JENKINS DOWNGRADE

* Install java and Jenkins and apt update
* Cd /usr/share/java
* Delete Jenkins war file
* Paste wget \*Jenkins war link older version you want to downgrade\*
* Restart Jenkins \*Systemctl restart jenkins\*

JENKINS UPGRADE

* Install java and Jenkins and apt update
* Cd /usr/share/java
* Delete Jenkins war file
* Paste wget \*Jenkins war link older version you want to upgrade\*
* Restart Jenkins \*Systemctl restart jenkins\*

JENKINS MIGRATION FROM SERVER TO SERVER

* Install \*job import plugin\* in plugin manager in new server want to migrate jobs
* Add credentials in the old server to access new server login user name and pwd
* Add system tool config
* select \*job import\* add new service Jenkins URL and token credential
* Jenkins dashboard
* select job import plugin
* select the added server
* select the Query
* select the job to migrate
* Restart once

JENKINS BACKUP-Plugin

* Install \*Backup\* in plugin manager
* Manage Jenkins
* select backup manager
* setup the backup path,format,name,etc as what all required
* Backup manager-> backuphudson manager
* Backup now

JENKINS THIN BACKUP-Plugin

* Install \*thin Backup\* in plugin manager
* Manage Jenkins -> THIN BACKUP-Plugin settings->backup location path->file format->save
* Manage jenkins->backup now

JENKINS THIN RESTORE

* Manage Jenkins->restore
* Restart the Jenkins

JENKIS MASTER SLAVE CONCEPT WINDOWS AND LINUX

LINUX

* Create 2 EC2 Machines ( one for master another for slave ) use key pem
* Install Java and jenkins in Master EC2
* Install java only in slave and create a general directory -mkdir \*\*\*\* and change the directory and copy the present working directory
* Manage Jenkins in Master EC2 machine
* Select nodes->New node-name the server
* Paste the present working directory in \*Remote root directory\*
* Lable it \*\*\*slave1 \*\*\* as per pipeline call requirement
* Select \*Launch method\*->Launch agent via SSH -> Paste slave machine ip in \*HOST\*
* Add credentials as kind of SSH username with private key
* Type id and slave machine login name \*ubuntu and paste the pem file in private key directly option\*and add the same
* Host Key Verification Strategy \*select manual trusted key verification strategy\*
* Save it and refresh the nodes
* Go to Master EC2 and select the project -> Configure -> \*Restrict where this project can be run\*
* Select the label name as given of slave label and select required environments ->save and apply

REMOTE TRIGGER

* Select the project -> configuration-> \*Build Triggers\* -> Authentication token
* Paste Jenkins URL,job name,project name,token no **\* http://18.60.56.140:8080/job/project1\***
* Save and apply and -> \*Build now\*

GIT HUB WEBHOOK

* Select the Repo -> settings
* Select \*Webhooks\* ->add webhooks
* Confirm the access ->Payload url \*Jenkins ip <http://18.60.56.140:8080/github-webhook/>
* Add webhook

Poll SCM: periodically polls the SCM to check whether changes were made (i.e. new commits) and builds the project if new commits were pushed since the last build.

You must schedule the polling duration in the scheduler field. Like we explained above in the Build periodically section. You can see the Build periodically section to know how to schedule.

After successfully scheduled, the scheduler polls the SCM according to your specified duration in scheduler field and builds the project if new commits were pushed since the last build.