JENKINS

==========

1. Introduction to CI & CD

2. Introduction to Jenkins

3. Installing Jenkins

4. Accessing Jenkins Console

5. Overview of Jenkins Console

6. JDK and Maven Setup in Jenkins

7. Types of Jobs in Jenkins

8. Creating a Freestyle Job, Understanding Build Now, Build History, Accessing Builds, Configuring the Exisiting Jobs

9. Deep dive - Job Options - Part 1

==========
==========

CI & CD - Continuous Integration and Continuous Delivery/Deployment

CI = CB (Continuous Build) + CT (Continuous Test)

Continuous Build ----> v1, v2, v3, v4, ..., v10 (Current)

There will always be changes in the code over a period of time, and whenever the code changes, we have to build the code. Building the code means packaging the code (war files)

Continuous Test -----> v1, v2, v3, v4, ..., v10 (Current)

Whenever the code gets updated, we (Testing team) need to test the code for code quality checks.

The process of building the code and testing the code is known as Continuous Integration

CD - Continuous Delivery

Deploying the code manually to the production environment.

CD - Continuous Deployment

Deploying the code automatically to the production environment.

Pre-CI ------> Manual Deployment

Post-CI ------> Automated Deployment

Introduction to Jenkins

=> It is a free and open-source s/w

=> It is an automation tool for CI&CD purposes.

=> Developed by using Java

=> It is a part of Hudson Project

=> Before changing the name to jenkins, we used to call jenkins as Hudson

Developers ----> Source code ----> Testing of code ----> keep the code in repo ----> take the code from the repo ----> deploy the code

=> Using jenkins, we can implement CI&CD process.

=> Jenkins is built on java-11 version s/w

=> Jenkins was invented by Kosukha Kawaguchi in 2004 July in Sun Micro Systems

=> Currently, jenkins is owned by Orcale, and made it free.

=> Jenkins is a service. Not a tool.

=> Jenkins will run on Port Number 8080

Build and Deployment Process

---

1. Take the latest code from repo.

2. Compile the source code

3. Execute the code (JUnit Testing)

4. Perform code review

5. Package the code (war files)

6. Deploy the war file into the server(s)

All the above tasks can be automated using Jenkins.

Installation of Jenkins on EC2 Instance:

****

Part 1

****

1. Launch and Connect to the Instance using MobaXTerm

2. Become Root User

    sudo su

    cd (You will see only IP of Instance)

    sudo yum update -y

****

Part 2

****

https://www.jenkins.io/doc/tutorials/tutorial-for-installing-jenkins-on-AWS/

3. Add Jenkins repo. to our YUM repo:

    sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

4. Import a key-file from Jenkins-CI to enable the installation from package:

sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key

sudo yum upgrade

****

Part 3

****

5. Installing Java

sudo amazon-linux-extras install java-openjdk11 -y

java -version

6. Install Jenkins:

sudo yum install jenkins -y

7. Lets start, enable, and check the status of Jenkins

sudo systemctl enable jenkins

sudo systemctl start jenkins

sudo systemctl status jenkins

You should see Active & Running in Green Colour, which means the Jenkins Service is successfully installed.

Note:

To remove jenkins: sudo yum remove jenkins

****

Part 4

****

By default, the Jenkins tool will run on Port No. 8080

Enable Port No. 8080 for the EC2 Instance inorder to access Jenkins Tool.

Check Instance ----> Security ----> Click on link below the Security Group ----> Add Rule ----> Custom TCP, Port No. 8080 ----> Save Rules.

Access Jenkins:

<public ip>:8080

cat <Paste the red code visible while accessing jenkins in browser> and execute in MobaXTerm

Copy the code visible in MobaXTerm and Paste in Notepad

8f39e64834e34e3ba2e6b0ef55695ea0

Paste the above code in Jenkins Browser under "Administrative Password"

Click on "Install Suggested Plugins"

Create First Admin User

Provide Username, Password, Full Name, and Email

Save and Continue

Copy the Jenkins URL and Paste in Notepad

http://3.135.249.138:8080/

Save and Finish

Start Using Jenkins

You will see Jenkins Console

From now, we are going to create the JENKINS Jobs

Types of Jobs in Jenkins

---

1. Freestyle Projects - Using Jenkins dashboard we will create the jenkins jobs
2. Pipelines - Using code we will create the jobs

Lets create a simple job in Jenkins and run the job

Jenkins Console ----> New Item ----> Enter Name of the Job ----> Select 'Free Style Project' ----> OK ----> To Configure our Jenkins Job, we need to configure 6 steps (General, SCM, Build Triggers, Build Environment, Build Steps, Post-Build Action)

How to increase the build limits?

Build Triggers

In order to build the builds periodically we have to write CRON expression.

The CRON expression is configured in Build Triggers option

If you want to build triggers periodically, we have to write the cron expression

* * * * * * (Every minute the build will execute)

H * * * * (Every Hour the build will execute)

H/2 * * * * (Every Half-an-Hour the build will execute)

Build Steps

Adding a Shell scripting for execution

How to restore a deleted job?

Plugin - Job Configuration History

Note: Servers follow UTC Timezone

Throttle Builds

To restrict the number of builds in an hour/minute/day we will use THROTTLE BUILD

Trigger Builds Remotely

3.107.101.13:8080/job/Demo2/build?token=Kastro-Job or /buildWithParameters?token=Kastro-Job

Project 1: Integration of GITHUB + Jenkins + Maven

POLL-SCM

By
Kastro Kiran V

LinkedIn: https://www.linkedin.com/in/kastro-kiran/

YouTube: https://www.youtube.com/playlist?list=PLs-PsDpuAuTdOcZa-DDgG8KRbtMI_XRrC