# Maven私有仓库

在2017/12/06 19:29上被[亮 刘](http://er.msqltec.com:8080/xwiki/bin/view/XWiki/liuliang)修改

### 前言：

　　公司一般都有个自己的私服来管理各种jar包，原因大概有这么3个，分别是：

　　　　1、有的公司不能访问外网，只能通过私服来管理jar包和插件；

　　　　2、公司网速比较慢，通过公司的私服来获取jar包比较快；

　　　　3、内部的的一些jar包放在私服上，比较方便的大家使用。

　　这里简单介绍下搭建Nexus环境，并且在eclipse mvn插件环境下来上传jar包。

### Nexus环境搭建：

　　　说明：这里是Windows环境，Linux类似。

　　1、下载Nexus安装文件：<https://www.sonatype.com/download-oss-sonatype> ，目前是[nexus-2.14.5-02-bundle.tar.gz](https://www.sonatype.com/oss-thank-you-tgz)这个最新版本；

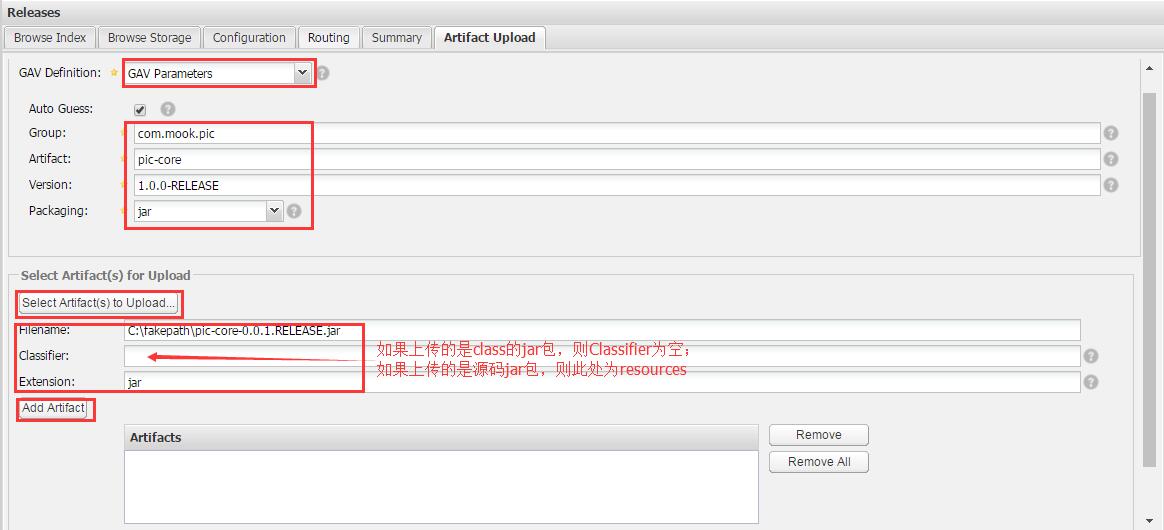
　　2、解压到任意目录，我这里解压到E盘，进入E:\nexus-2.14.5-02\bin\jsw\windows-x86-64，点击install-nexus.bat安装成为windows服务,start-nexus.bat启动服务，这里选择自己计算机对应的文件；

　　3、访问http://localhost:8081/nexus/，这就是私服；

　　4、点击右上角登录，默认帐号admin，密码admin123；

　　5、点击左边Repositories，可以看到一系列仓库类型，我们一般使用3个，分别是Releases，Snapshots和3rd party。这3个仓库风别是装稳定版，开发版和第三方jar包　。

　　6、举例，手动上传一个jar到Releases仓库里面。



　　7、上面是手动方式，另外一种是利用mvn直接发布到私服上面来，需要修改2个地方，第一是在settings文件的servers节点下加入，这是配置私服的密码和仓库，配合pom文件中的仓库地址就构成了完成的访问私服的要素，帐号和密码之所以在settings中设置，是由于settings文件是本地的，而pom.xml文件是公共的，不安全，所以放在settings中：

<server>

<id>nexus-releases</id>

<username>admin</username>

<password>admin123</password>

</server>

<server>

<id>nexus-snapshots</id>

<username>admin</username>

<password>admin123</password>

</server>

　　第二是在pom.xml中增加，下面的id和settings中的id必须要一样。

<distributionManagement>

<repository>

<id>release</id>

<name>Nexus Release Repository</name>

<url>http://localhost:8081/nexus/content/repositories/release/</url>

</repository>

<snapshotRepository>

<id>snapshots</id>

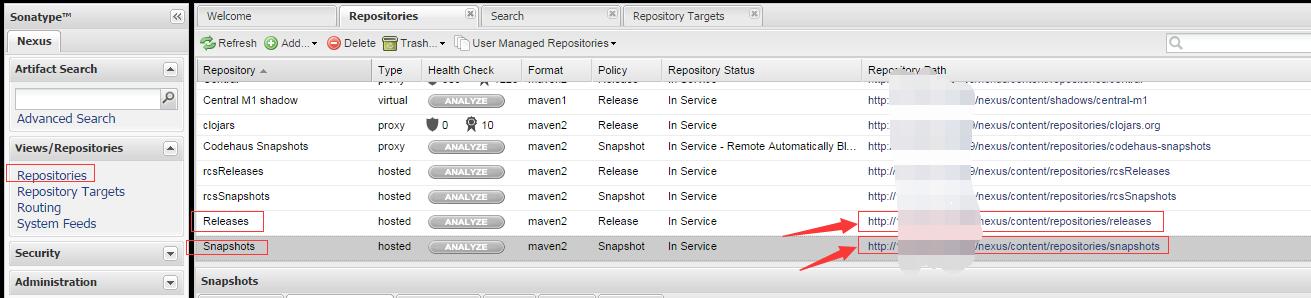
<name>Nexus Snapshot Repository</name>

<url>http://localhost:8081/nexus/content/repositories/snapshots/</url>

</snapshotRepository>

</distributionManagement>

　　上面的私服两个仓库的url从什么地方获取呢？如下图：



　　8、新建一个mvn项目，

　　　　1、运行source:jar在target下生成源码jar包；

　　　　2、运行deploy，class的jar包发布到私服；

　　　　3、运行source:jar deploy，连同源码一起发布到私服。

　　9、完成，在其他项目中可以使用上面的jar包了。

更改MAVEN库为阿里maven库，下载很快

在自己的maven文件中settings.xml中增加才下内容即可

 <profiles>  
      <profile>  
        <id>dev</id>  
        <!-- repositories and pluginRepositories here-->  
        <repositories>  
        <repository>  
        <id>nexus-aliyun</id>  
        <name>Nexus aliyun</name>  
        <url>http://maven.aliyun.com/nexus/content/groups/public</url>  
        <releases>  
        <enabled>true</enabled>  
        </releases>  
        <snapshots>  
        <enabled>true</enabled>  
        </snapshots>  
        </repository>  
        </repositories>  
    </profile>  
  </profiles>  
<activeProfiles>  
  <activeProfile>dev</activeProfile>  
  </activeProfiles>

所有的子系统版本号都增加了-SNAPSHOT

代表使用的是开发版的功能库，如果使用正式版去掉-SNAPSHOT就可以了

公司MAVEN库环境配置

在IDEA开发工具下file-->settings-->maven 设置成自己的settings.xml文件

配置maven中conf下的settings.xml文件

在<servers>标签中增加

<server>  
      <id>Releases</id>  
      <username>admin</username>  
      <password>admin123</password>  
    </server>  
     <server>  
        <id>snapshots</id>  
        <username>admin</username>  
        <password>admin123</password>  
    </server>

在<mirrors>标签中增加  注意url就行，其它无所谓

<mirror>  
        <id>nexus-aliyun</id>  
        <mirrorOf>\*</mirrorOf>  
        <name>Nexus aliyun</name>  
        <url>http://192.168.1.10:8081/nexus/content/groups/public/</url>  
    </mirror>

为了能下载snapshots库中的包在<profiles>标签下增加以下内容

<profile>  
        <id>nexus</id>  
        <!--Enable snapshots for the built in central repo to direct -->  
        <activation>  
            <activeByDefault>true</activeByDefault>  
        </activation>  
        <!--all requests to nexus via the mirror -->  
        <repositories>  
            <repository>  
                <id>maven-releases</id>  
                    <url>http://192.168.1.10:8081/nexus/content/repositories/Releases/</url>  
                    <releases><enabled>true</enabled></releases>  
                    <snapshots><enabled>false</enabled></snapshots>  
                </repository>  
                <repository>  
                    <id>maven-snapshots</id>  
                    <url>http://192.168.1.10:8081/nexus/content/repositories/snapshots/</url>  
                    <releases><enabled>false</enabled></releases>  
                    <snapshots><enabled>true</enabled></snapshots>  
            </repository>  
        </repositories>  
    </profile>

打包上传到开发库

-----------------------------------------------------------------------------------------------------

如果 没有下载maven也可以创建一个settings.xml文件

<?xml version="1.0" encoding="UTF-8"?>  
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"  
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
          xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-1.0.0.xsd">  
  <pluginGroups>  
  </pluginGroups>  
  <proxies>  
  </proxies>  
  <servers>  
    <server>  
      <id>Releases</id>  
      <username>admin</username>  
      <password>admin123</password>  
    </server>  
     <server>  
        <id>snapshots</id>  
        <username>admin</username>  
        <password>admin123</password>  
    </server>  
  </servers>  
  <mirrors>  
      <mirror>  
        <id>nexus-aliyun</id>  
        <mirrorOf>\*</mirrorOf>  
        <name>Nexus aliyun</name>  
        <url>http://192.168.1.10:8081/nexus/content/groups/public/</url>  
    </mirror>  
  </mirrors>  
  <profiles>  
  </profiles>  
</settings>

Linux 安装

Docker search nexus

Docker pull \*\*

docker run -id --privileged=true --name=nexus3 --restart=always -p 8081:8081 -v /kichun/nexus3/nexus-data:/var/nexus-data

<profiles>

<profile>

<id>dev</id>

<repositories>

<repository>

<id>local-nexus</id>

<url>http://192.168.3.128:8081/repository/maven-public/</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</repository>

</repositories>

</profile>

</profiles>

<activeProfiles>

<activeProfile>dev</activeProfile>

</activeProfiles>



image.png

配置maven settings文件的服务器用户名密码  
注意：id为私服中releases和snapshots仓库名，必须一致

<!-- servers

| This is a list of authentication profiles, keyed by the server-id used within the system.

| Authentication profiles can be used whenever maven must make a connection to a remote server.

|-->

<servers>

<server>

<id>maven-releases</id>

<username>admin</username>

<password>admin123</password>

</server>

<server>

<id>maven-snapshots</id>

<username>admin</username>

<password>admin123</password>

</server>

</servers>

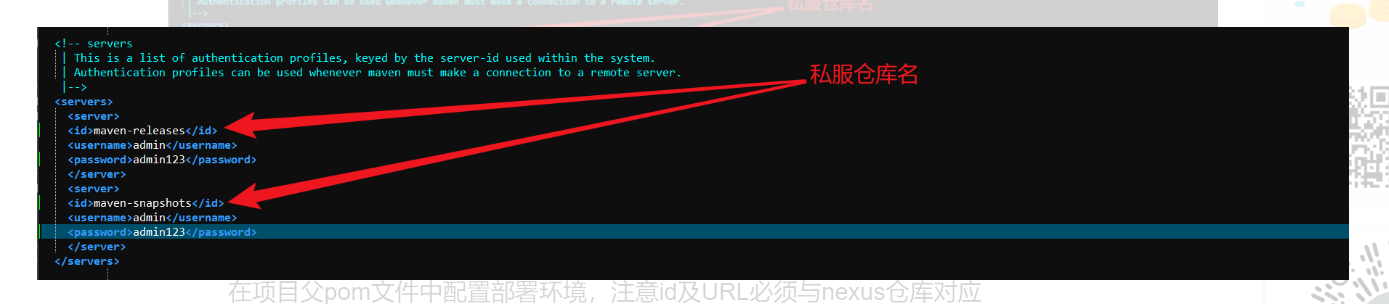


image.png

在项目父pom文件中配置部署环境，注意id及URL必须与nexus仓库对应

<!--私服仓库-->

<distributionManagement>

<repository>

<id>maven-releases</id>

<name>Nexus Release Repository</name>

<url>http://192.168.3.128:8081/repository/maven-releases/</url>

</repository>

<snapshotRepository>

<id>maven-snapshots</id>

<name>Nexus Snapshot Repository</name>

<url>http://192.168.3.128:8081/repository/maven-snapshots/</url>

</snapshotRepository>

</distributionManagement>

