NEXUS SETUP AND INTEGRATED TO JENKINS

STEP-1: LAUNCH INSTANCE WITH T2.MEDIUM AND 8081 PORT AND 20 GB OF EBS VOLUME

STEP-2: INSTALL JAVA AS THE NEXUS DEPENDENCY

-→ (**yum install java-17-amazon-corretto -y)**

STEP-3: CREATE A FOLDER (**mkdir /app**)

SETP-4: GO TO FOLDER (**cd /app**)

STEP-5: DOWNLOAD THE NEUXS

wget -O nexus.tar.gz https://download.sonatype.com/nexus/3/latest-unix.tar.gz

STEP-6: UNTAR THE FILE (**tar -zxvf nexus.tar.gz)**

STEP-7: RENAME THE NEXUS FILE (**mv nexus-* nexus**)

STEP-8: ADD NEXUS USER (**useradd nexus**)

STEP-9: CHANGE THE OWNERS OF THE FILES

(**chown -R nexus:nexus file_names**)

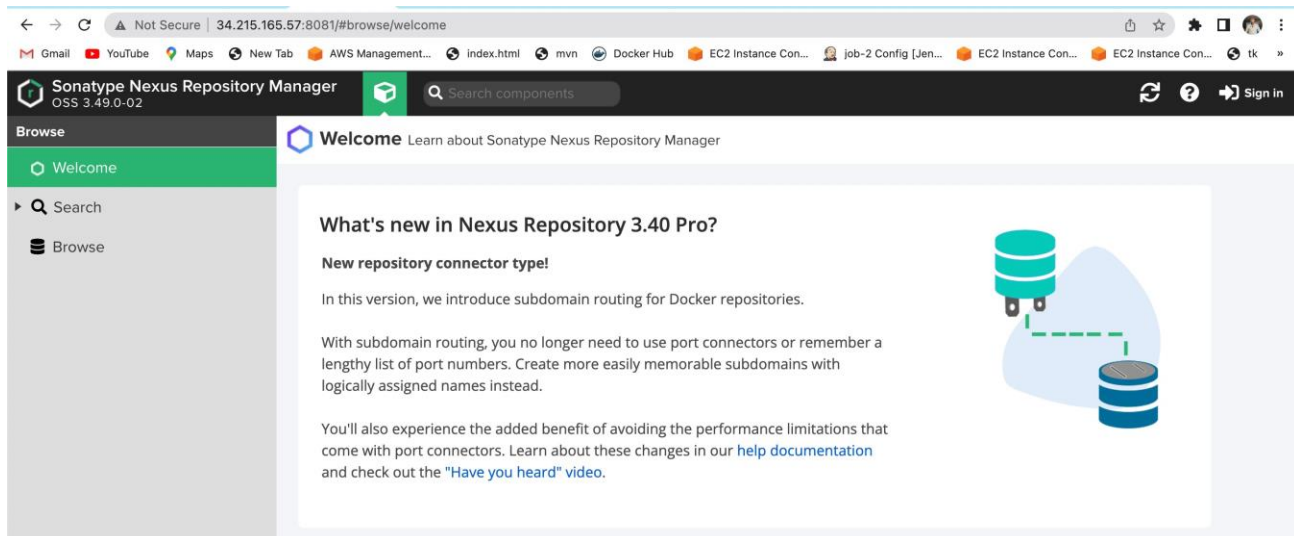STEP-10: CHANGE nexus.rc file

(**vim /app/nexus/bin/nexus.rc**)

run_as_user="nexus"   <---- inside nexus.rc

STEP-11: NOW START NEXUS SERVICE **(./nexus/bin/nexus start)**

AND SEE STATUS **(./nexus/bin/nexus status)**

-→ wait for 2 minutes and copy the **public-ip** of your nexus server and access it with **8081** port
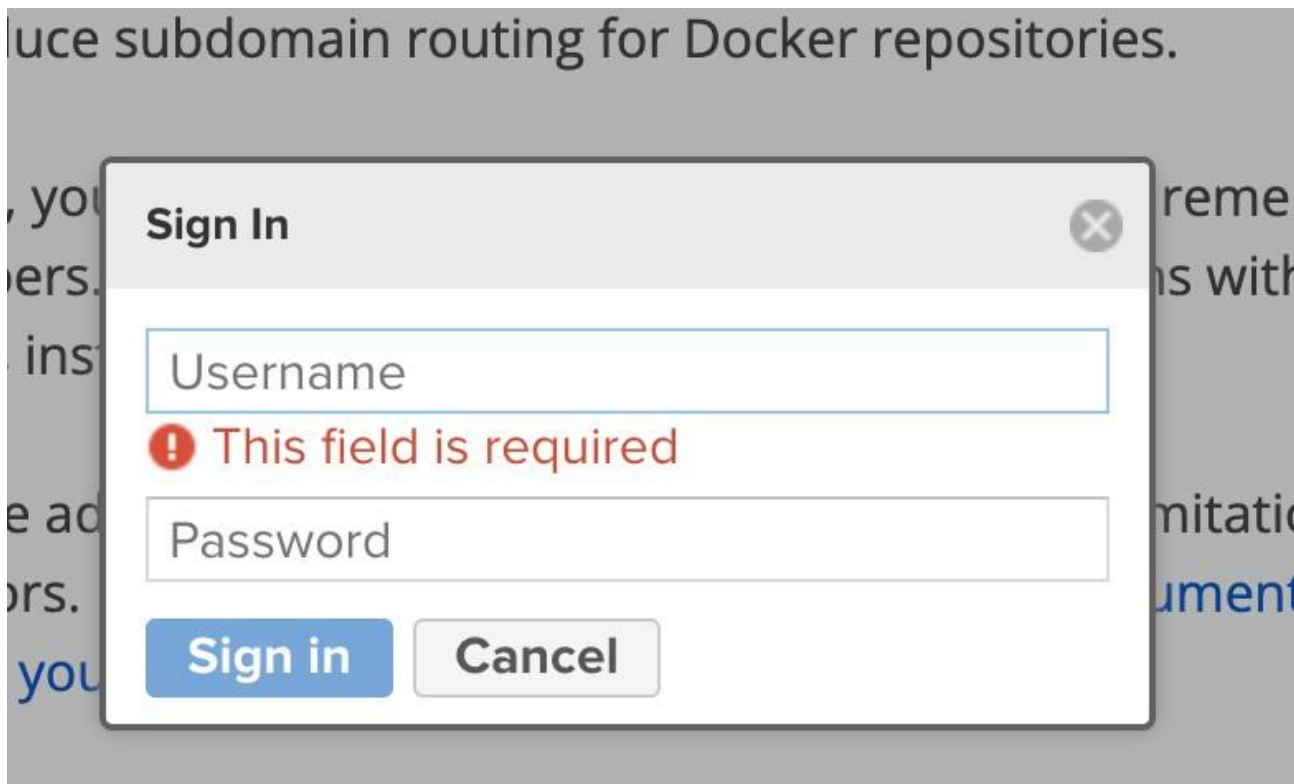
CLICK ON **SIGN-IN**

CLICK ON **SIGN-IN**

IT WILL ASKS THE USER NAME & PASSWORD

USERNAME**: admin**

PASSWORD: **cat /app/sonatype-work/nexus3/admin.password** use this command in our server to get the

password and paste it into our nexus dashboard

NOW GO TO SERVER ADMINISTRATION AND CONFIGURATION

SELECT REPOSITORIES

SELECT CREATE-REPOSITORY

Repositories / **Select Recipe**

Recipe ↑	
apt (hosted)	›
apt (proxy)	›
bower (group)	›
bower (hosted)	›
bower (proxy)	›
cocoapods (proxy)	›
conan (proxy)	›
conda (proxy)	›
docker (group)	›
docker (hosted)	›
docker (proxy)	›
gitlfs (hosted)	›
go (group)	›
go (proxy)	›
helm (hosted)	›
helm (proxy)	›
maven2 (group)	›
maven2 (hosted)	›
maven2 (proxy)	›
npm (group)	›
npm (hosted)	›

SELECT MAVEN2(HOSTED)

Repositories / **Select Recipe** / **Create Repository: maven2 (hosted)**

Name:
A unique identifier for this repository

🚫 This field is required

Online: ✓ If checked, the repository accepts incoming requests

Maven 2

Version policy:
What type of artifacts does this repository store?

Release ▾

Layout policy:
Validate that all paths are maven artifact or metadata paths

Strict ▾

Content Disposition:
Add Content-Disposition header as 'Attachment' to disable some content from being inline in a browser.

Inline ▾

GIVE REPOSITORY NAME AS myapp-releases and click on create repositories

now you can see our repository created in dashboard

NOW GO TO JENKINS AND INSTALL PLUGIN "**nexus artifactory uploader**"

ADD CREDENTIALS :

and add description as nexus credentials

CLICK ON TEST CONNECTION, THEN YOU WILL SEE THE MESSAGE

NOW GO AND CREATE A JOB

GIT: URL OF OUR PROJECT

ADD BUILD STEP: invoke top-level maven targets

Build Steps

≡ **Invoke top-level Maven targets** ?   ×

Goals

clean package   ▼

Advanced ⌄

ADD BUILD STEP: **Nexus artifact uploader**

≡ **Nexus artifact uploader**   ×

Nexus Details

Nexus Version

NEXUS3   ⌄

Protocol

HTTP   ⌄

Nexus URL ?

34.25.165.57:8081

Credentials

admin/****** (nexus)   ⌄

Add ▾

GroupId

in joughama

in.javahome

Version

8.2.4

Repository ?

ashokit-releases

Artifacts

Artifact   ✕

ArtifactId

myweb

Type ?

.war

Classifier ?

File ?

target/myweb-8.3.1.war

Add

NOW GO TO POST BUILD ACTIONS: deploy war/ear to container

Post-build Actions

Configure

⚙ General
⎇ Source Code Management
🕐 Build Triggers
🌐 Build Environment
☰ Build Steps
📦 Post-build Actions

☰ **Deploy war/ear to a container**   ✕

WAR/EAR files ?

target/*.war

Context path ?

swiggy

Containers

☰ **Tomcat 9.x Remote**   ✕

Credentials

tomcat/****** (tomcat credentials)   ⌄

Add ▾

Tomcat URL ?

http://52.25.184.150:8080/

NOW SAVE AND BUILD THE JOB IF THE BUILD GETS SUCCESS: then we

can see that war files are stored in nexus repository

==

Or or or or or or or or or or or or or or or

==

Install nexus **platform plugin** in jenkins

Go to jenkins —> configuration system —> nexus

Display name: nexus

Server-id: nexus-repo

Server url: public ip of nexus:8081

Credentials: ****

Test connection

Go to job:

Configure —>

Add a build step: **Nexus Repository Manager Publisher**

Nexus instance: auto update Nexus-repo:

sample-releases packages:

Group: search in pom.xlm

Artifact id: search in pom.xlm

Version: search in pom.xlm

Package: war

Artifact:

File path: target/myweb-8.2.5.war

Extension: .war