<u>如何发布Jar包到Maven中央仓库</u>

本博客曾经介绍了<u>《如何手动添加依赖的jar文件到本地Maven仓库》</u>

这里的方法非常的简单,而且局限性很大:只能提供给本人开发使用,无法共享给其他需要的人。本文将介绍如何把自己开发

出来的Java包发布到Maven中央仓库(<u>http://search.maven.org/</u>

),这样任何人都可以搜索到这个包并使用它。如果你现在还不了解Maven是啥东西,请你先找本 Maven书看看。

将自己的Java包发布到Maven中央仓库需要经过许多繁琐的步骤,本文将详细地介绍这些步骤。相信大家在看完本文之后也会发布jar包到Maven中央仓库。

创建工单

Soark技术博

在发布Java包到Maven中央仓库首先需要在

https://issues.sonatype.org/secure/Dashboard.jspa

网站创建一个工单(Issues),第一次使用这个网站的时候需要注册自己的帐号(这个帐号和密码需要记住,后面会用到),之后创建自己的Issue,点击导航最上面的Create按钮,然后会弹出下面的对话框,将Project和Issue Type设置为下图的内容:



Create Issue		Configure Fiel	lds 🕶
Project [*] Issue Type [*]	Community Support - Open Community Support - Open New Project Some issue types are unavailable due to incompatible field configuration and/or world	Row associations	^
Summary*			1
Description	Style • B I <u>U</u> <u>A</u> • ^a A • ² A •	• +• *	*
		_	-
	Create an	other Create (Cancel

如果想及时了

解Spark、Hadoop或者Hbase相关的文章,欢迎关注微信公共帐号:iteblog_hadoop

然后根据你Java包的功能分别写上Summary、Description、Group Id、SCM url以及Project URL等必要信息,可以参见我之前创建的Issue:

https://issues.sonatype.org/browse/OSSRH-24874

。创建完之后需要等待Sonatype的工作人员审核处理,审核时间还是很快的,我的审核差不多等 待了两小时。当Issue的Status变为RESOLVED后,就可以进行下一步操作了。

如果你的Group

Id填写的是自己的网站(我的就是这种情况),Sonatype的工作人员会询问你那个Group Id是不是你的域名,你只需要在上面回答是就行,然后就会通过审核。

使用gpg生成密钥对

上面创建的issuce经过审核之后,我们可以使用gpg生成密钥对,这里分两种情况:

1、如果使用的是Windows,可以到<u>https://www.gpg4win.org/download.html</u>

下载gpg4win,推荐使用 Gpg4win-Vanilla 2.3.3版本,因为它仅包括 GnuPG,这个工具才是我们所需要的;

2、如果使用的是Linux,可以通过yum install gpg命令安装gpg。

之后可以通过gpg --version命令查看是否安装成功,如果出现版本等信息说明安装成功了,如下:

[iteblog@www.iteblog.com ~] \$ gpg --version gpg (GnuPG) 2.0.14 libgcrypt 1.4.5 Copyright (C) 2009 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html> This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

Home: ~/.gnupg
Supported algorithms:
Pubkey: RSA, ELG, DSA
Cipher: 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH, CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2

安装完gpg之后,下面一次按照如下的命令设置好你的名字,邮箱,其他步骤可以使用默认值。 不过输入Passphrase的值需要记住,这个相当于密钥的密码,发布过程中进行签名操作的时候会 用到:

\$ gpg --gen-key gpg (GnuPG) 1.4.19; Copyright (C) 2015 Free Software Foundation, Inc. This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:

(1) RSA and RSA (default)

- (2) DSA and Elgamal
- (3) DSA (sign only)

(4) RSA (sign only)

Your selection?

RSA keys may be between 1024 and 4096 bits long.

What keysize do you want? (2048)

Requested keysize is 2048 bits

Please specify how long the key should be valid.

0 = key does not expire <n> = key expires in n days <n>w = key expires in n weeks <n>m = key expires in n months <n>y = key expires in n years Key is valid for? (0) Key does not expire at all Is this correct? (y/N) Y

You need a user ID to identify your key; the software constructs the user ID from the Real Name, Comment and Email Address in this form: "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: iteblog Email address: wyphao.2007@163.com Comment: flink-elasticsearch-connector You selected this USER-ID: "iteblog (flink-elasticsearch-connector) <wyphao.2007@163.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O You need a Passphrase to protect your secret key.

We need to generate a lot of random bytes. It is a good idea to perform some other action (type on the keyboard, move the mouse, utilize the disks) during the prime generation; this gives the random number generator a better chance to gain enough entropy. +++++

.+++++

gpg: /c/Users/iteblog/.gnupg/trustdb.gpg: trustdb created gpg: key B15C5AA3 marked as ultimately trusted public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
pub 2048R/B15C5AA3 2016-09-19
 Key fingerprint = DB61 9873 924C 020E 20E7 E461 0170 C912 B15C 5AA3
vid

uid iteblog (flink-elasticsearch-connector) <wyphao.2007@163.com> sub 2048R/31A906E1 2016-09-19

到这里我们就设置好密钥对了。上面代码中导数第四行的B15C5AA3需要记住,其相当于我们生成的key,后面会用到。

设置Maven配置

tp://www.iteblog

技术博

为了发布更简便,我们可以工程的pom.xml文件里面加入以下的配置:

<parent>

<groupId>org.sonatype.oss</groupId> <artifactId>oss-parent</artifactId> <version>7</version>

</parent>

并增加Licenses、SCM、Developers信息:

<licenses>

<license>

<name>The Apache Software License, Version 2.0</name> <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url> <distribution>repo</distribution> </license>

</licenses>

<scm>

```
<url>https://github.com/397090770/flink-elasticsearch2-connector</url>
<connection>git@github.com:397090770/flink-elasticsearch2-connector.git</connection>
<developerConnection>https://www.iteblog.com</developerConnection>
</scm>
```

```
<developers>
<developer>
<name>iteblog</name>
<email>wyphao.2007@163.com</email>
<url>https://www.iteblog.com</url>
</developer>
</developers>
```

最后修改Maven的setting.xml配置文件,设置如下:

<servers> <server> <id>sonatype-nexus-snapshots</id>



<username>Sonatype网站的账号</username>
 <password>Sonatype网站的密码</password></server>
 <id>sonatype-nexus-staging</id>
 <username>Sonatype网站的账号</username>
 <password>Sonatype网站的密码</password>
 </server>
 </server>

上面的username和password就是你在步骤一注册的帐号和密码。

部署和发布Jar包

配置好pom.xml文件和Maven的setting.xml文件之后,我们就可以部署Jar包了,运行如下命令:

\$ mvn clean deploy -P sonatype-oss-

thin clean deploy i sondype oss
release -Darguments="gpg.passphrase=设置gpg设置密钥时候输入的Passphrase"
[INFO] Scanning for projects
[INFO]
[INFO]
[INFO] Building flink-elasticsearch2-connector 1.0.1
[INFO]
[INFO]
[INFO] maven-clean-plugin:2.5:clean (default-clean) @ flink-elasticsearch2-connector
[INFO] Deleting D:₩work₩flink-elasticsearch2-connector₩target
[INFO]
[INFO] maven-enforcer-plugin:1.0:enforce (enforce-maven) @ flink-
elasticsearch2-connector
[INFO]
[INFO] maven-source-plugin:2.1.2:jar-no-fork (attach-sources) @ flink-
elasticsearch2-connector
[INFO]
[INFO] maven-javadoc-plugin:2.7:jar (attach-javadocs) @ flink-elasticsearch2-connector
[INFO] Not executing Javadoc as the project is not a Java classpath-capable package
[INFO]
[INFO] maven-gpg-plugin:1.1:sign (sign-artifacts) @ flink-elasticsearch2-connector
GPG Passphrase: ****************
*[INFO]
[INFO] maven-install-plugin:2.4:install (default-install) @ flink-elasticsearch2-connector

[INFO] --- maven-install-plugin:2.4:install (default-install) @ flink-elasticsearch2-connector ---[INFO] Installing D:₩work₩flink-elasticsearch2-connector₩pom.xml to D:₩q₩repos₩com ₩iteblog₩flink-elasticsearch2-connector₩1.0.1₩flink-elasticsearch2-connector-1.0.1.pom



[INFO] Installing D:₩work₩flink-elasticsearch2-connector₩target₩flink-elasticsearch2-conn ector-1.0.1.pom.asc to D:₩q₩repos₩com₩iteblog₩flink-

elasticsearch2-connector₩1.0.1₩flink-elasticsearch2-connector-1.0.1.pom.asc [INFO]

[INFO] --- maven-deploy-plugin:2.7:deploy (default-deploy) @ flink-elasticsearch2-connector ---Uploading: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flinkelasticsearch2-connector/1.0.1/flink-elasticsearch2-connector-1.0.1.pom

Uploaded: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flinkelasticsearch2-connector/1.0.1/flink-elasticsearch2-connector-1.0.1.pom (4 KB at 0.4 KB/sec) Downloading: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flinkelasticsearch2-connector/maven-metadata.xml

Uploading: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flink-elasticsearch2-connector/maven-metadata.xml

Uploaded: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flinkelasticsearch2-connector/maven-metadata.xml (321 B at 0.0 KB/sec)

Uploading: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flink-elasticsearch2-connector/1.0.1/flink-elasticsearch2-connector-1.0.1.pom.asc

Uploaded: https://oss.sonatype.org/service/local/staging/deploy/maven2/com/iteblog/flink-ela sticsearch2-connector/1.0.1/flink-elasticsearch2-connector-1.0.1.pom.asc (473 B at 0.1 KB/sec)

[INFO] -----

[INFO] BUILD SUCCESS [INFO] -------[INFO] Total time: 34.272s

[INFO] Finished at: Mon Sep 19 14:15:21 CST 2016

[INFO] Final Memory: 21M/224M

[INFO] ------

运行上面的命令之后,我们再上传到第三方的key验证库,如下:

\$ gpg --list-keys
/c/Users/iteblog/.gnupg/pubring.gpg

pub 2048R/B15C5AA3 2016-09-19

uid iteblog (flink-elasticsearch-connector) <wyphao.2007@163.com> sub 2048R/31A906E1 2016-09-19

iteblog@iteblog MINGW64 /d/work/flink-elasticsearch2-connector (master) \$ gpg --keyserver hkp://keyserver.ubuntu.com:11371 --send-keys B15C5AA3 gpg: sending key B15C5AA3 to hkp server keyserver.ubuntu.com

之后我们的Java包将会发布到sonat

ype的构件仓库中,可以到<u>https://oss.sonatype.org/#stagingRepositories</u>

查看发布好的Jar。进入之后会看到中间一个Table窗口,将滑动条移到最后,找到我们刚刚发布的Jar包,然后依次点击上方的Close->Confirm,这将会弹出类似于下面的对话框,在其中输入我们Jar包的描述信息,这个信息将会在Maven搜索结果当作简介介绍我们Jar包的,所以建议输的详细点。

	ou are abou	ut to close 1 r	epository.	
	no may tan	e sonie ane i	o complete.	
Descrip	ion			

如果想及时了

解Spark、Hadoop或者Hbase相关的文章,欢迎关注微信公共帐号:iteblog_hadoop

当状态变成closed后,执行Release->Confirm,同样会弹出一个类似于上面的对话框,我们还是 输入那些介绍信息即可,当这步执行完之后,构件将会自动删除,并经过几小时后便可以在Mave n中央仓库搜索到(<u>http://search.maven.org</u>)。如下:

The Central Repository		/	SEARCH	ADVANCED SEARCH	BROWSE QUICK STATS
	iteblog			SEARCH	
	New: About Central	Advanced Search AP	I Guide Help		
Search Results <1 > displaying 1 to 1 of				< 1 > displaying 1 to 1 of 1	
GroupId Artifa	ictld	Latest Version	Updated	Download	
com.iteblog flink-el	lasticsearch2-connector	1.0.2 all (2)	20-Sep-2016	pom jar javadoc.jar sou	rces.jar

如果想及时了

解Spark、Hadoop或者Hbase相关的文章,欢迎关注微信公共帐号:iteblog_hadoop

<u>mvnrepository</u>网站的显示:

Home » com.iteblog » flink-elasticsearch2-connector » 1.0.2



Flink Elasticsearch2 Connector » 1.0.2

ElasticSearchOutputFormat for Flink DataSet, Through the ElasticSearchOutputFormat, we can save Flink DataSet to elasticsearch.

License	Apache 2.0
HomePage	https://www.iteblog.com
Date	(Sep 20, 2016)
Files	Download (JAR) (5 KB)
Repository	Central

Maven	radle SBT Ivy Grape Leiningen Buildr	
h<br <depen <g< th=""><th>/mvnrepository.com/artifact/com.iteblog/flink-elasticsearch2-connector> .>com.iteblog</th><th></th></g<></depen 	/mvnrepository.com/artifact/com.iteblog/flink-elasticsearch2-connector> .>com.iteblog	
	tId>flink-elasticsearch2-connector >1.0.2	
<th>></th> <th></th>	>	

Include comment with link to declaration

如果想及时了

解Spark、Hadoop或者Hbase相关的文章,欢迎关注微信公共帐号:iteblog_hadoop

升级Jar包

我们开发的Jar包在后面可能需要更新,比如从1.0.1版本升级到1.0.2版本。升级Jar包比初次 发布的步骤简单的多,我们只需要更新项目工程代码,并修改pom.xml文件里面的版本号,最后 重新执行上面的部署和发布Jar包步骤即可。

> 本博客文章除特别声明,全部都是原创! 原创文章版权归过往记忆大数据(<u>过往记忆</u>)所有,未经许可不得转载。 本文链接:【】()