**Jenkins Guide and step by step process in windows**

**Jenkins Server installation docker file**

**ARG TAG=18362**

**FROM servercore:$TAG**

**ARG JENKINS\_VERSION=2.222.3**

**ENV VERSION $JENKINS\_VERSION**

**#COPY THE JENKINS INSTALLATION FILES INTO THE CONTAINER**

**RUN Powershell -command wget $('http://ftp-chi.osuosl.org/pub/jenkins/windows-stable/jenkins-{0}.zip' -f $env:VERSION) -outfile jenkins.zip ; \**

**Expand-Archive -Path jenkins.zip -DestinationPath c:\ ; \**

**rm jenkins.zip**

**#INSTALLING IIS**

**RUN Powershell.exe -Command Install-WindowsFeature Web-Server**

**#INSTALLING JENKINS**

**RUN ["msiexec.exe", "/i", "C:\\jenkins.msi", "/qn"]**

**Jenkins Agent installation docker file**

**FROM servercore:18362**

**ADD /jre1.8.0\_91 /jre1.8.0\_91**

**SHELL ["powershell"]**

**ARG BASE\_URL**

**ARG SECRET**

**RUN (New-Object System.Net.WebClient).DownloadFile('{0}/jnlpJars/agent.jar' -f $env:BASE\_URL, 'agent.jar') ;**

**ENTRYPOINT ["C:\\jre1.8.0\_91\\bin\\java.exe", "-jar", ".\\agent.jar"]**

**Running procedure**

1. Download JRE1.8.0\_91 zip file and extract it.
2. Create Folder name Jenkins and place above Jenkins server docker file in this folder.
3. Build docker file using **docker build -t Jenkins c:\jenkins –build-arg TAG=<build number> --build-arg JENKINS\_VERSION=2.222.3**
4. Run docker image using **docker run -it Jenkins PowerShell**
5. Get container Ip by using ipconfig
6. Open Jenkins server using http://<containerip>:8080.
7. Install recommended plugins
8. Provide login credentials
9. Create Folder name Jenkins client and place above Jenkins agent docker file in this folder.
10. Build dockerfile using docker build -t jenkinsagent c:\jenkinsclient --build-arg BASE\_URL=http://172.20.154.36:8080/
11. Add Node in server and copy jnlpUrl.
12. Run Jenkins agent using docker run -ti jenkins\_agent -jnlpUrl http://172.20.154.36:8080/computer/windows\_ci/slave-agent.jnlp -secret c206dbd5b1b9ecacb2b9c0c2762938b0200a00e55ae636d882b43a1cdce40820 -workDir "c:\jenkins"
13. Agent will be added and running now.

**Test Pipeline**

pipeline {

agent any

stages {

stage('Build') {

steps {

echo 'Building..'

}

}

stage('Test') {

steps {

echo 'Testing..'

}

}

stage('Deploy') {

steps {

echo 'Deploying....'

}

}

}

}

**Docker with Jenkins pipeline (windows not supported for now available in future)**

pipeline {

agent{

docker {

image 'dockerserver123/servercore:golang'

}

}

stages {

stage('Test') {

steps{

bat 'go version' (replace with sh for linux)

}

}

}

}