**JENKINS**

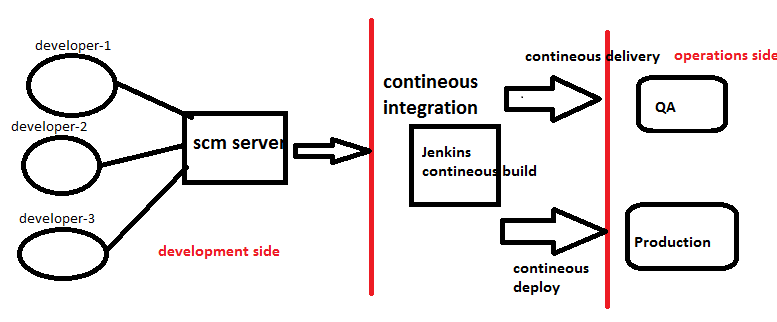
<http://13.127.152.23:8080/login?from=%2F>

**Q) What is Jenkins?**

It is a coontinuous integration (CI) and continuous deployment (CD) tool.

**Q) What is CI and CD?**

CI means integrate the developer’s code. CD means deliver or deploy the code into the different environment (QA, pre-production, production).



- It is a java based application.

- It will help to automate the process of code integration, code build, and code delivery.

- It is an open source tool like Hudson/bamboo/teamcity.

- Jenkins has original name as Hudson.

- Jenkins is the core part of the Devops.

- Jenkins will connect to the scm, qa, tomcat, production, etc. to interact with these system.

- It is interacted with those systems with the plugins.

- It is plugin based tool.

- Nearly 1000+ plugins Jenkins will support.

**Q) Important section of Jenkins?**

- Plugin management.

- Global tool management.

- Job management.

- Jenkins dashboard.

- User management.

- Security management.

- Slave connection.

**Q) Role of Devops Engineer (or) Build and Release Engineer?**

- Installation and configuration.

- Plugin management.

- User management.

- Security management.

- Global tool management.

- creating jobs.

- monitoring jobs.

- Jobs pipeline

- integrating different tools.

- taking Jenkins backup.

**NOTE**: In Jenkins server git must and should install.

**Jenkins install and configure:**

1. Using yum packages.

2. Using docker image.

3. Using jenkins.war.

4. Using tomcat

**Pre-requisites:**-

- Java 1.8.0 --- yum -y install java-1.8.0-openjdk

- Java 1.8.0 --- yum -y install java-1.8.0-opnjdk-devel

**Steps to install:**

Step-1.

- set up the repositories.

- install Jenkins (yum -y install Jenkins)

Step- 2.

- set up docker environment

- download Jenkins image.

- start the container.

Step- 3.

- download jenkins.war LTS file from official site

- start jenkins.war

#java -jar jenkins.war

Step- 4.

- download tomcat 7

- download jenkins.war

- copy jenkins.war file into the webapps folder of the tomcat (/opt/apache-tomcat-7-xx/webapps/)

- start the tomcat. (./bin/startup.sh)

#cp gitbucket.war /opt/apache-tomcat/webapps/

#ls /opt/apache-tomcat/webapps/

**Configuration:**

1) Unlock the Jenkins use admin password

- cat /home/ramki/.jenkins/secrets/AdminPasword

2) Install the required plugins.

3) Setup the user and password.

4) Login to the dashboard using username and password.

**Plugin Management:**

Q) What is plugin?

Plugin is an additional software.it will help to interact with the other systems.

Under plugin management we can install, update, and remove the plugins.

Manage Jenkins--manage plugins

https://mvnrepository.com/artifact/org.jenkins-ci.plugins/mantis/0.26

http://repo.spring.io/plugins-release/org/jenkins-ci/plugins/mantis/0.26/mantis-0.26.hpi

**Global tool management:**

* These are tools it will help develop the application like maven.
* Difference between tool and plugin
* We can maintain multiple versions of tool.
* But we can maintain only single version of plugin.

Manage Jenkins----global tool configuration

* **Job management:**

**Q) What is job?**

Job means task. To build, integration, delivery these kind of tasks we can handle in Jenkins job management.

* Job is a core part of the Jenkins.
* Job will pull the code from scm server.
* Build the application if successfully build delivery the code.

**There are different types of jobs:**

1) Free style

We can integrate any type of files.

2) Maven job

We can integrate only java files

3) Pipeline jobs

Sequence of jobs, one job is connected to another job.

4) Multipage line jobs

**Q) How to schedule a job in Jenkins?**

- get the code from scm server (gitbucket)

- setting trigger (build process trigger)

- Setup build environment

- build action

- Post build action

**Build maven Job in Jenkins:**

Code compile:

Code test:

Code integration:

Code package:

For job pipeline install the "build pipeline plugin"

Here trigger the job as build after other job.

**Deploy:**

For deploy the code into tomcat install "deploy to container plugin"

At post build action setup the "deploy war/ear to container"

**Backup:**

For Jenkins server backup install "backup plugin"

**Email notification in Jenkins:**

Install the plugin "email extension plugin", "email extension template plugin"

Configure mail server, port, SMTP authentication in "system configure"

**User management:**

1) Install necessary plugin

Role based authorization strategy

2) Create users

Manage Jenkins ---> Manage users --> create users

dev\_user

test\_user

manager

3) Create roles

Developer

Tester

For this you need to enable "role based strategy" in "configure global security"

4) Create roles and assign users to the roles

Manage Jenkins ---> Manage & assign roles

I)

Global roles and project roles

| |

job\_creator java,php, maven.

ii)

Global roles

admin job\_creator

dev\_user

test\_user

manager

iii) project roles

Java php maven

dev\_user

tset\_user

manager

**Jenkins Cli:**

We can access Jenkins from cli

1)download the jenkins-cli.war and run it

2)in security management enable "cli over remoting"

3) assign anonymous to admin role

- java -jar jenkins-cli.jar -s http://192.168.43.183:8080/jenkins/ help

Then run the commands it is unsecure procedure we can Ssh instead of this.

1) Ssh plugin is required.

2) Create public and private key of the system user.

3) Place public key of user in admin user configuration section.

4) Enable Ssh in configure global security.

- ssh -l (user name) -p 2020 localhost version

**Set up slave in Jenkins:**

1) Manage Jenkins --> Manage nodes

2) Add node

3) Set up node

name:new\_node

doc\_directory:/home/krishna/

Launch method :via ssh

host:(ip)

credentials:

Host verification: known host verification

From Jenkins master cmd "ssh <node ip>"

* **How to trouble shoot jenkins admin password**

**If we lose the Jenkins admin password how we login to the Jenkins server?**

**Trouble shoot it?**

**In cli**

Cd .jenkins

#ls | grep Config.xml

Using Config.xml file.

#vi Config.xml

--first change the line from (use security) true (/use security)

To (use security) false (/use security)

-then go to the Jenkins portal and “Config global security”

Here

Enable “security relam as use jenkins own database”

And setup “logged in user can do anything”

Again sign up and then give the admin access to particular user..

FAQs

What is job

Job means task. To build