

# Nexus(8081)

## Introduction:

Nexus is a Artifactory repository manager allows to store and retrieve build artifacts.

Nexus Repository Manager is available from Sonatype, Inc. "Sonatype" and "Sonatype Nexus" are trademarks of Sonatype, Inc.

## Pre requisites :

1. We need to take t2.large ec2-instance in AWS.
2. First we need install java on ec2-instance.
3. Install Nexus on ec2-instance.

## Nexus Installation Steps:

4. Step1: Install java on ec2
5. yum install -y java-1.8.0-openjdk-devel.x86\_64
6. Step2 : Install Nexus on ec2

Go to /opt directory

cd /opt and download nexus from nexus site like below command

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

and unzip the file latest-unix.tar.gz by using tar

```
tar -xvf latest-unix.tar.gz
```

rename extracted filename to nexus3 like below

```
mv nexus-3.20.1-01/ nexus3
```

Change ownership to nexus3 and sonatype-work like below

```
chown -R ec2-user:ec2-user nexus3 sonatype-work
```

go to nexus3 directory under nexus3 directory after that go to bin directory

and edit file nexus.rc in that file comment out and give user as ec2-user like below steps.

```
cd nexus3/
```

```
cd bin
```

```
ls
```

```
vi nexus.rc ----- In this like run_as_user="ec2-user"
```

after that do softlink to opt/nexus3/bin/nexus /etc/init.d/nexus

```
ln -s /opt/nexus3/bin/nexus /etc/init.d/nexus
```

and make it as nexus as a service like this

```
cd /etc/init.d/  
chkconfig --add nexus  
chkconfig nexus on  
Start service of nexus  
sudo service nexus start
```

### Accessing From Browser :

Ec2-instance Public IP address:portnumber for ex: 30.40.5.3:8081

You will get default user as admin and password like below path.

```
cat /opt/sonatype-work/nexus3/admin.password
```

cfe5e876-ffbb-4b2f-a7b3-38a369f15136 -----→ like this



and reset your password

### Creating Repositories:

Go to settings →select repository →create repository→one for sample-relases

and one for sample-snapshots like below steps.

Create repository and maven(2hosted)-→give reponame → sample-relases

And give version policy →release →create repository

**Name:** A unique identifier for this repository

sample-releases

**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

Similar way Create repository for sample-snapshots like below

Create repository and maven(2hosted)-→give reponame → sample-snapshots

And give version policy →snapshot →create repository

**Name:** A unique identifier for this repository

sample-snapshots

**Online:** ☒ If checked, the repository accepts incoming requests

## Maven 2

### Version policy:

What type of artifacts does this repository store?

Snapshot

### Layout policy:

Validate that all paths are maven artifact or metadata paths

Strict

## Jenkins Integrate with Nexus:

Step1: Go to Jenkins Server provide Nexus Credentials in /etc/maven/settings.xml like below

```
<server>
  <id>nexusRepo</id>
  <username>admin</username>
  <password>admin123</password>
</server>
```

Step2:

Provide sample-releases and sample-snapshot urls under distribution tag in POM.xml

```
7
3      <distributionManagement>
3          <snapshotRepository>
3              <id>nexusRepo</id>
1              <url>http://13.232.58.254:8081/repository/sample-snapshots</url>
2          </snapshotRepository>
3
4          <repository>
5              <id>nexusRepo</id>
5              <url>http://13.232.58.254:8081/repository/sample-releases</url>
7          </repository>
3      </distributionManagement>
3
```

Step3:

Create Jenkins job under Free-style under SCM and Build like below screen shots

The screenshot shows the 'Source Code Management' configuration page in Jenkins. Under the 'SCM' section, 'Git' is selected. The 'Repositories' section contains one repository with the URL 'https://github.com/srinivas1987devops/myweb.git' and credentials set to '- none -'. The 'Branches to build' section has a branch specifier of '\*/master'. Buttons for 'Advanced...', 'Add Repository', and 'Add Branch' are visible.

Source Code Management	
<input type="radio"/> None	
<input checked="" type="radio"/> Git	
Repositories	<div>Repository URL: <input type="text" value="https://github.com/srinivas1987devops/myweb.git"/></div> <div>Credentials: <input type="text" value="- none -"/> <span>Add</span></div> <div><span>Advanced...</span></div> <div><span>Add Repository</span></div>
Branches to build	<div>Branch Specifier (blank for 'any'): <input type="text" value="*/master"/></div> <div><span>Add Branch</span></div>

**Build**

Invoke top-level Maven targets

Goals

clean deploy

▼

Advanced...

Add build step ▼

Step4: Finally Build the Job and you will get Artifatcs under Nexus..

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>in.javahome</groupId>
  <artifactId>myweb</artifactId>
  <packaging>war</packaging>
  <version>8.2.0-SNAPSHOT</version>
  <name>Java Home myweb</name>
  <url>http://maven.apache.org</url>
```