUNTU DEBIAN )

# Jenkins Debian Packages

This is the Debian package repository of Jenkins to automate installation and upgrade. To use this repository, first add the key to your system:

curl -fsSL <https://pkg.jenkins.io/debian-stable/jenkins.io.key> | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

Then add a Jenkins apt repository entry:

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

Update your local package index, then finally install Jenkins:

sudo apt-get update

sudo apt-get install fontconfig openjdk-11-jre

sudo apt-get install jenkins

STEP 7: sudo service Jenkins status

Press q to quit enter

Step 8: goto browser paste ur public ip

127.0.0.0:8080

Copy above password

STEP 9: goto putty

Sudo cat paste above passwd

Copy passwd

Paste in browser

Step 10: install suggsted Jenkins

STEP 11: USERNAME…………..

Save and finish

Start Jenkins

Jenkins console

New job--🡪freestyle-🡪build now ---- SELECT SHELL SCRIPT

TYPE == echo “welcome” --🡪apply save

INSTALL PIPELINE PACKAGE

GO TO MANAGE PLUGINS

MANAGE ----TYPE PIPELINE---GO DOWN UNDER PIPELINE

WRITE SCRIPT

pipeline {

agent any

stages {

stage('build') {

steps {

echo "Hello World!"

}

}

}

}

.

sudo yum update –y

$ sudo wget -O /etc/yum.repos.d/jenkins.repo <http://pkg.jenkins.io/redhat/jenkins.repo>

$ sudo rpm --import <https://pkg.jenkins.io/redhat/jenkins.io.key>

$ sudo yum install jenkins –y

# 1. Installing Jenkins

First, update the default Ubuntu packages lists for upgrades with the following command:

sudo apt-get update

Then, run the following command to install JDK 11:

sudo apt-get install openjdk-11-jdk

Now, we will install Jenkins itself. Issue the following four commands in sequence to initiate the installation from the Jenkins repository:

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

sudo apt-get install Jenkins

Once that’s done, start the Jenkins service with the following command:

sudo systemctl start jenkins.service

To confirm its status, use:

sudo systemctl status Jenkins

With Jenkins installed, we can proceed with adjusting the firewall settings. By default, Jenkins will run on port 8080.

In order to ensure that this port is accessible, we will need to configure the built-in Ubuntu firewall (ufw). To open the 8080 port and enable the firewall, use the following commands:

sudo ufw allow 8080

sudo ufw enable

Once done, test whether the firewall is active using this command:

sudo ufw status

With the firewall configured, it’s time to set up Jenkins itself. Type in the IP of your EC2 along with the port number. The Jenkins setup wizard will open.

To check the initial password, use the cat command as indicated below:

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

All Set! You can now start automating...

# 2. How to Configure and Run Jenkins Behind Apache Reverse Proxy?

Installing Apache Install Apache from Repo

sudo apt-get update

sudo apt-get install apache2 -y

Enable proxy, proxy\_http, headers module

sudo a2enmod proxy

sudo a2enmod proxy\_http

sudo a2enmod headers

Edit Apache Configuration file

cd /etc/apache2/sites-available/

sudo vim jenkins.conf

Then, In the file enter the following code snippet to make the Apache works for Jenkins. Then, In this ServerName should be your domain name, ProxyPass should point your localhost point to Jenkins (Port 8080) and ProxyPassReverse should be added for both localhost address and Domain address. In the block, we need to give access to the apache to handle the Jenkins.

<Virtualhost \*:80>

ServerName your-domain-name.com

ProxyRequests Off

ProxyPreserveHost On

AllowEncodedSlashes NoDecode

<Proxy http://localhost:8080/\*>

Order deny,allow

Allow from all

</Proxy>

ProxyPass / http://localhost:8080/ nocanon

ProxyPassReverse / http://localhost:8080/

ProxyPassReverse / http://your-domain-name.com/

</Virtualhost>

Enable and Restart Jenkins

sudo a2ensite jenkins

sudo systemctl restart apache2

sudo systemctl restart jenkins

Configuring Firewall

sudo ufw allow ssh

sudo ufw allow http

sudo ufw allow https

Now, enable firewall by passing following command.

ufw enable

That’s all. Now on, your Jenkins server will run behind the Apache’s Reverse Proxy.