**NEXUS:**

**TYPES OF REPOSITORIES:**

* **Proxy** = proxy repo is proxy of remote repository. By default, nexus ships with the following configured proxy repositories.

**apache snapshots** = contains snapshots releases from apache software foundation.

**Codehaus snapshots** = contains snapshots releases from codehaus.

**Central** = it is known as maven central containing release components. It is default built-in repository for maven. Nexus directly connects to central repository via **HTTPS** usin this url **https://repo1.maven.org/maven2/.**

* **Hosted** = it is a repository which is hosted by repository manager. By default, nexus will come with some configured hosted repositories.

**3rd party** = it should be used for 3rd party dependencies, which are not available in maven central repo.

**Ex: oracle jdbc driver**.

**Releases =** it is where you organization will publish internal releases**.**

**Snapshots =** it is where you organization will publish internal snapshots.

**INSTALLATION:**

Prerequistes:

**Java**

**Min 5 GB ram**

* Download sonartype nexus tar from official website. Untar the file.
* Create user “**nexus**” in your linux server.
* Change **owner** and **group** permissions of nexus dir to nexus user.

**chown –R nexus:nexus nex-dir**.

* In nexus 3, Go to **/{NEXUX\_HOME}/bin/nexus.rc**
* Edit the **nexus.rc** file. Set **RUN\_AS\_USER=nexus**
* In nexus 2, Go to **/{NEXUX\_HOME}/bin/**
* Edit the **nexus** file**.** Set **RUN\_AS\_USER=nexus**
* Start the nexus server = **./bin/nexus start**
* Open nexus in your browser with your serverip and port 8081. **Serverip:8081**
* By default,

**username = admin**

**Password = admin123**

* After login to nexus, click on repositories to see the repositories list.
* You can create a new repository.

**Click create repsitory**

**Select type of repository = maven 2 (hosted).**

**Select version policy = release, snapshot (based on your repo type).**

**Click create repository.**

* After creating maven repositories, you have to integrate nexus with maven, to deploy artifacts to nexus to share them with other developers teams with in our organization.

**MAVEN-NEXUS INTEGRATION:**

* You have to specify distribution management section with nexus repository url’s.
* Go to pom.xml, at the end of build section. Paste the nexus url’s.

**<distributionManagement>**

**<repository>**

**<id>deployment</id>**

**<name>Internal releases</name>**

**<url>http://3.214.215.161:8081/repository/smaple\_release/</url>**

**</repository>**

**<snapshotRepository>**

**<id>deployment</id>**

**<name>snapshot releases</name>**

**<url>http://3.214.215.161:8081/repository/sample\_snapshot/</url>**

**</snapshotRepository>**

**</distributionManagement>**

Go to maven settings.xml, at the end of server deployment repo (🡪). Paste this.

**<server>**

**<id>deployment</id>**

**<username>admin</username>**

**<password>admin123</password>**

**</server>**

* In the username and password, mention your nexus credentials.
* Now, we have integrated nexus with maven. To check whether it is successful (or) not. Deploy the artifact with maven goal.

**mvn clean package deploy**

* After deploying, check in your nexus repository, whether the artifacts came (or) not.

**Mirrors** : which is used to download from arty which is present in nexus group

It should be place in maven/conf/settings.xml and delete the mirrors group and place exactly above of </settings>

<mirrors>

<mirror>

<id>central</id>

<name>central</name>

<url>http://your-host:8081/repository/maven-group/</url>

<mirrorOf>\*</mirrorOf>

</mirror>

</mirrors>

</settings> for place were to keep

And now configure your projects.

If you want only to download dependencies from Nexus, put this in the pom.xml:

<repositories>

<repository>

<id>maven-group</id>

<url>http://your-host:8081/repository/maven-group/</url>

</repository>

</repositories>

</project>

for to place exactly at this point

tomcat install

start the tomcat /tom/bin/start

copy the war file and place in /top/webapps its by using manvalli

**access tomcat from remote machine go to**

add normal user for tomcat to add

vi /opt/apache-tomcat-9.0.31/conf /tomcat-users.xml

and add user to it

vi /opt/apache-tomcat-9.0.31/conf/Catalina/localhost/ manager.xml

<Context privileged="true" antiResourceLocking="false"

docBase="${catalina.home}/webapps/host-manager">

<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="^.\*$" />

</Context>

**JENKINS-NEXUS INTEGRATION:**

* To integrate nexus with jenkins, you have to download **nexus plugin** from sonatype website and upload the plugin to jenkins.
* Nexus version 3 is still not supported by jenkins, you should have installed nexus 2 to integrate nexus with jenkins.
* After downloading the plugin (**nexus-jenkins-plugin.hpi**).

**Go to jenkins**

**Go to manage jenkins**

**Click manage plugins.**

**Go to advanced tab.**

**In upload plugin section, click upload file.**

**Choose your plugin from the system.**

**Click upload.**

**Once you upload the plugin, jenkins will install the plugin and waits for the restart to update the changes.**

**Restart the jenkins.**

* Now in the configure system, you will see sonatype nexus option.

**Click on “add nexus repo manager”**

**Mention displayname**

**Mention ID**

**Specify nexus url**

**Specify credentials**

**Click “ test connection”**

* If you have configured everything correctly , it wil show your nexus version and your hosted repositories in nexus.
* Now, we have integrated jenkins-nexus successfully. We have to deploy war to nexus from jenkins using jobs.

**Create a job,**

**In the postbuild step**

**Choose Nexus repository manager publisher**

**Nexus instance** = the name of nexus server which configured in configure system.

**Nexus repository** = select a repo from list which hosted in nexus server.

* Click on Add package, Choose maven package.

**Group** = specify the group id based on pom.xml

**Artifact** = specify the artifact id based om pom.xml

**Version** = specify version based on pom.xml

**Packaging** = specify type of package based on pom.xml (war, ear, jar)

* Click **Add** **Artifact** **Path**. Choose **maven** **artifact**.
* Specify the **war file path** (jenkins workspace).
* Save the job and build it.
* In the console, you will the output like jenkins is deploying war to nexus.
* Once the job is finished building successfully, You can check in your nexus repository for the war.