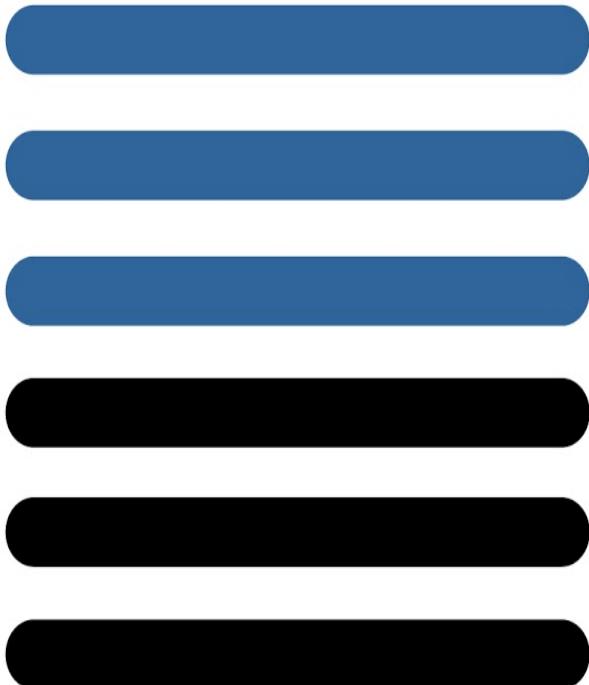


Installing Sonatype Nexus (Version 2.1) & Required Dependencies

Step By Step Guide *demonstrating a shared Nexus repository as part of a continuous integration pipeline*



Note: Copy & Paste commands into terminal window from link below
https://raw.githubusercontent.com/louyiheidar/ExerciseTasksLH/master/SonatypeNexusInstallation/Commands_PlainTextFormat

Installation of Java 8 (Ubuntu 14.04 LTS)

Nexus Repository Manager requires a Java 8 Runtime Environment (JRE)

Step 1)

```
# Add the WebUpd8 Team Personal Package Archive (PPA)  
$ sudo apt-add-repository ppa:webupd8team/java
```

Step 2)

```
# Update packages  
$ sudo apt-get update
```

Step 3)

```
# Install Oracle Java 8 with the PPA installer  
$ sudo apt-get install oracle-java8-installer  
  
# Verify Java is installed & version is 1.8.x
```

```
vagrant@agentansible: ~  
vagrant@agentansible:~$ java -version  
java version "1.8.0_131"  
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)  
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
```

Installation of Git

Note: Jenkins requires Git installed as a dependency

```
# Install Git by using Ubuntu's default repositories  
$ sudo apt-get update  
$ sudo apt-get install git  
  
# Verify Git is installed
```

```
vagrant@masteransible: ~  
vagrant@masteransible:~$ git --version  
git version 1.9.1
```

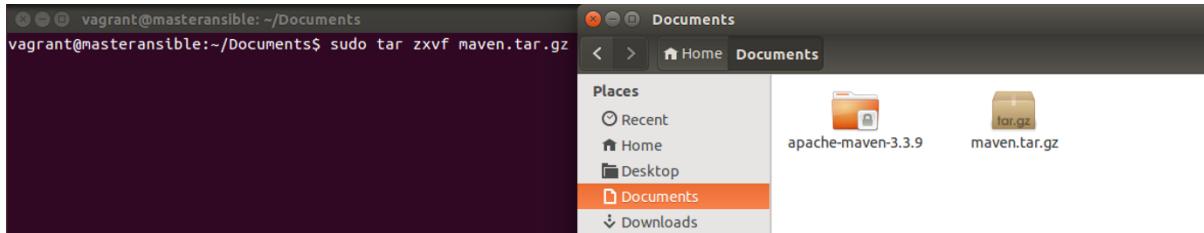
Installation of Maven

Step 1)

```
# Download Maven using link below  
https://www.dropbox.com/s/1p07os3l44h7rqj/maven.tar.gz?dl=0
```

Step 2)

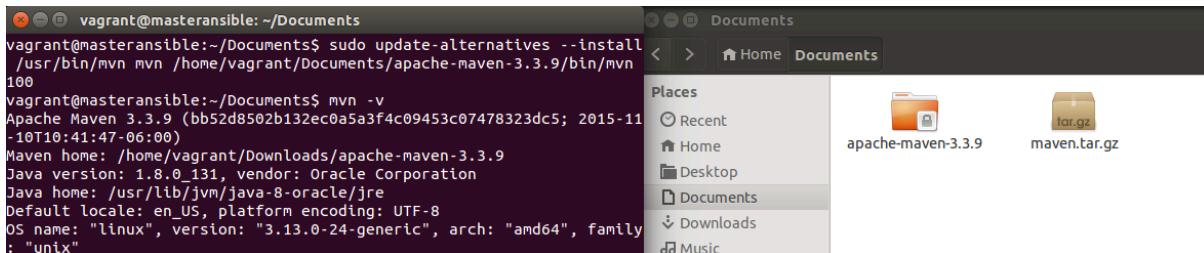
```
# Navigate to file directory and extract maven.tar.gz  
$ sudo tar zxvf maven.tar.gz
```



Step 3)

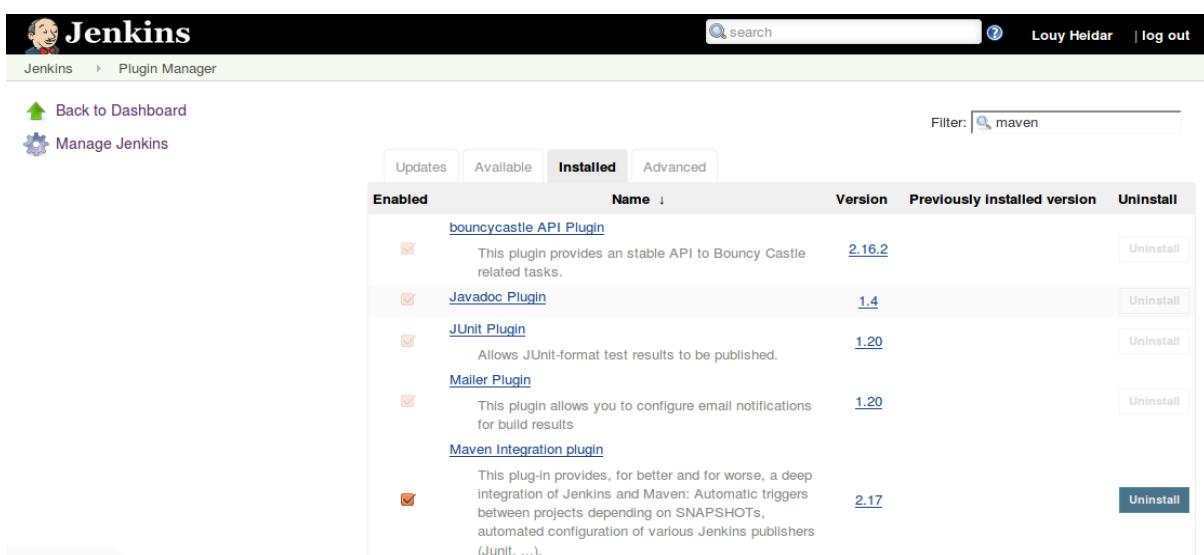
```
# Set Maven to Environment Path (command below executed in a single line)  
$ sudo update-alternatives --install /usr/bin/mvn mvn  
/home/vagrant/Documents/apache-maven-3.3.9/bin/mvn 100
```

```
# Verify Maven is installed & configured correctly  
$ mvn -v
```

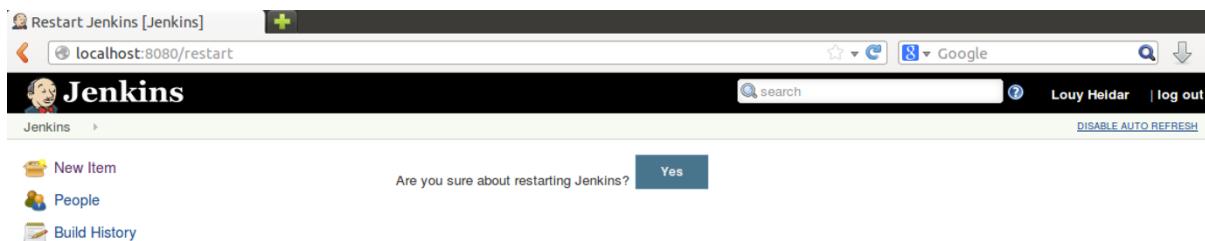


Step 4)

```
# Install Maven Integration Plugin  
Jenkins > Manage Jenkins > Manage Plugins
```

A screenshot of the Jenkins Plugin Manager interface. The 'Installed' tab is selected. A search bar at the top right is set to 'maven'. The table lists several plugins, with the 'Maven Integration plugin' highlighted. The table columns are: Enabled, Name, Version, Previously Installed version, and Uninstall.

```
# Restart Jenkins  
localhost:8080/restart
```



Installation of Nexus

Step 1)

Note: In order to install the open source version of Nexus you need to visit Nexus OSS and download the **2.x zip** version.

Installed & tested version is **nexus-2.14.4-03-bundle.zip**

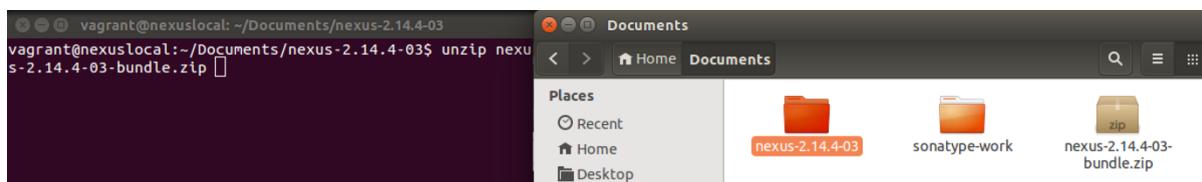
<http://www.sonatype.org/nexus/go/>

Step 2)

```
# Install 'Unzip'  
$ sudo apt-get install unzip
```

Step 3)

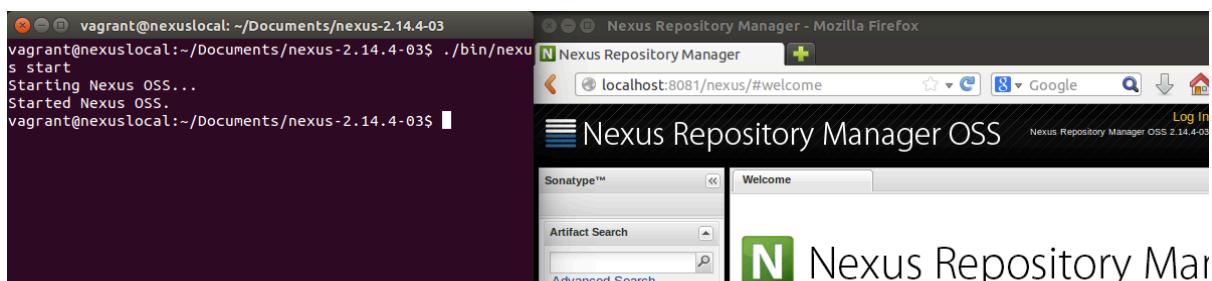
```
# Extract the archive (nexus-2.14.4-03-bundle.zip)  
$ unzip nexus-2.14.4-03-bundle.zip
```



Step 4)

Start Nexus, after extracting, the nexus script must be executed with a start parameter in the nexus folder

```
$ ./bin/nexus start
```



Note: To stop Nexus you just have to write stop instead of start

```
$ ./bin/nexus stop
```

Step 5)

<http://localhost:8081/nexus>

The screenshot shows the Nexus Repository Manager OSS interface. The title bar says "Nexus Repository Manager - Mozilla Firefox". The main content area features a large "Nexus Repository Manager OSS" logo. Below it is a search bar with placeholder text: "Type in the name of a project, class, or artifact into the text box below, and click Search. Use 'Advanced Search' on the left for more options." To the right of the search bar is a "Log In" link and the text "Nexus Repository Manager OSS 2.14.4-03". On the left, there is a sidebar titled "Sonatype™" with sections for "Artifact Search", "Advanced Search", "Views/Repositories", "Repositories", and "Help". The main content area also includes a "See What's New" section with a "More Improvements and Bug Fixes" heading and a "Download 3.4.0 Now" button. A sidebar on the right is titled "Get Nexus Resources" with links for "Documentation" and "Knowledge Base". At the bottom of the page, there is a link to "Register for All Day DevOps 2017".

Advanced Network Troubleshooting – Optional Steps

(Nexus user interface not displaying)

Create an SSH Tunnel to your Nexus server.

```
# Run the following to start a tunnel (this assumes you are running nexus on  
the default 8081 port on your server with hostname 'servername')
```

```
ssh -L 8081:localhost:8081 servername
```

```
vagrant@masteransible: ~/Downloads/nexus-3.4.0-02$ ssh -L 8081:localhost:8081 8.8.8.8
```

```
# Locate servername
```

```
nslookup
```

```
> dig +short -x 172.17.25.208
```

```
vagrant@masteransible:~$ nslookup  
> dig +short -x 172.17.25.208  
Server:      8.8.8.8  
Address:     8.8.8.8#53
```

<http://localhost:8081/nexus>

The screenshot shows the Nexus Repository Manager OSS 2.14.4-03 interface. The top navigation bar includes links for Sonatype, Nexus Repository Manager, and a search bar pointing to localhost:8081/nexus/#welcome. On the right, there are icons for Log In, Help, and Home. The main content area features the Nexus logo and the text "Nexus Repository Manager OSS". Below this is a search bar with placeholder text: "Type in the name of a project, class, or artifact into the text box below, and click Search. Use 'Advanced Search' on the left for more options." A note at the bottom encourages users to share their feedback. The left sidebar contains sections for Artifact Search, Advanced Search, Views.Repositories, Repositories, and Help.

Nexus User Settings

Step 6)

Login as Administrator from Nexus

Default user is 'admin' and password is 'admin123'

Step 7)

Create Nexus User Profile

Nexus Repository Manager OSS > Security > Users > Add > Save

The screenshot shows the 'Users' configuration screen in the Nexus Repository Manager. The top navigation bar indicates the user is 'admin'. The left sidebar has a red circle around the 'Users' link under the 'Security' section. The main content area displays a table of existing users and a form for adding a new user named 'louyheidar'. The 'Role Management' section on the right shows various roles like Analytics, Metrics Endpoints, and Nexus Administrator Role, with a red circle around the 'Save' button. The URL in the browser is http://localhost:8081/nexus/#/users/louyheidar.

User ID	Realm	First Name	Last Name	Email	Status	Roles
admin	default	Administrator	User	changeme@yourcompany.com	Active	Nexus Administrator Role
deployment	default	Deployment	User	changeme1@yourcompany.com	Active	Repo: All Repositories (Full Control), Nexus Deplo...
louyheidar	default	Louy	Heidar	louy.heidar@qa.com	Active	Repo: All Maven1 Repositories (View), Repo: All ...
anonymous	default	Nexus	Anonymous User	changeme2@yourcompany.com	Active	Repo: All Repositories (Read), Nexus Anonymou...

Jenkins Configurations

Step 8)

Download Nexus Jenkins Plugin

(nexus-jenkins-plugin-1.1.0-05.hpi)

<http://download.sonatype.com/nexus/ci/latest.hpi>

Step 9)

Install Nexus Jenkins Plugin

Jenkins > Manage Jenkins > Manage Plugins

Enabled	Name	Version	Previously Installed version	Uninstall
<input checked="" type="checkbox"/>	bouncycastle API Plugin	2.16.2		<button>Uninstall</button>
<input checked="" type="checkbox"/>	Credentials Plugin	2.1.14		<button>Uninstall</button>
<input checked="" type="checkbox"/>	Nexus Platform Plugin	1.1.0-05		<button>Uninstall</button>
<input checked="" type="checkbox"/>	Pipeline: Step API	2.12		<button>Uninstall</button>

Step 10)

Configure Settings for 'Nexus Repository Manager Server' & Test for Successful Connection

Jenkins > Manage Jenkins > Configure System

Breakdown Instructions:

1. In the Sonatype Nexus section, click the Add Nexus Repository Manager Server dropdown menu and then select Nexus Repository Manager 2.x Server.

Enter the following (Text Field Inputs):

- **Display Name:** Name of the server you want shown when selecting Nexus Repository Manager instances for build jobs.
- **Server ID:** A unique ID used to reference Nexus Repository Manager in Build Pipeline scripts. It should be alphanumeric without spaces.
- **Server URL:** Location of your Nexus Repository Manager server.
- **Credentials:** Select the Add button to enter your Nexus Repository Manager username and password using the Jenkins Provider Credentials: Jenkins modal

window. Once added, select your Nexus Repository Manager username and password from the Credentials dropdown list.

2. Click the Test Connection button.

3. After a successful connection to Nexus Repository Manager, click the Save button

Game of Life (GOL)

Note: Continuous Integration (CI) is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.

An Example GitHub repository will be used to integrate both Jenkins and Nexus. Jenkins will be used as a build tool for GOL project. The project will then be uploaded to a shared nexus repository as part of a continuous integration pipeline.

Step 11)

Fork or Clone the example repository below to your personal GitHub account to edit and apply changes to Pom.xml file

<https://github.com/wakaleo/game-of-life>

Step 12)

Add Server Configurations to Pom.xml file

Example Repository with applied changes below:

<https://github.com/louyiheidar/gameoflife/blob/master/pom.xml>



```
473             <groupId>gr.spinelli</groupId>
474             <artifactId>UMLGraph</artifactId>
475             <version>4.6</version>
476         </dependency>
477     </dependencies>
478     <build>
479         <plugins>
480             <plugin>
481                 <groupId>org.apache.maven.plugins</groupId>
482                 <artifactId>maven-jxr-plugin</artifactId>
483                 <version>2.1</version>
484             </plugin>
485         </plugins>
486     </build>
487 
```

```
488     <distributionManagement>
489         <repository>
490             <id>releases</id>
491             <name>Releases</name>
492             <url>localhost:8081/nexus/content/repositories/releases</url>
493         </repository>
494         <snapshotRepository>
495             <id>snapshots</id>
496             <name>Snapshots</name>
497             <url>localhost:8081/nexus/content/repositories/snapshots</url>
498         </snapshotRepository>
499     </distributionManagement>
500 
```

```
501 </project>
```

Online XML Validators can be used for troubleshooting syntax issues before running the applied changes for the Pom.xml file

<https://codebeautify.org/xmlvalidator>

The screenshot shows the Code Beautify XML Validator interface. At the top, there's a browser header with 'Secure https://codebeautify.org/xmlvalidator'. Below it is the 'Code Beautify' logo and a navigation bar with links like 'JSON Formatter', 'My Ip', 'Search', 'Recent Links', 'Sample', 'More', 'Sign in', and a help icon. The main area is titled 'XML Validator' with a 'Save & Share' button. A large text area contains a multi-line XML code snippet, with line numbers from 490 to 506 on the left. Below the text area are four buttons: 'Validate', 'Load URL', 'Browse', and 'Download'. To the right of these buttons is a 'Sample X' button with a trash icon. At the bottom, a green bar indicates 'Valid XML'. Below the bar is a table with four columns: 'Total Character' (14908), 'Total Word' (547), 'Total Lines' (506), and 'Size' (15.05 KB).

Total Character	Total Word	Total Lines	Size
14908	547	506	15.05 KB

Details used for Server Configurations (Pom.xml)

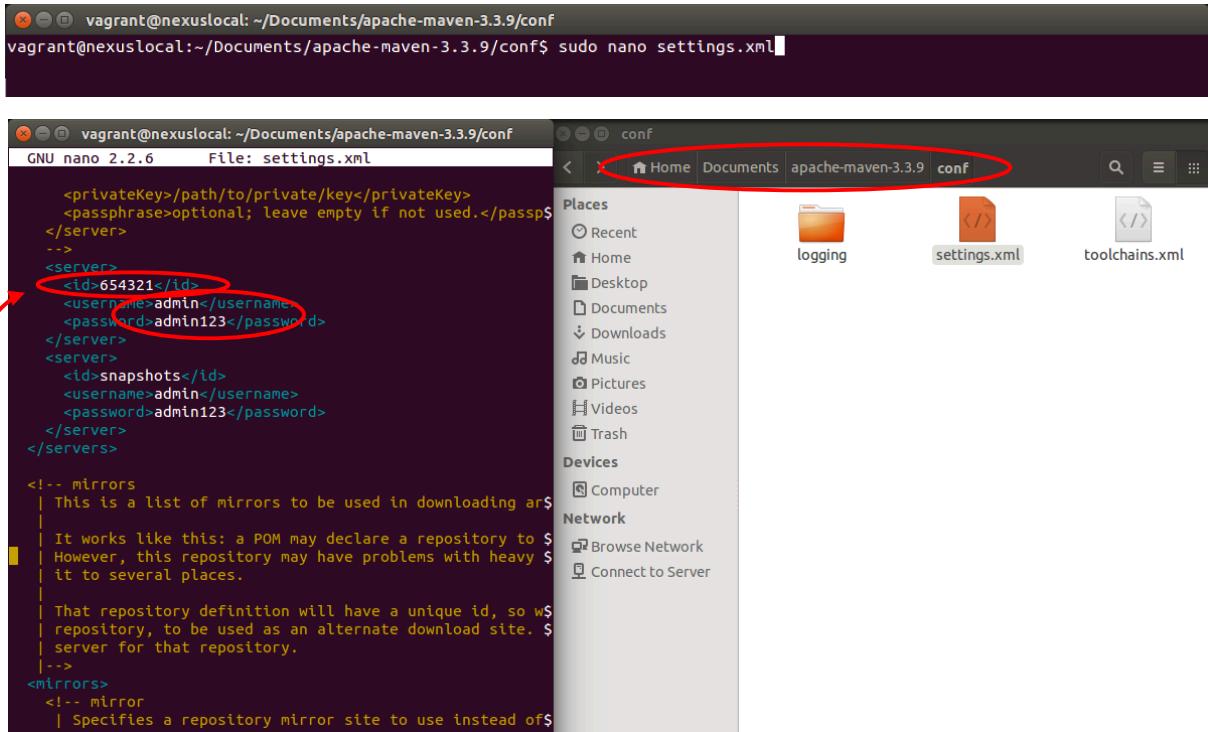
Nexus Repository Manager OSS > Views.Repositories > Repositories > Releases > Configurations

The screenshot shows the Nexus Repository Manager OSS interface. The top navigation bar includes 'Welcome', 'Users', 'Repositories', and a search bar. On the left, a sidebar menu lists 'Artifact Search', 'Advanced Search', 'Views/Repositories' (with 'Repositories' highlighted and circled in red), 'Repository Targets', 'Routing', 'System Feeds', 'Security', 'LDAP Configuration', 'Privileges', 'Roles', 'Users', 'Administration', 'Analytics', 'Capabilities', 'Logging', 'Plugin Console', 'Scheduled Tasks', and 'Server'. The main content area shows a table of repositories: Central (proxy, maven2, Release, In Service, http://localhost:8081/nexus/content/repositories/central), Central M1 shadow (virtual, maven1, Release, In Service, http://localhost:8081/nexus/content/shadows/central-m1), GOLNexus (proxy, maven2, Release, In Service, http://localhost:8081/nexus/content/repositories/LocalNexus), Releases (hosted, maven2, Release, In Service, http://localhost:8081/nexus/content/repositories/releases), and Snapshots (hosted, maven2, Snapshot, In Service, http://localhost:8081/nexus/content/repositories/snapshots). Below the table, under 'Releases', the 'Configuration' tab is selected (circled in red). The configuration form includes fields for 'Repository ID' (releases), 'Repository Name' (Releases), 'Repository Type' (hosted), 'Provider' (Maven2), 'Format' (maven2), and 'Repository Policy' (Release). A note says 'Default Local Storage Location' is file:/home/vagrant/Downloads/sonatype-work/hexus/storage/releases/. To the right, a 'Releases Configuration' section is shown. At the bottom, another 'Snapshots' tab is selected (circled in red) in a similar configuration form, with a 'Configuration' tab also circled in red. A 'Schemas' tab is also present. To the right, a 'Snapshots Configuration' section is shown.

Step 13)

Locate (settings.xml) and set the server configurations for Maven

Define settings for 'settings.xml' file (ServerID, Username, Password)



Matching Jenkins Settings

Sonatype Nexus

Nexus Repository Manager Servers

Nexus Repository Manager 2.x Server

Display Name: GOLNexus

Server ID: **654321**

Server URL: http://localhost:8081/nexus/

Credentials: admin/*****

Jenkins > Configure System > Sonatype Nexus

Test connection

Step 14)

Set Maven Home Directory Location & Apply Changes

Jenkins > Global Tools Configuration > Maven

Jenkins > Global Tool Configuration

Maven

Maven installations

Maven Name: MAVEN_HOME

MAVEN_HOME: /home/vagrant/Documents/apache-maven-3.3.9/

Install automatically

Delete Maven

Add Maven

List of Maven installations on this system

Step 15)

Create a new Freestyle Project

The screenshot shows the Jenkins interface for creating a new project. The title bar says 'Jenkins'. Below it, there's a search bar and a user 'Louy Heldar | log out'. The main area has a heading 'Enter an item name' with a required field 'GOLNexus'. Below this are two options: 'Freestyle project' (selected) and 'Maven project'.

Step 16)

After Creating Project, Set 'Repository URL' with the 'Jenkins credentials'

The screenshot shows the 'Source Code Management' tab for the 'GOLNexus' project. The 'Git' option is selected. Under 'Repositories', a single repository is configured with a 'Repository URL' of 'https://github.com/louyheldar/gameoflife.git' and 'Credentials' set to 'admin/*****'. There are buttons for 'Advanced...', 'Add...', and 'Add Repository'.

Under the Build Tab, Select 'Invoke top-level Maven Target'

Select the set name for Maven configuration (Maven Version) & Create Clean Package (Goals)

The screenshot shows the 'Build' tab for the 'GOLNexus' project. It includes sections for 'Invoke top-level Maven targets', 'Maven Version' (set to 'MAVEN HOME'), and 'Goals' (set to 'clean package'). There are red circles highlighting the 'Maven Version' and 'Goals' fields.

Step 17)

Build Tab, Select 'Nexus Repository Manager Publisher' & Configure Settings

Nexus Repository Manager Publisher

Nexus Instance: GOLNexus

Nexus Repository: Releases

Packages

Group	com.wakaleo.gameoflife
Artifact	gameoflife
Version	1.0
Packaging	war

Artifacts

Add Package

'Add Artifact Path' configured in Step 19

Matching Pom.xml (Code below required for referencing the same settings)

```
506 lines (499 sloc) | 14.6 KB
1  <?xml version="1.0" encoding="UTF-8"?>
2  <project
3      xmlns="http://maven.apache.org/POM/4.0.0"
4      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"
5      <modelVersion>4.0.0</modelVersion>
6      <groupId>com.wakaleo.gameoflife</groupId>
7      <artifactId>gameoflife</artifactId>
8      <version>1.0-SNAPSHOT</version>
9      <packaging>pom</packaging>
10     <name>gameoflife</name>
11     <url>https://github.com/wakaleo/game-of-life</url>
```

Step 18)

Locate Jenkins Workspace (default location below)

```
root@nexuslocal:~/jenkins/workspace/GOLNexus/gameoflife-web/target
root@nexuslocal:~/jenkins/workspace/GOLNexus/gameoflife-web/target# ls
classes      gameoflife.war      generated-test-sources  maven-archiver  surefire          test-classes
gameoflife    generated-sources jacoco.exec        maven-status   surefire-reports
root@nexuslocal:~/jenkins/workspace/GOLNexus/gameoflife-web/target# pwd
/root/.jenkins/workspace/GOLNexus/gameoflife-web/target
```

Jenkins.instance.getJob('<job-name>').lastBuild.workspace;

Back to List

Status

Configure

Build History

Load Statistics

Script Console

Build Executor Status

```
1 Jenkins.instance.getJob('GOLNexus').lastBuild.workspace;
```

Result

Result: /root/.jenkins/workspace/GOLNexus

Step 19)

Under 'Maven Artifacts', Add 'File Path' for the Jenkins workspace

Nexus Instance: GOLNexus

Nexus Repository: Releases

Packages:

- Group: com.wakaleo.gameoflife
- Artifact: gameoflife
- Version: 1.0
- Packaging: war

Artifacts:

Maven Artifact

File Path: gameoflife-web\target\gameoflife.war

Step 20)

Build Project after applying the settings above

New Item

People

Build History

Project Relationship

S	W	Name	Last Success	Last Failure	Last Duration
		GOLNexus	1 min 16 sec - #8	5 min 21 sec - #7	44 sec

```
easyb testing
[INFO]
[INFO] --- maven-war-plugin:2.1.1:war (default-war) @ gameoflife-web ---
[INFO] Packaging webapp
[INFO] Assembling webapp [gameoflife-web] in [/root/.jenkins/workspace/GOLNexus/gameoflife-web/target/gameoflife]
[INFO] Processing war project
[INFO] Copying webapp resources [/root/.jenkins/workspace/GOLNexus/gameoflife-web/src/main/webapp]
[INFO] Webapp assembled in [318 msecs]
[INFO] Building war: /root/.jenkins/workspace/GOLNexus/gameoflife-web/target/gameoflife.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] gameoflife ..... SUCCESS [ 7.432 s]
[INFO] gameoflife-build ..... SUCCESS [ 4.538 s]
[INFO] gameoflife-core ..... SUCCESS [ 19.152 s]
[INFO] gameoflife-web ..... SUCCESS [ 8.154 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 39.610 s
[INFO] Finished at: 2017-07-27T20:41:06-05:00
[INFO] Final Memory: 29M/229M
[INFO] -----
Uploading Maven asset with groupId: com.wakaleo.gameoflife artifactId: gameoflife version: 1.1 To repository: releases
Successfully Uploaded Maven Assets
Finished: SUCCESS
```

Artifacts should successfully display with Nexus after a successful Jenkins build

The screenshot shows the Nexus Repository Manager interface. On the left, there's a sidebar with 'Sonatype™' branding and links for Artifact Search, Advanced Search, Views/Repositories (Repositories, Repository Targets, Routing, System Feeds), Security (LDAP Configuration, Privileges, Roles, Users), and Administration (Analytics, Capabilities, Logging, Plugin Console, Scheduled Tasks, Server). The main area is titled 'Repositories' and shows a table of repositories: Apache Snapshots (proxy, maven2, Snapshot, In Service, Path: http://localhost:8081/nexus/content/repositories/apache-snapshots...), Central (proxy, maven2, Release, In Service, Path: http://localhost:8081/nexus/content/repositories/central), Central M1 shadow (virtual, maven1, Release, In Service, Path: http://localhost:8081/nexus/content/shadows/central-m1), Releases (hosted, maven2, Release, In Service, Path: http://localhost:8081/nexus/content/repositories/releases), and Snapshots (hosted, maven2, Snapshot, In Service, Path: http://localhost:8081/nexus/content/repositories/snapshots). Below the table, under 'Releases', is a tree view: Releases > com > wakaleo > gameoflife > 1.0 > gameoflife-1.0.war and > 1.1 > gameoflife-1.1.war. A red oval highlights the 'gameoflife' folder under 'wakaleo'.

This screenshot shows the same Nexus interface as above, but with a different focus. It highlights the 'Administration' section in the sidebar. In the main 'Publishing' area, it shows the status as 'Published.', a message 'Prefix file published successfully.', and the date 'Published on: Thu Jul 27 2017 20:16:48 GMT-0500 (CDT)'. A red oval highlights this publishing information.

Steps above should successfully demonstrate:

- 1) Installing Nexus with the required dependencies
- 2) Configuring Jenkins using the GOL (Game of Life) Project
- 2) Generating an Artifact and uploading to a Nexus Repository