**Nexus Repository:**

* **Nexus is an Opensource Software & free**
* **Nexus developed by using java**
* **Nexus developed by Sonatype**
* **Nexus is called as artifactory server**
* **It is used to store and retrieve build artifacts and shared libraries**

**Note:**

* **Build artifacts are nothing but our project package files ( jar or war)**
* **Why are we storing the build artifacts on nexus ?**

**For feature purpose / backup purpose to deploy the application.**

* **Share libraries (jars) means the jars which are used across multiple projects within the company.(Common Jar files)**

**What is different between Nexus and GitHub?**

1. **GitHub is SCM Software which is used to store Source code of the project**
2. **Nexus is Artifact Repository which is used to store the build artifacts**

**Nexus Setup:**

* **Take t2.medium instance**
* **Java s/w is required to install Nexus**
* **Connect to t2.medium instance using mobaxterm or putty**

For ubuntu,

**$ sudo su -**

**$ cd /opt**

**$ sudo apt update**

**$ sudo apt install openjdk-8-jdk**

**$ sudo su -**

**$ cd /opt**

**$ yum install java -y**

sudo dnf install java-1.8.0-amazon-corretto

alternatives --config java

**Links to download :** [**https://help.sonatype.com/repomanager3/product-information/download**](https://help.sonatype.com/repomanager3/product-information/download)

**Lates versions:** [**https://download.sonatype.com/nexus/3/nexus-3.53.1-02-unix.tar.gz**](https://download.sonatype.com/nexus/3/nexus-3.53.1-02-unix.tar.gz)

**$ tar -xvzf nexus-3.53.1-02-unix.tar.gz**

**$ mv /opt/nexus-3.53.1-02 /opt/nexus ( instead of using lengthy name )**

**#As a good security practice, Nexus is not advised to run nexus service as a root user, so create a new user called nexus and grant sudo access to manage nexus services as follows.**

**$ useradd nexus**

**#Give the sudo access to nexus user**

**$ visudo  
nexus ALL=(ALL) NOPASSWD: ALL**

**#Change the owner and group permissions to /opt/nexus and /opt/sonatype-work directories.**

**$ chown -R nexus:nexus /opt/nexus**

**$ chown -R nexus:nexus /opt/sonatype-work**

**$ chmod -R 775 /opt/nexus**

**$ chmod -R 775 /opt/sonatype-work**

**# Open /opt/nexus/bin/nexus.rc file and uncomment run\_as\_user parameter and set as nexus user.**

**$ vi /opt/nexus/bin/nexus.rc**

**run\_as\_user="nexus"**

**# Create nexus as a service**

**$ ln -s /opt/nexus/bin/nexus /etc/init.d/nexus**

**# Switch as a nexus user and start the nexus service as follows.**

**$ su - nexus**

**# Enable the nexus services**

**$ sudo systemctl enable nexus**

**# Start the nexus service**

**$ sudo systemctl start nexus**

**#Access the Nexus server from Laptop/Desktop browser.**

**URL : http://IPAddess:8081/**

**Note: Enable this 8081-port number in Security Group**

**# Default Username**

**Username: admin**

**# We can copy nexus password using below command**

**$ sudo cat /opt/sonatype-work/nexus3/admin.password**

**🡺 We can change PORT Number by feature in nexus default properties**

**/opt/nexus/etc/nexus-default.properties**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Integrate Maven with Nexus:**

**Create Repositories in Nexus to store build artifacts**

**We will create 2 types of repositories in Nexus**

**1) snapshot**

**2) release**

**If project is under development, then that project build artifacts will be stored into snapshot repository**

**If project development completed and released to production, then that project build artifacts will be stored to release repository**

* **Open Pom.xml and check the version for sample.**

**Note: Based on <version/> name available in project pom.xml file it will decide artifacts should be stored to which repository**

**Step-1  
Create Snapshot Repositories:**

**Create Repositories🡺 Repositories 🡺 Create Repositories 🡺 Create name 🡺Version Policy (Snapshot)🡺 Deployment Policy (Allow Redeploy) 🡺Create Repositories**

**Then copy the URL :** [**http://18.119.141.42:8081/repository/arunit-snapshot/**](http://18.119.141.42:8081/repository/arunit-snapshot/)

**Step-2  
Create Release Repositories:**

**Create Repositories🡺 Repositories 🡺 Create Repositories 🡺 Create name 🡺Version Policy (Release)🡺 Deployment Policy (Allow Redeploy) 🡺Create Repositories**

**Then copy the URL:** [**http://18.119.141.42:8081/repository/arunit-realserepo/**](http://18.119.141.42:8081/repository/arunit-realserepo/)

**Step-3  
We need to establish the connectivity from Maven Project to Nexus Repositories.**

**Nexus Repositories Details we will configure in project POM.XML file like below.**

**<distributionManagement>**

**<repository>**

**<id>nexus</id>**

**<name>praveen IT Releases Nexus Repo</name>**

**<url>**http://52.194.183.131:8081/repository/release/**</url>**

**</repository>**

**<snapshotRepository>**

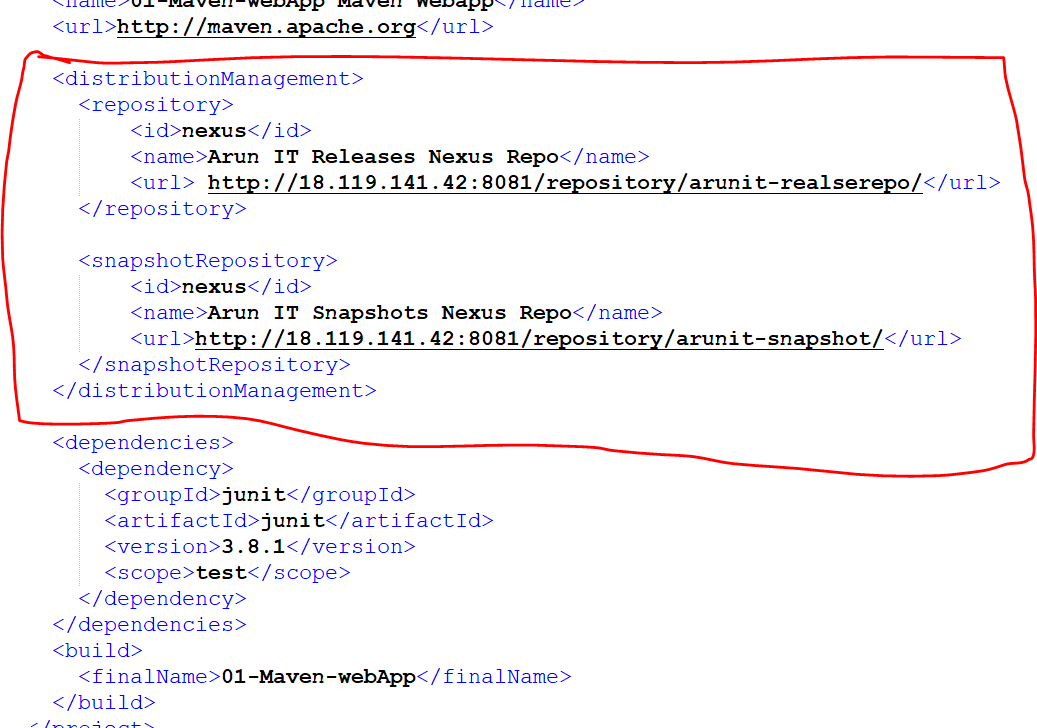
**<id>nexus</id>**

**<name>praveen IT Snapshots Nexus Repo</name>**

**<url>**http://52.194.183.131:8081/repository/snapshot/**</url>**

**</snapshotRepository>**

**</distributionManagement>**



1. **Nexus Server Credentials will be configured in Maven "settings.xml" file**
2. **Maven Location : C:\apache-maven-3.8.5\conf**
3. **In settings.xml file, under <servers> tag add below <server> tag**

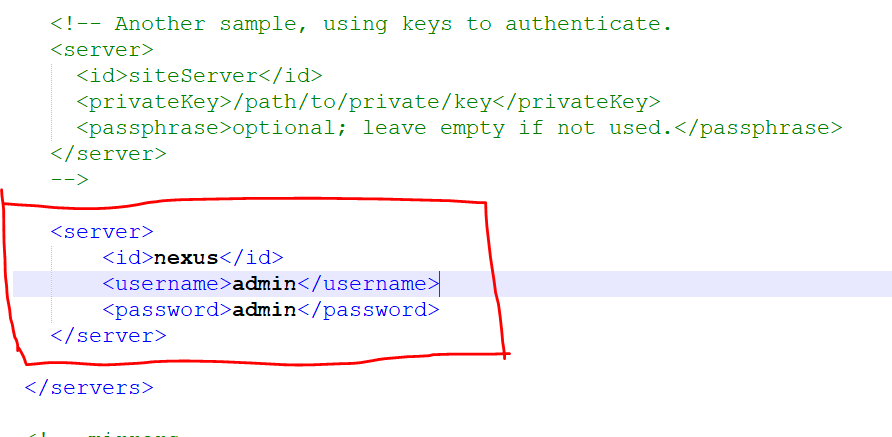
**<server>**

**<id>nexus</id>**

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

**<password>admin</password>**

**</server>**



**Once these details are configured then we can run below maven goal to upload build artifacts to Nexus Server**

**$ mvn clean deploy**

**Note: When we execute maven deploy goal, internally it will execute 'compile + test + package + install + deploy' goals.**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Remote Repository:**

* **Remote Repositories are used to maintain shared libraries in the company**
* **Shared libraries means the jars which are required for several project in the company**
* **Shared libraries are specifically to only that company projects.**
* **One Company shared libraries cannot access another company**

**What is the benefits of Remote Repository?**

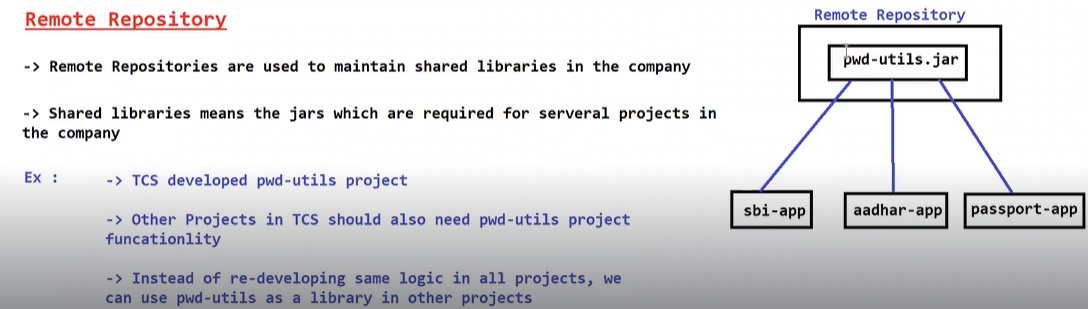
**To Share the common libraries for multiple Project**

**Example:**

**🡺 TCS Developed pwd-utils project**

**🡺Others Project in TCS should also need pwd-utils project functionality**

**🡺Instead of re-developing same logic in all projects , we can use pwd-utils as library in other project .**



* **Remote Repo used for shared libraries**
* **If we want to use few jars files in multiple projects in the company, then we will use remote repository**
* **Remote repository is specific for our company project**
* **Create Remote repos in Nexus and upload a jar fil**
  + **Go to Repositories**
  + **Create New Repositories**
  + **Choose Maven (Hosted Repositories)**
  + **Give a name for Repositories (arunitremoterepo)**

**Note: with above steps Remote Repositories got created**

* **Go to Browse Section**
* **Select Remote Repositories (By default is empty)**
* **Upload jar file with group id , artifactory id , Version**

**Group id: greenit .in**

**Artifactid: pwd-utils**

**Version: 1.0**

* **Take dependency details of uploaded jar file and in project pom.xml as dependency like below**

<dependency>

<groupId>greens.it</groupId>

<artifactId>pwd\_utilis</artifactId>

<version>1.0</version>

</dependency>

* **We need to add Remote Repository Details in pom.xml above <dependency/> tag**

**<repositories>**

**<repository>**

**<id>nexus</id>**

**<url>http://43.207.176.198:8081/repository/remote/url>**

**</repository>**

**</repositories>**

A screenshot of a computer

Description automatically generated