原连接; https://blog.csdn.net/u014229652/article/details/84867053#4\_123

文章目录

一、准备：

二、直接启动jenkins：

1、下载 Jenkins war包

2、运行

三、通过yum安装jenkins

四、通过nginx代理，用域名访问jenkins

五、使用jenkins启动springboot项目

1、浏览器访问jenkins

2、创建账户

3、设置root url

4、下载基础插件

5、下载其他插件

1、maven插件

2、下载publish over ssh插件

3、下载 ssh插件

4、Git Parameter

6、配置

1、JDK配置

2、maven配置

3、git配置

4、配置服务器

7、部署项目

①、首页点击新建：输入项目名称

②、丢弃旧的构建：

③、参数化构建过程

④、源码管理

⑥、构建环境

⑦、Build

⑧、执行shell

⑨、Send files or execute commands over SSH

⑩、sleep

⑪、创建start.sh脚本

一、准备：

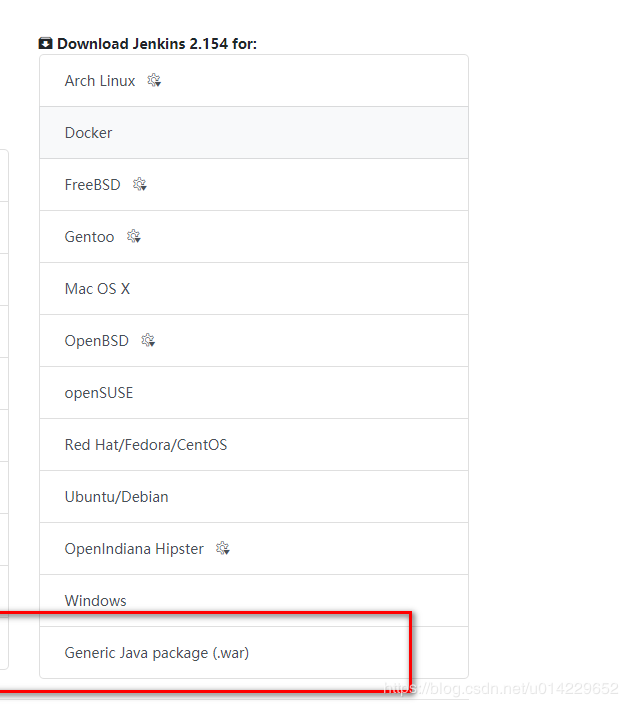
一台物理服务器，ip地址:192.168.1.1

服务器上提前配置好java环境

二、直接启动jenkins：

1、下载 Jenkins war包

官方下载地址



cd /home/download/jenkins

wget http://ftp-nyc.osuosl.org/pub/jenkins/war/2.154/jenkins.war

cp jenkins.war /usr/local/jenkins/

2、运行

在8080端口上运行jenkins

cd /usr/local/jenkins

nohup java -jar jenkins.war --httpPort=8080 >./nohup.log 2>&1 &

tail -1000f nohup.log

浏览器访问 http://localhost:8080.

三、通过yum安装jenkins

去找最新的yum源

https://pkg.jenkins.io/redhat-stable/

命令行：：

wget /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins-2.150.1-1.1.noarch.rpm

rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

yum install jenkins

# 修改端口

vi /etc/sysconfig/jenkins

# 修改JENKINS\_PORT="8080"，默认为“8080”

# 配置java环境

vi /etc/init.d/jenkins

# candidates 里面加入自己的jdk的java这个执行文件的文件目录

jenkins相关目录释义：

（1）/usr/lib/jenkins/：jenkins安装目录，war包会放在这里。

（2）/etc/sysconfig/jenkins：jenkins配置文件，“端口”，“JENKINS\_HOME”等都可以在这里配置。

（3）/var/lib/jenkins/：默认的JENKINS\_HOME。

（4）/var/log/jenkins/jenkins.log：jenkins日志文件。

启动：

service jenkins start

1

四、通过nginx代理，用域名访问jenkins

server {

listen 80;

server\_name jenkins.a.b;

location / {

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

# Fix the "It appears that your reverse proxy set up is broken" error.

proxy\_pass http://127.0.0.1:8080;

proxy\_read\_timeout 90;

#proxy\_redirect http://127.0.0.1:9792 https://jenkins.qinhej.top;

#Required for new HTTP-based CLI

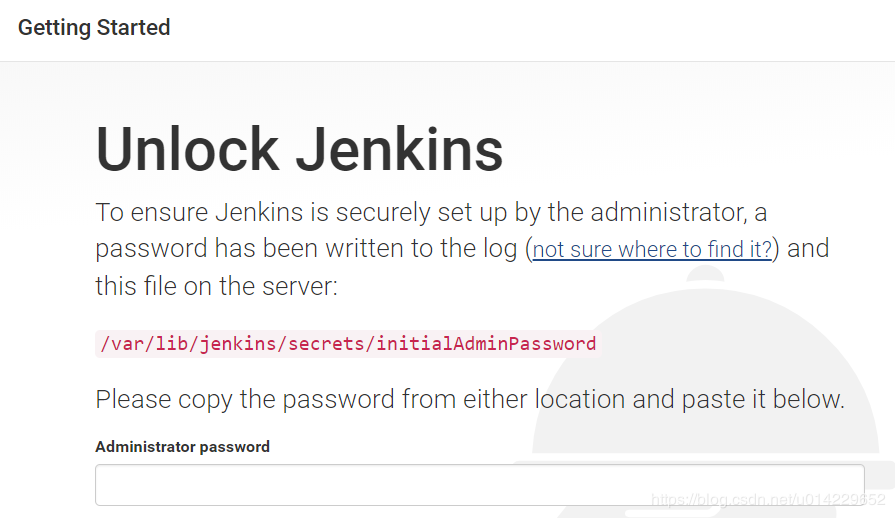
proxy\_http\_version 1.1;

proxy\_request\_buffering off;

}

}

五、使用jenkins启动springboot项目

1. 浏览器访问jenkins
2. 

输入初始密码

cat /var/lib/jenkins/secrets/initialAdminPassword

0094as2ba14b59b7f388zx92cd7e7efd

1

2

2、创建账户

输入完初始密码把安装插件的窗口关掉，直接进入jenkins，设置自己的用户名密码

系统管理—>管理用户 —> create User

输入账号密码保存，退出admin，重新登录自己的账号

3、设置root url

这个可设置可不设置，点击系统管理，如果没设置rooturl的时候可能会出现

Jenkins root URL is empty but is required for the proper operation of many Jenkins features like email notifications, PR status update, and environment variables such as BUILD\_URL.的提示

可以进入系统设置里面设置Jenkins URL

官方解释：

此项是可选的,指定安装Jenkins的HTTP地址,例如http://yourhost.yourdomain/jenkins/. 这个值用来在邮件中生产Jenkins链接.

此项是有必要的,因为Jenkins无法探测到自己的URL地址.

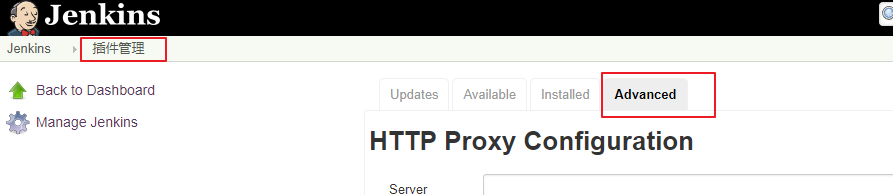
也可以直接忽略

4、下载基础插件

系统管理—>插件管理

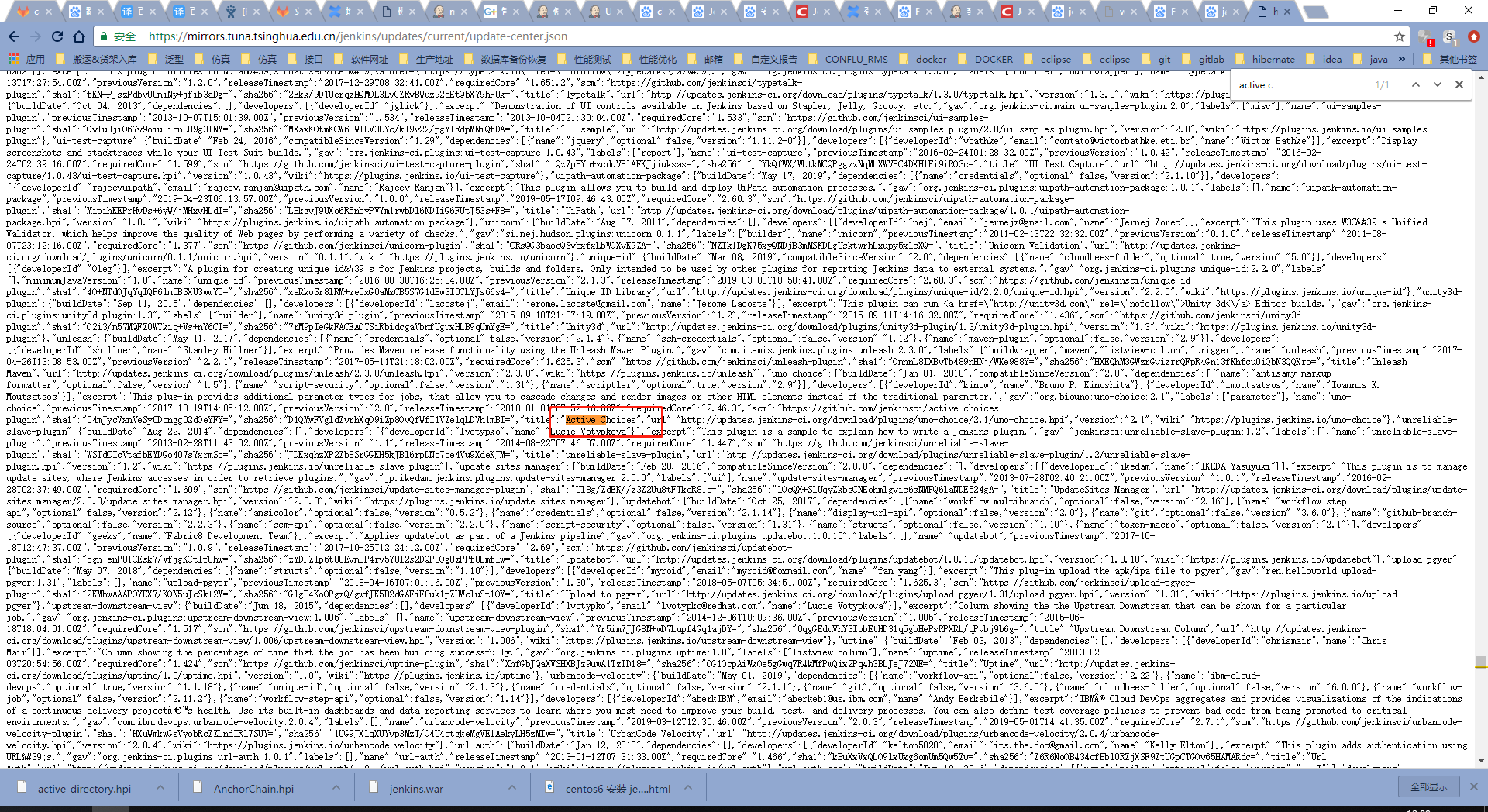
由于之前说过，安装Jenkins后首次访问时由于其他原因【具体未知】会产生离线问题。网上找了个遍还是不能解决，所以只能跳过常用插件安装这步。进入Jenkins后再安装这些插件。

在安装插件前，先修改“更新站点”信息，如下：



站点信息从：https://updates.jenkins.io/update-center.json 改为如下地址【三选一即可】

可以先在网址打开网址，搜索需要的插件；如果能找到就用它



http://mirror.xmission.com/jenkins/updates/update-center.json # 推荐

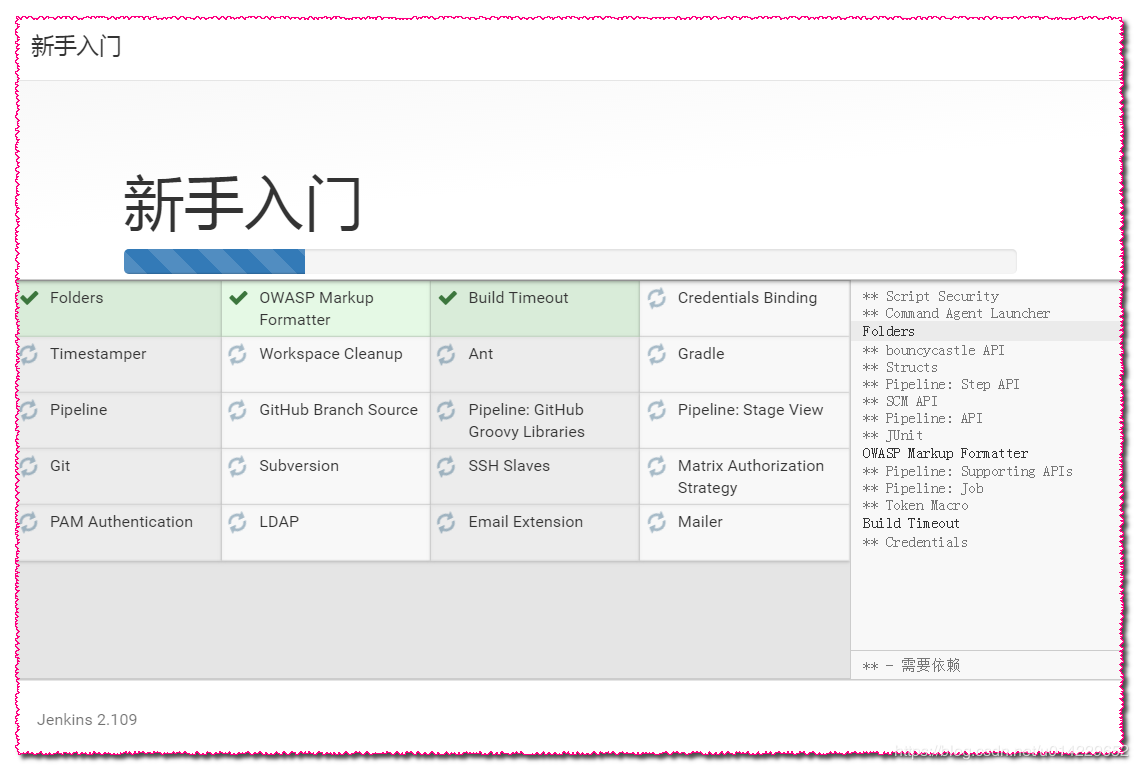
http://mirrors.shu.edu.cn/jenkins/updates/current/update-center.json

<https://mirrors.tuna.tsinghua.edu.cn/jenkins/updates/update-center.json>

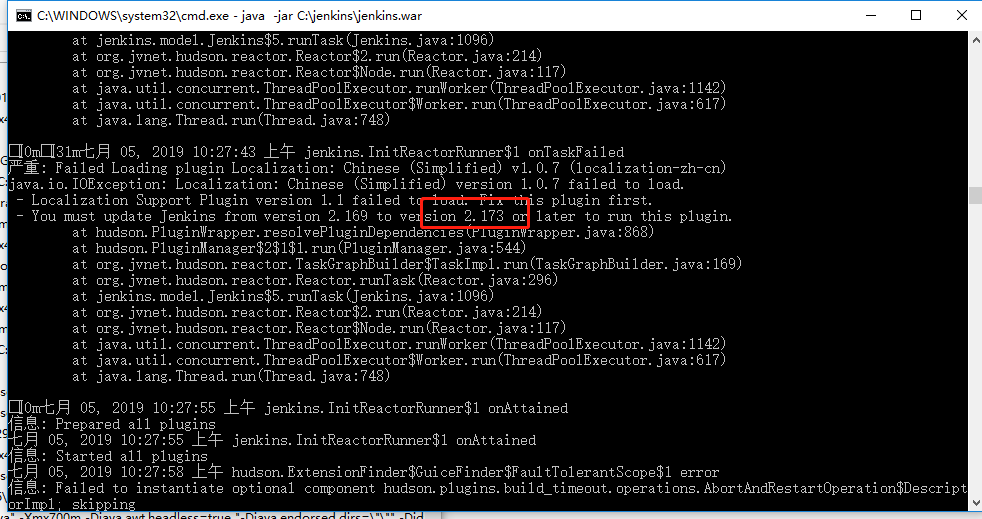
此网站有 activc choice/Extended Choice Parameter插件

<https://mirrors.tuna.tsinghua.edu.cn/jenkins/updates/current/update-center.json>

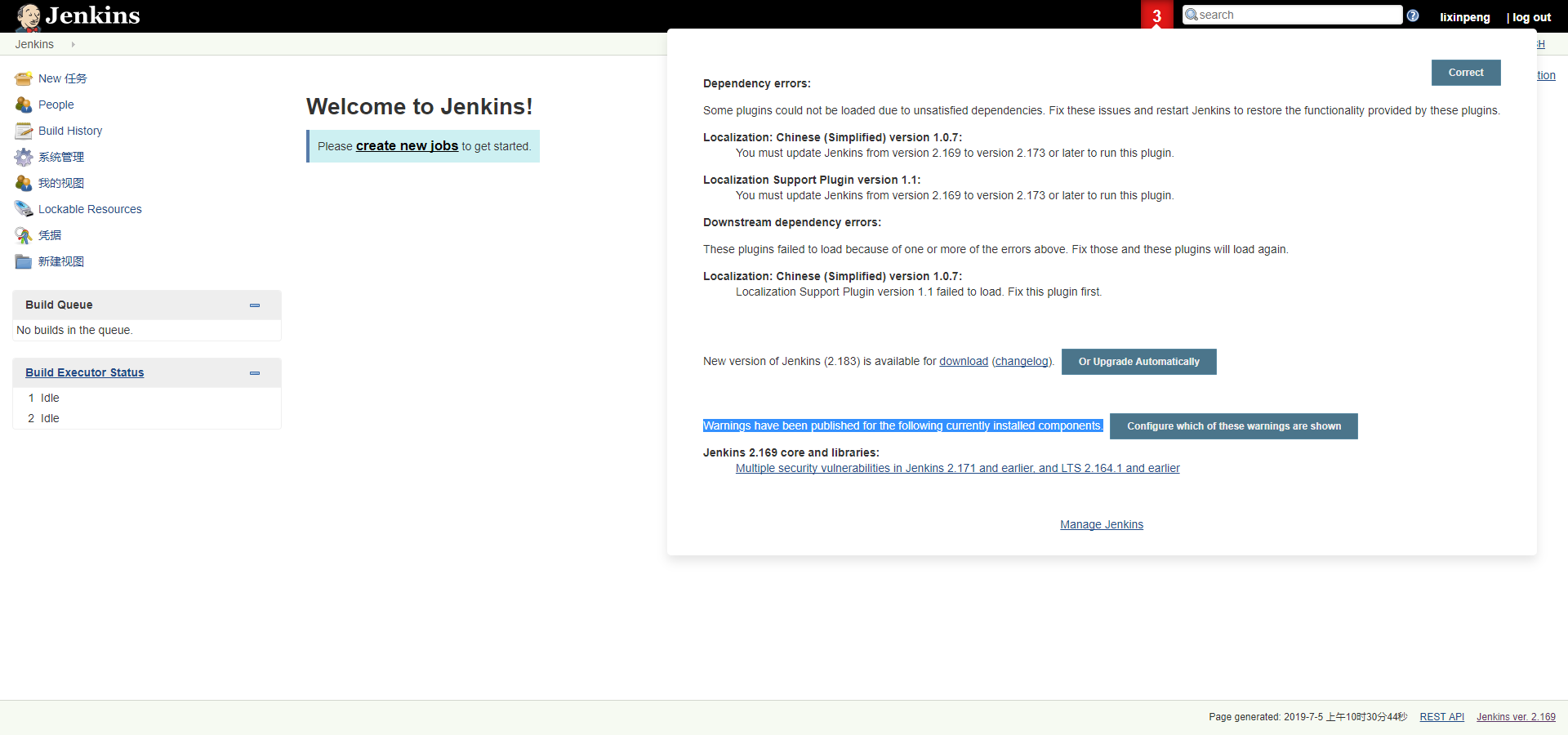
更改完毕后，最好重启Jenkins。



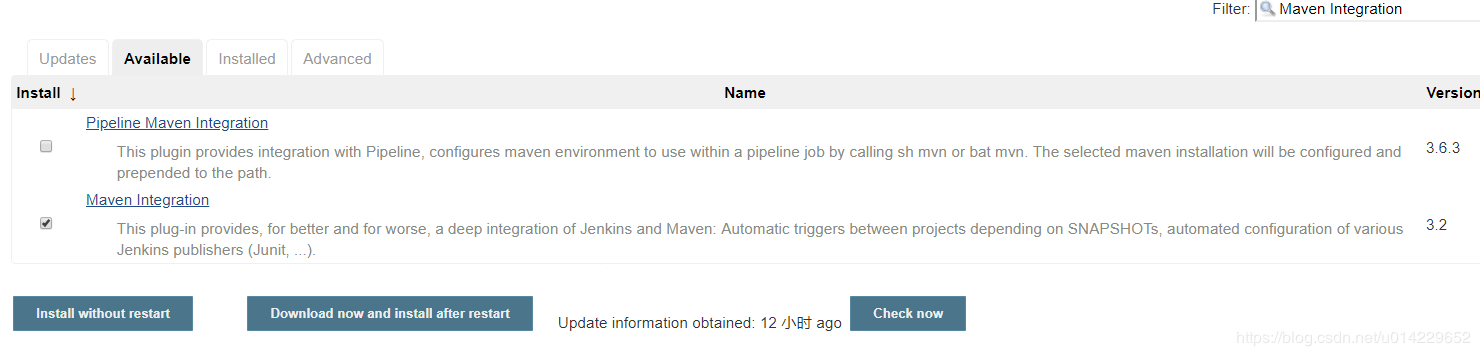
中文本地化插件可能安装失败，后台日志会提示升级jenkis到指定版本



登录会有提示



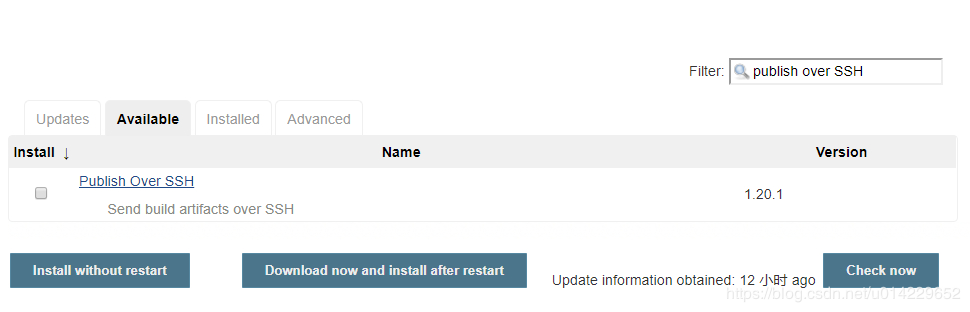
5、下载其他插件

1. maven插件
2. 

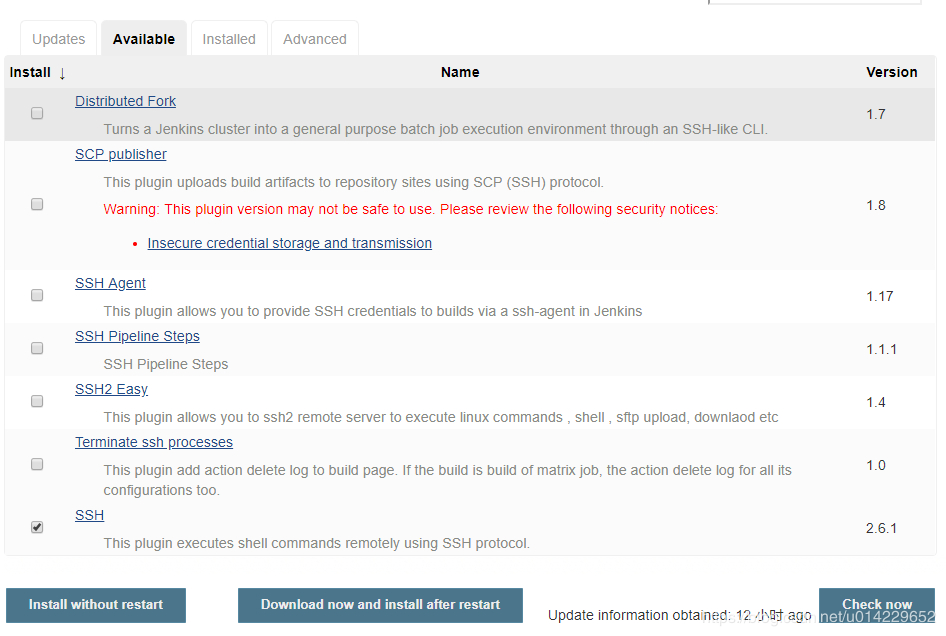
安装之后新建项目的时候才会有maven项目



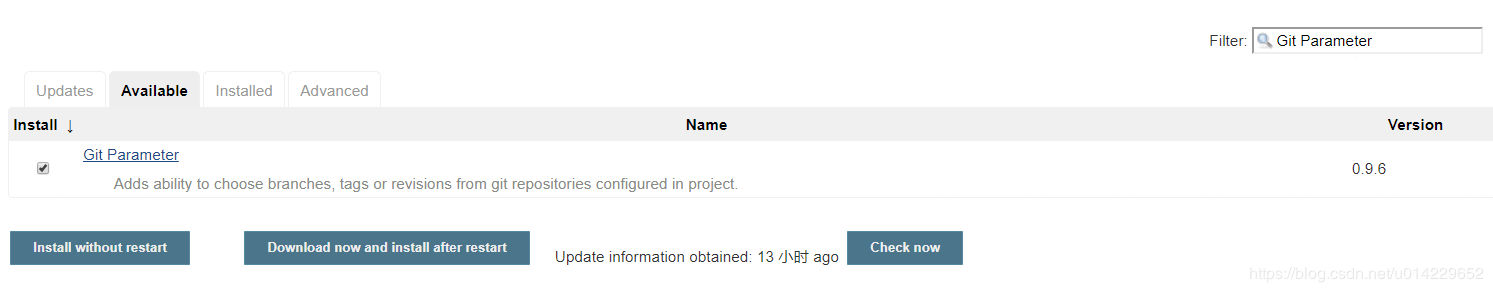
2、下载publish over ssh插件



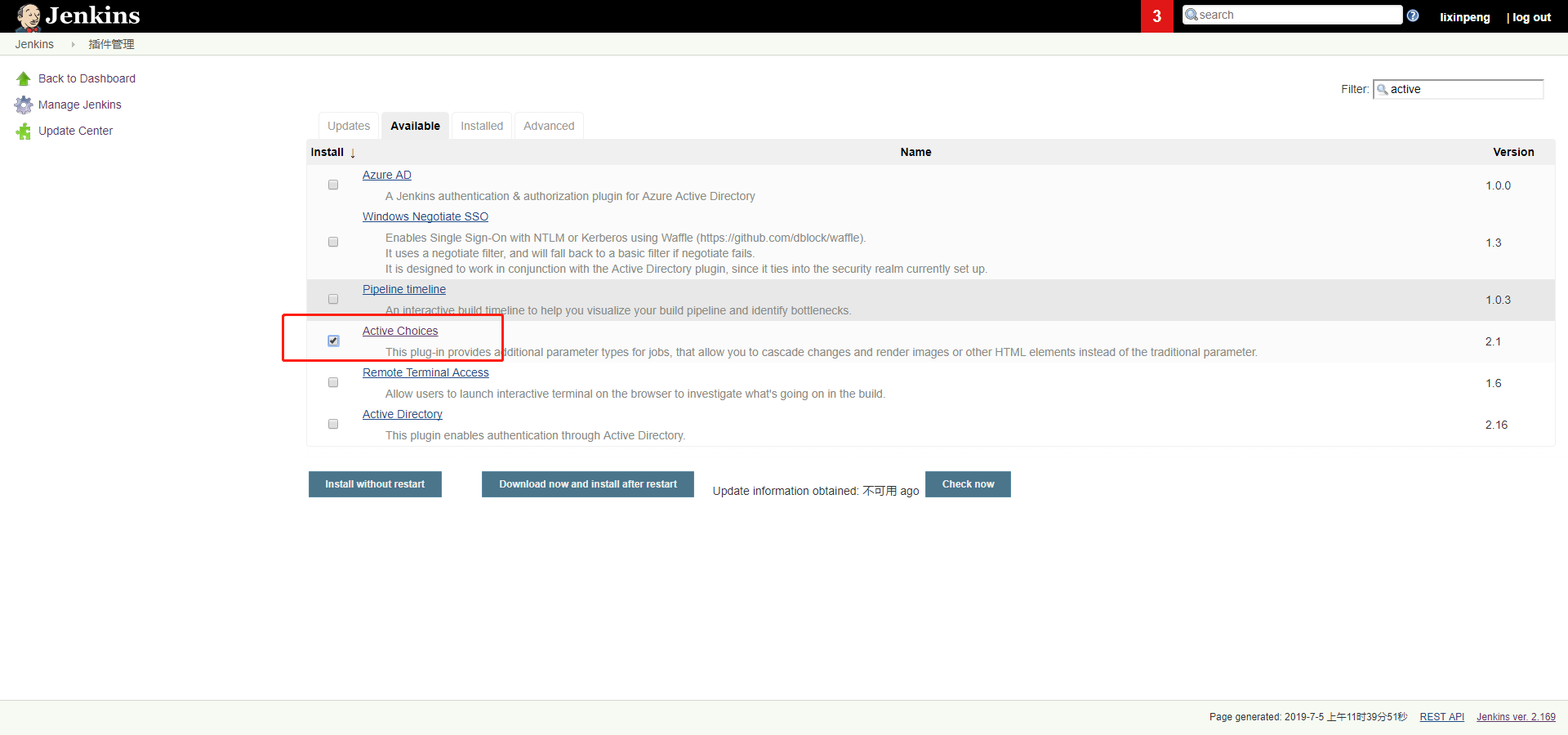
3、下载 ssh插件



4、Git Parameter（根据tag branch选择构建）



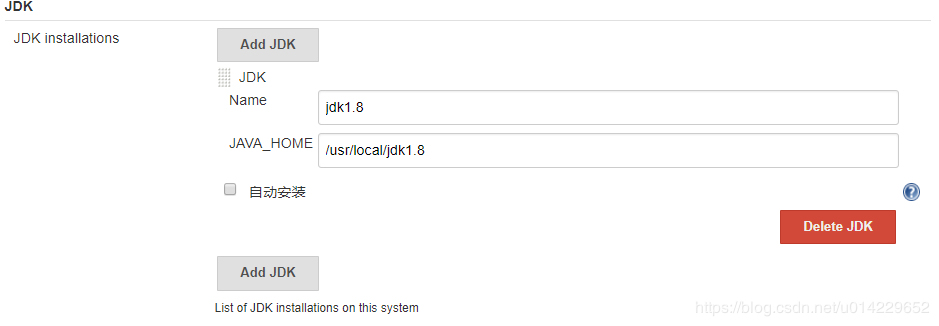
4.1 [Active Choices](https://plugins.jenkins.io/uno-choice)（）在线下载失败可以从网站下载http://updates.jenkins-ci.org/download/plugins/



6、配置

1、JDK配置

系统管理-》全局工具配置-》ADD JDK



2、maven配置

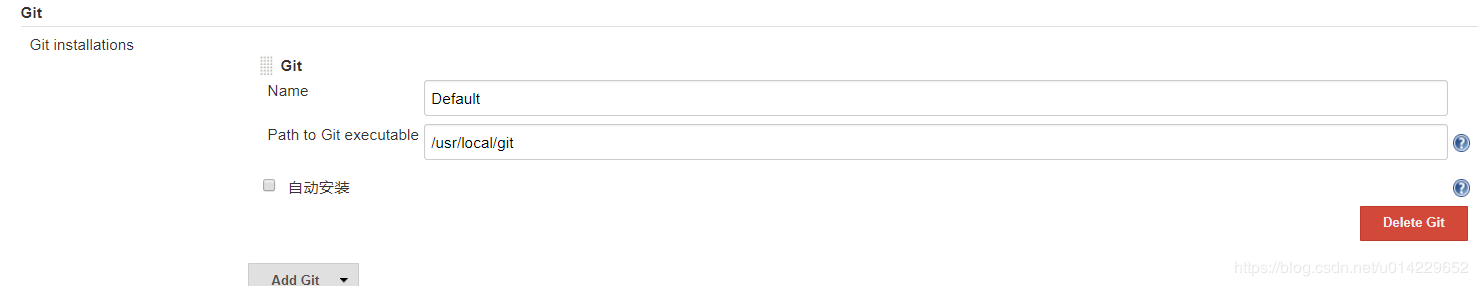
系统管理-》全局工具配置-》ADD MAVEN

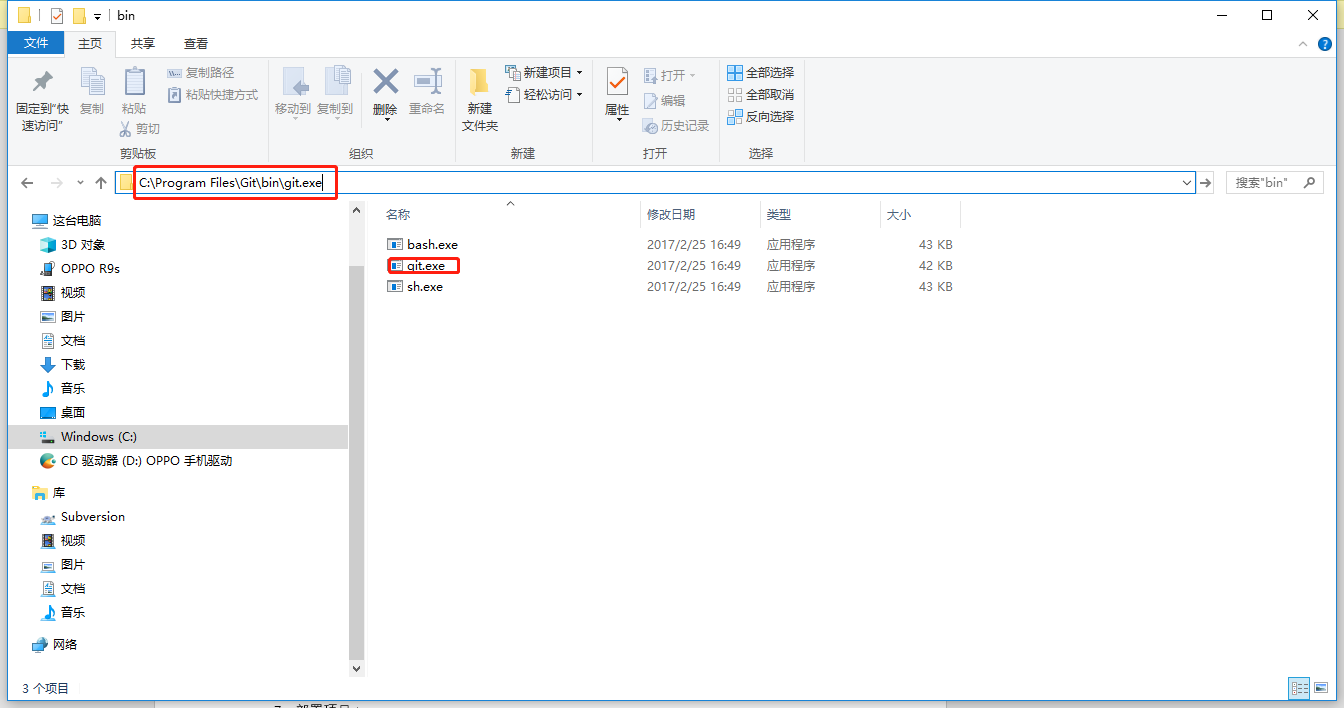


3、git配置

系统管理-》全局工具配置-》git（路径必须到bin层，如下图）

（）

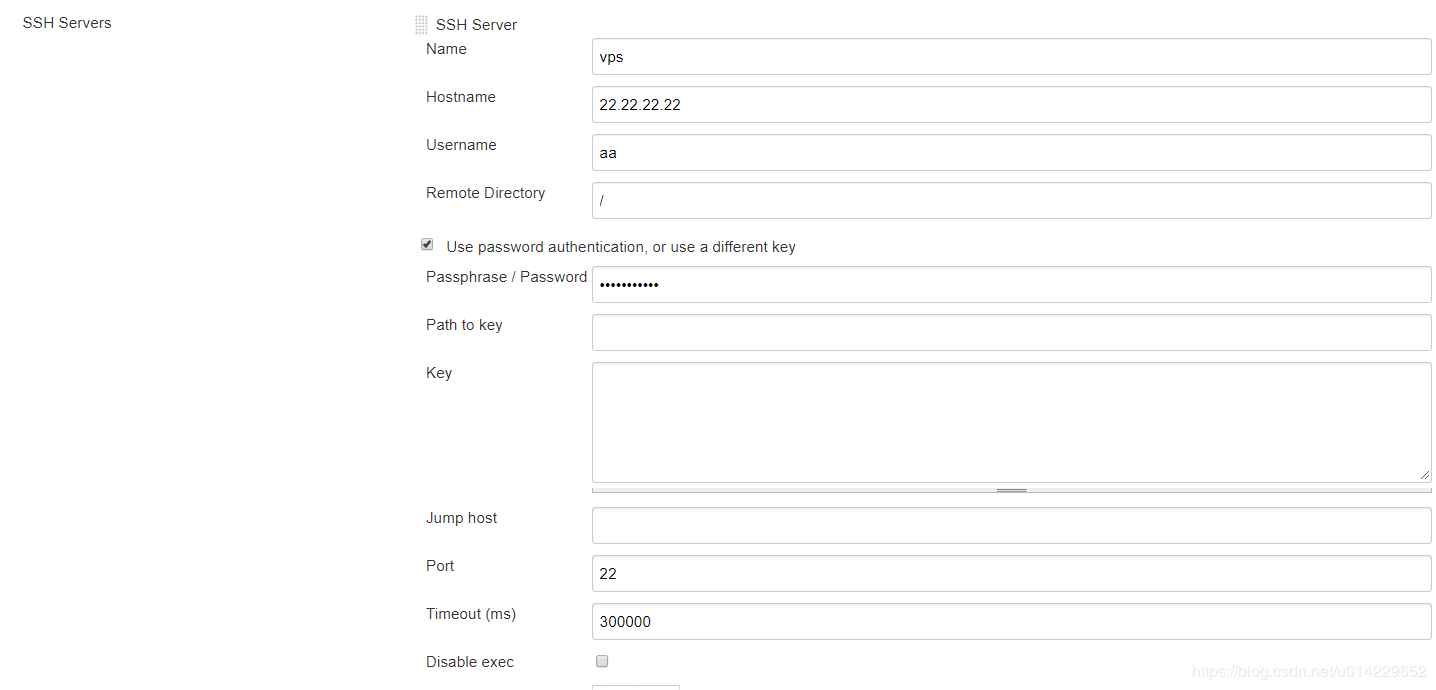




4、配置服务器

系统管理-》系统设置-》Publish over SSH-》SSH Servers-》add

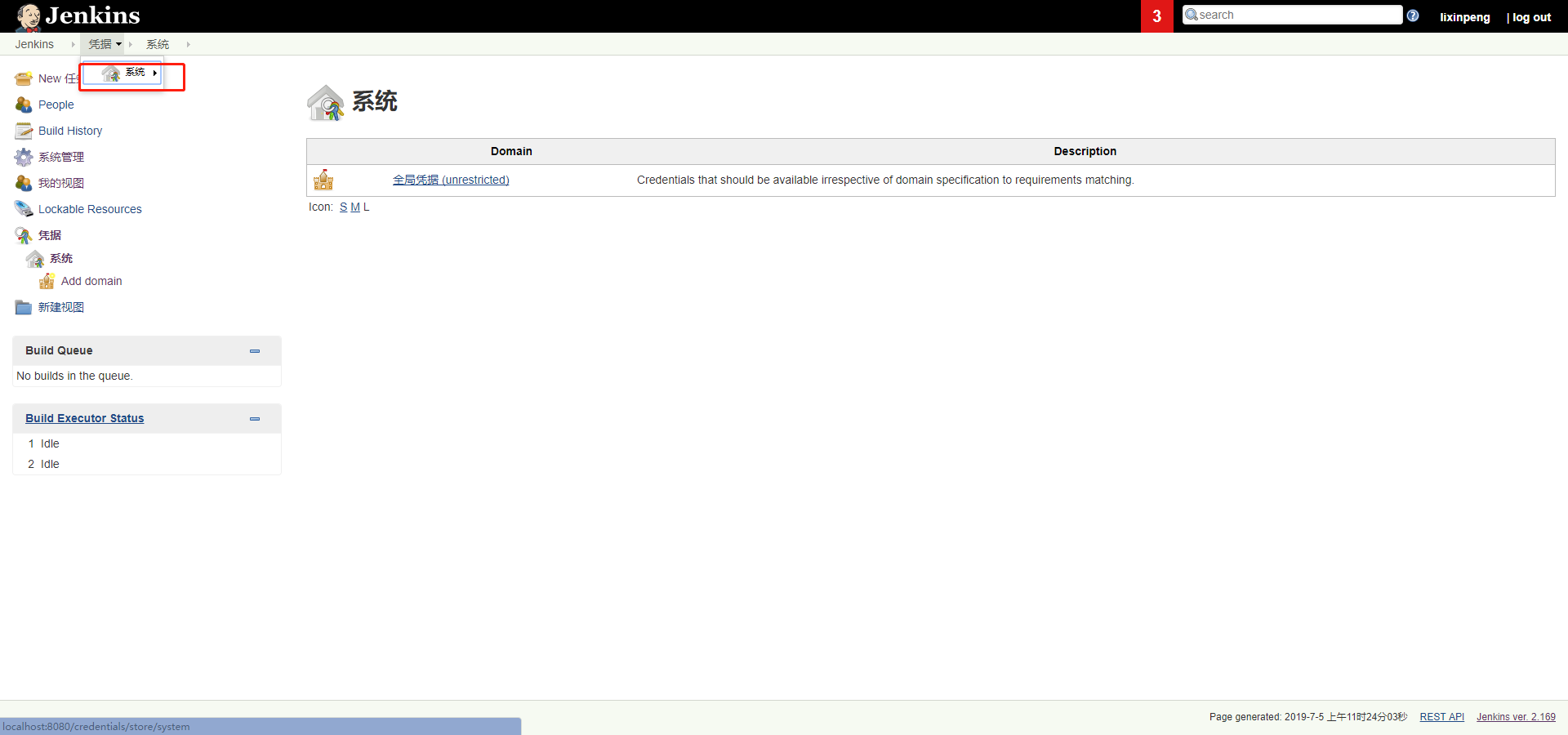
填写ip 用户名 密码



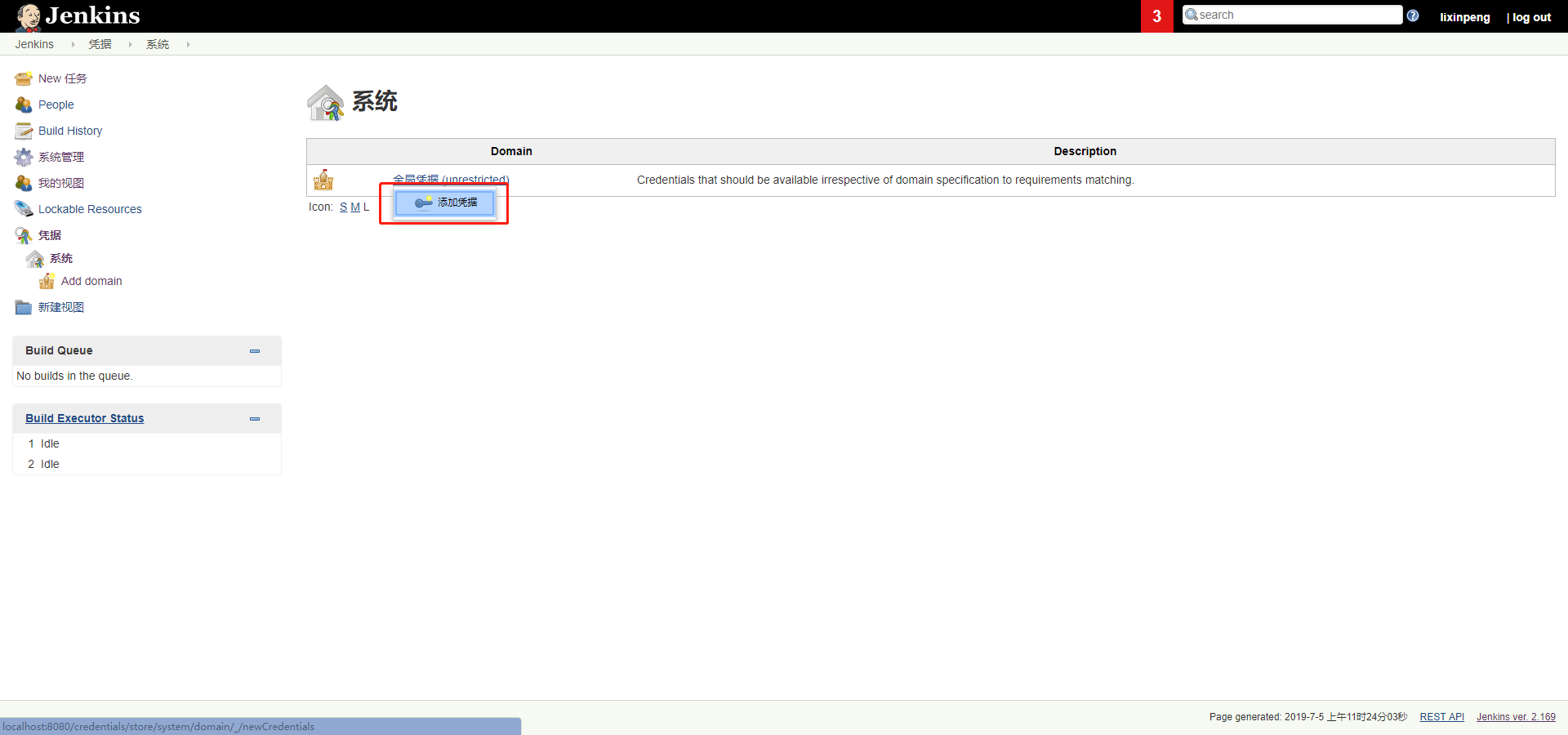
点击 TestConfiguration ，出现success为成功

\*\*\*\*因为管理的是git项目，需要有权限操作远程仓库步骤如下

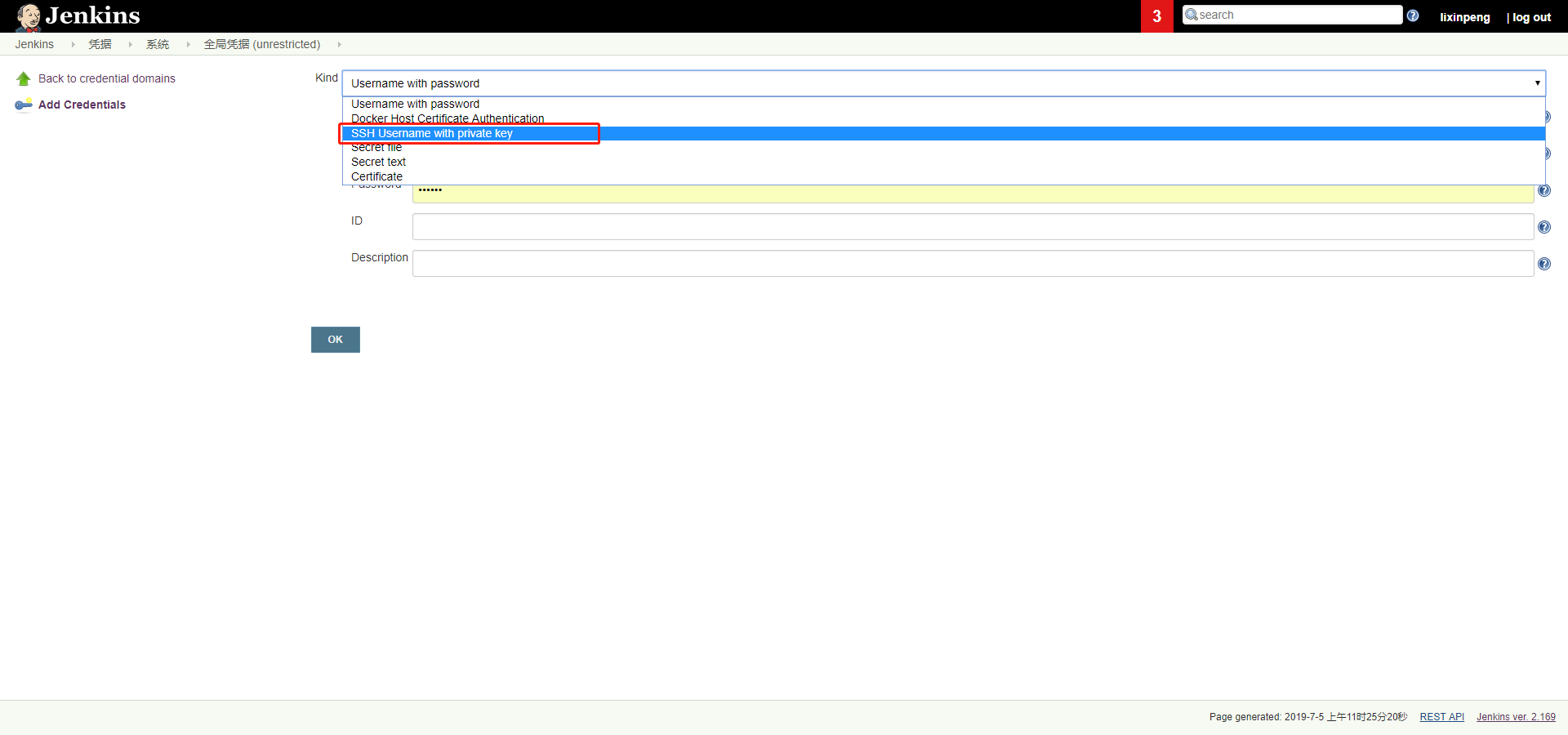
首页Jenkins-> Credentials（凭据），选择右边的Jenkins如下：



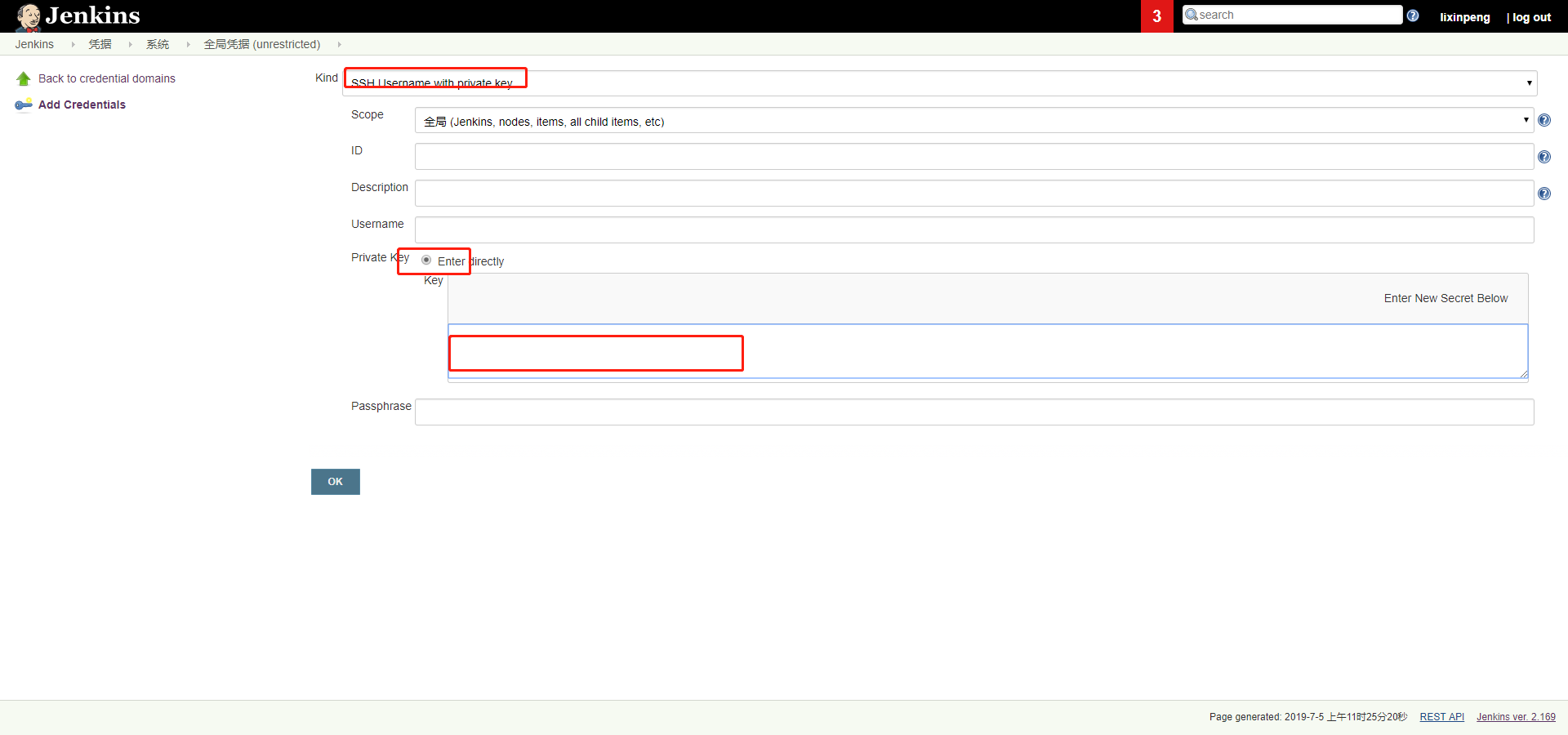
点击系统出现系统界面，点击添加凭据



选择ssh privat key（私钥）

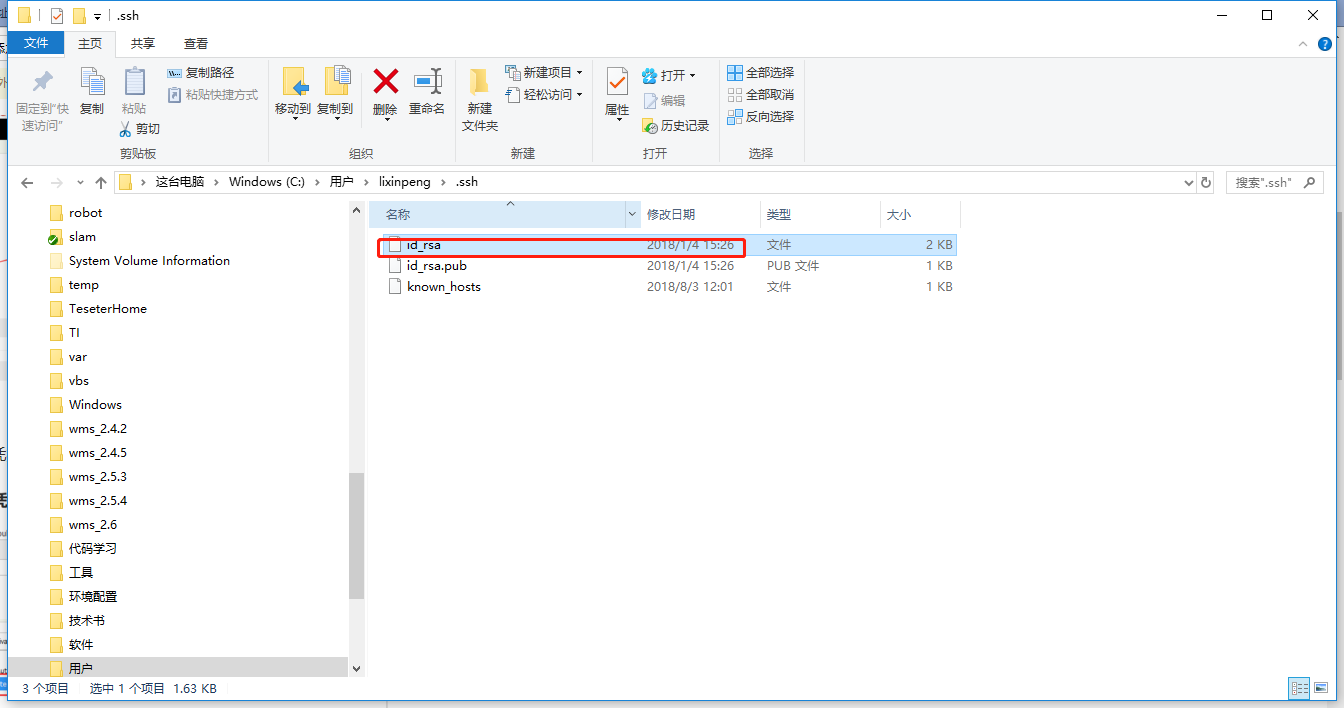


选择enter direckly

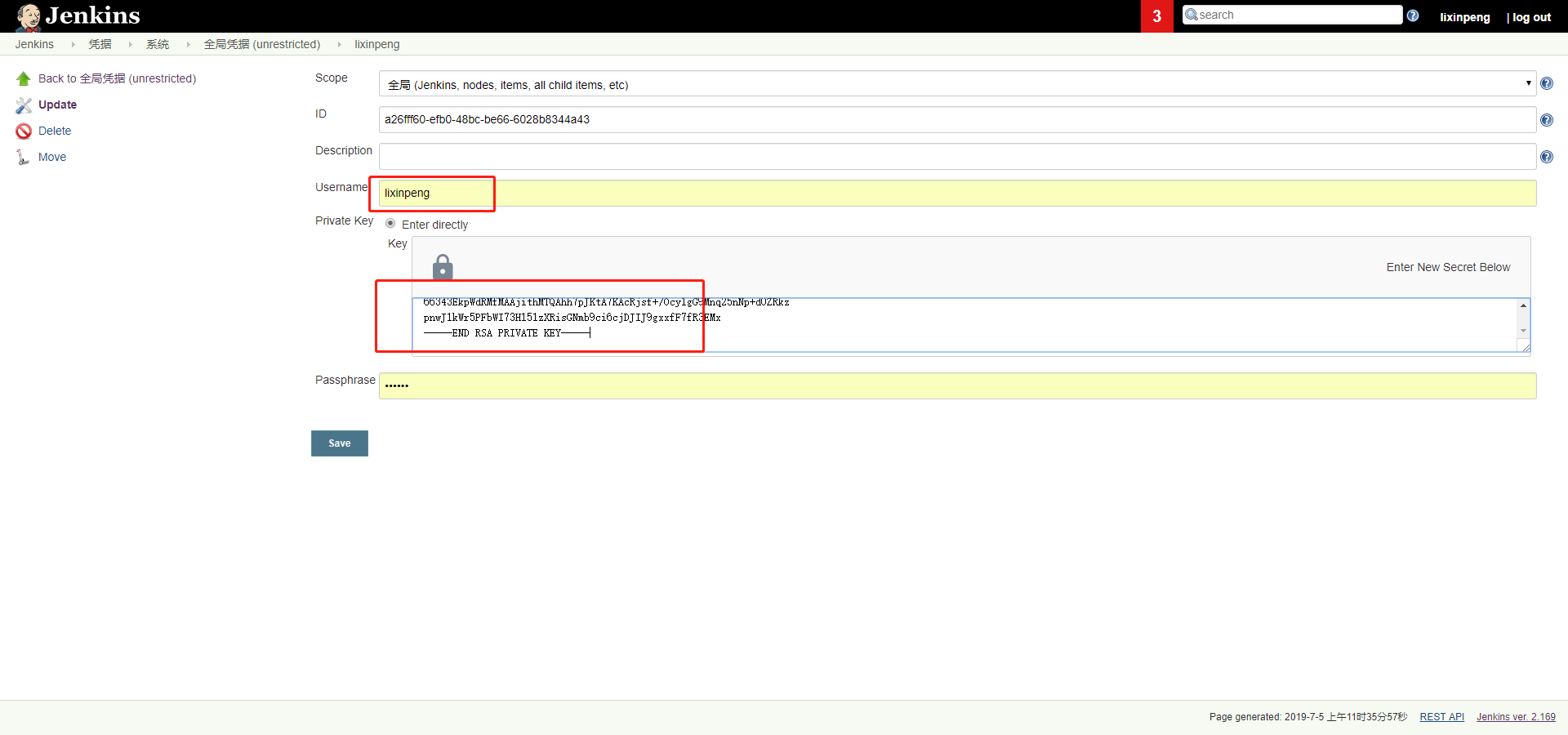


在主机上找到私钥id\_rsa，将内容粘贴进去即可

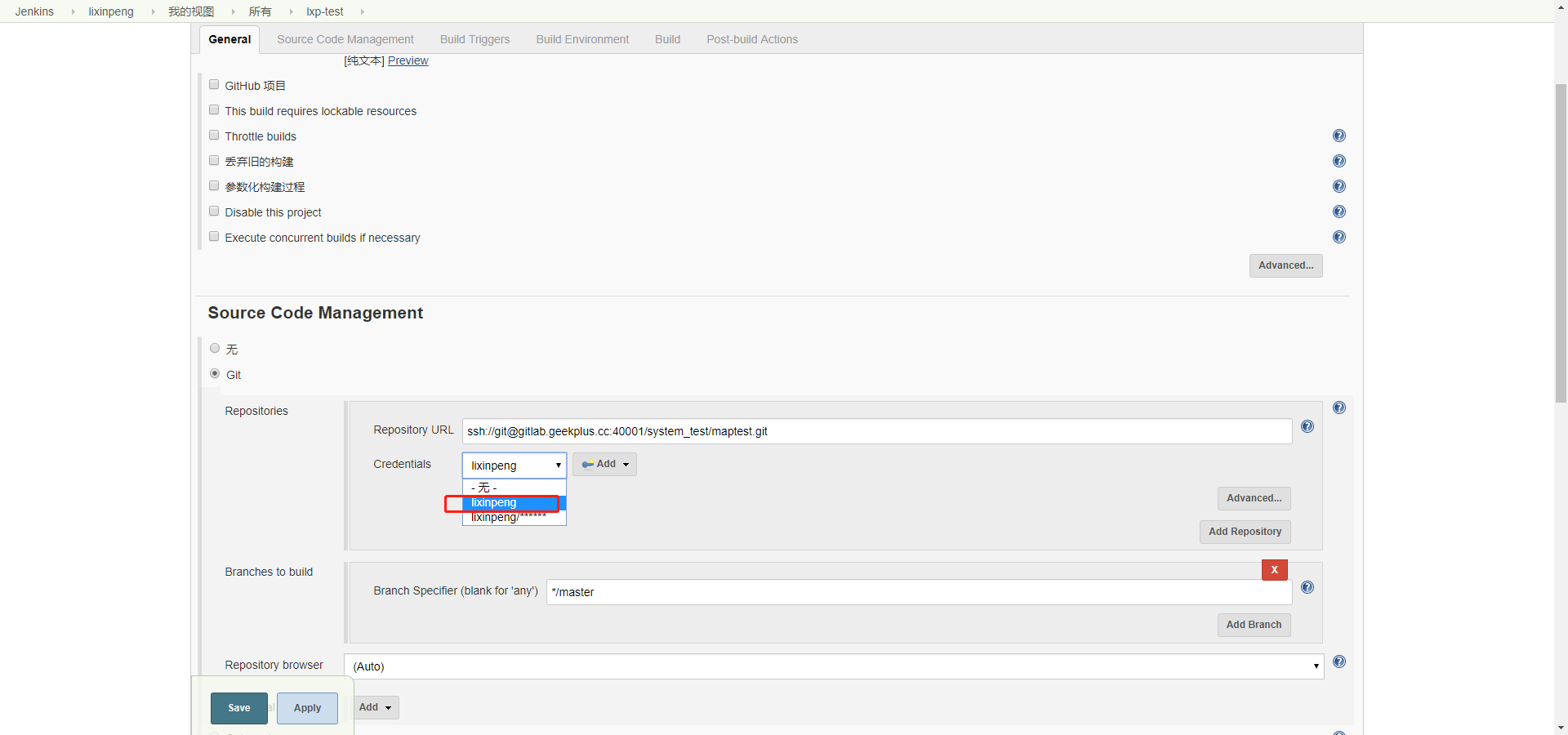
如下图



输入凭据的 用户username



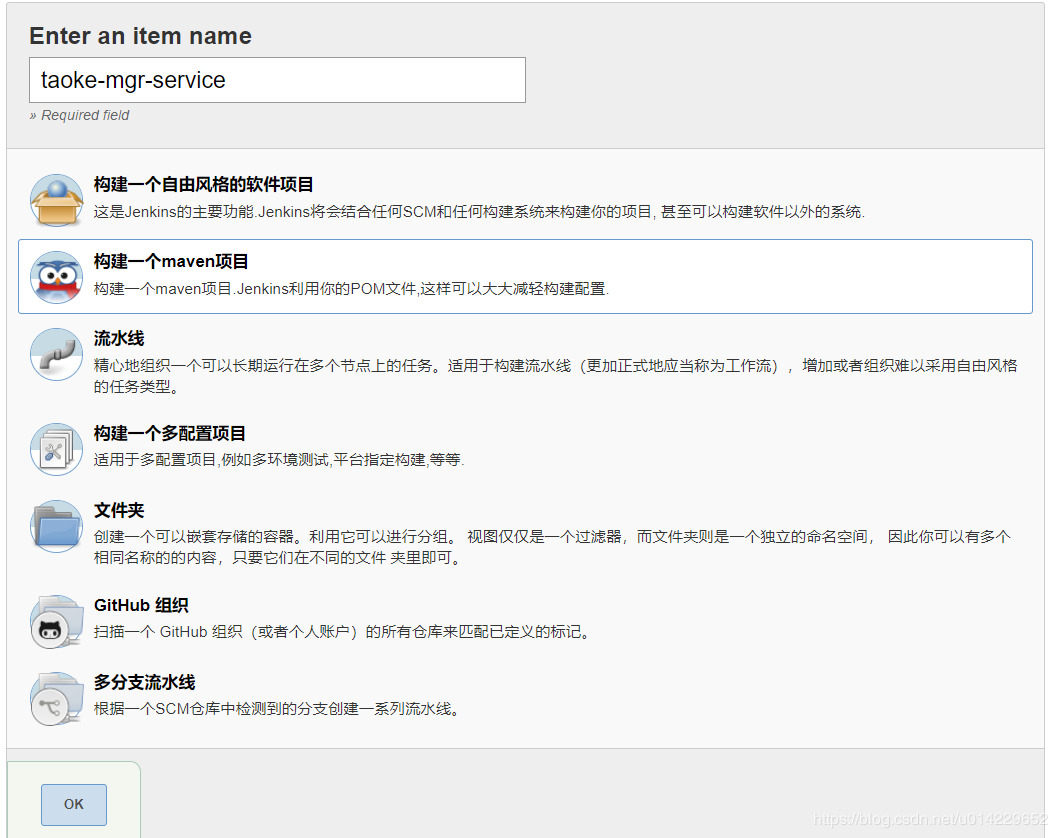
部署的时候，选择自己添加的用户对应的密钥,不会报错



7、部署项目

1. 、首页点击新建：输入项目名称

下方选择构建一个maven项目，点击确定。



②、丢弃旧的构建：



③、参数化构建过程

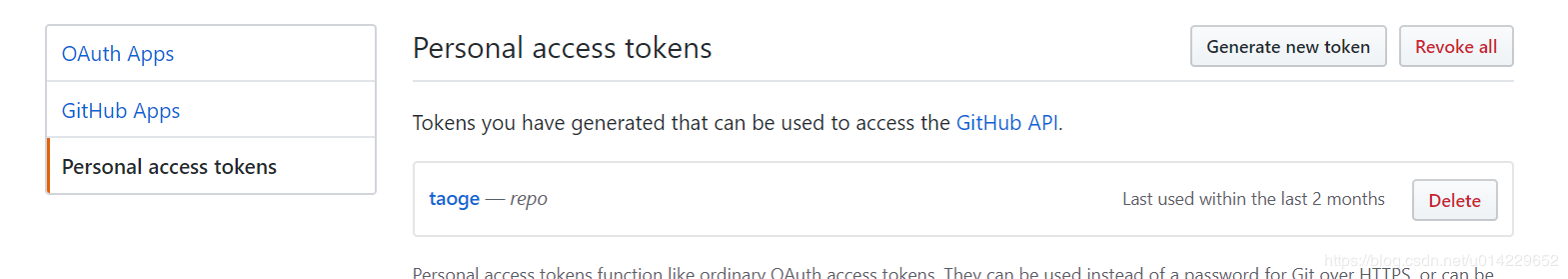


④、源码管理

这里源码用的github

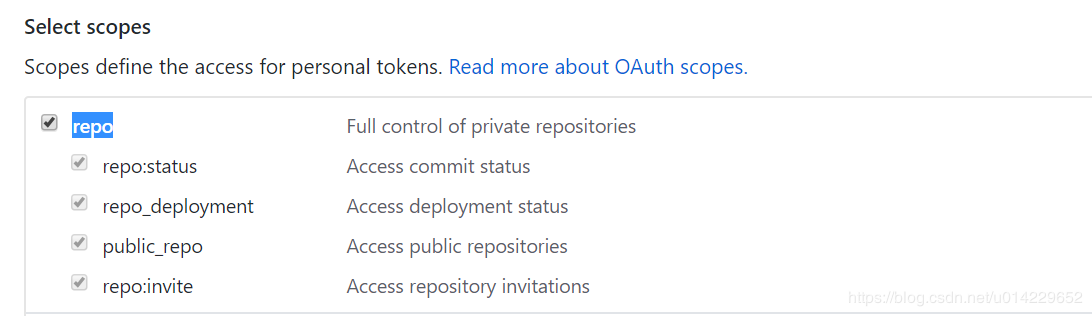
首先登录自己的githhub

用户Settings >> personal access tokens



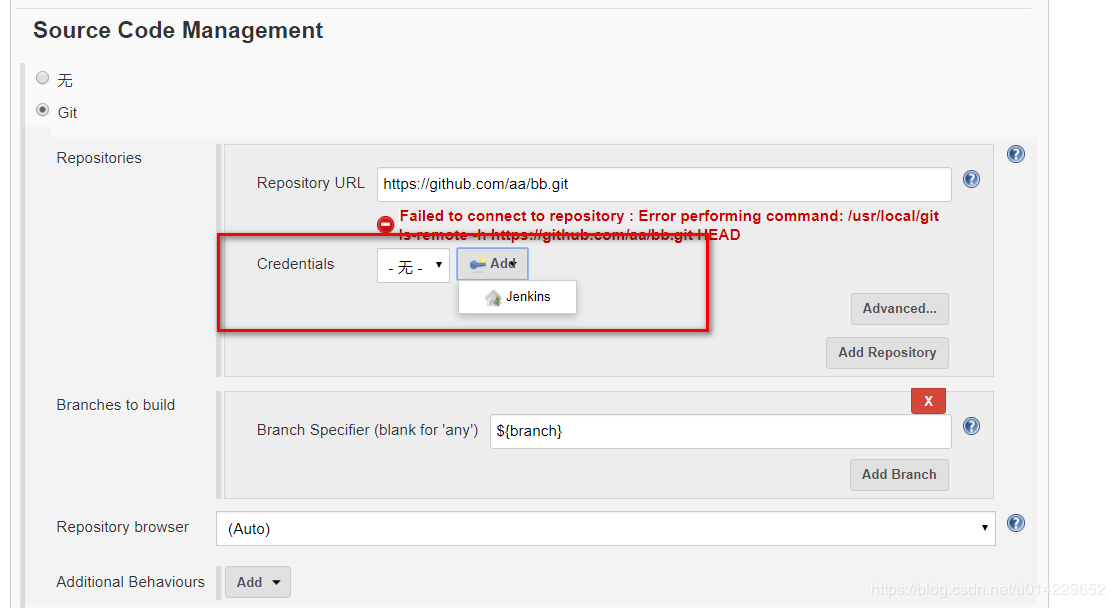
createNewToken

勾选repo

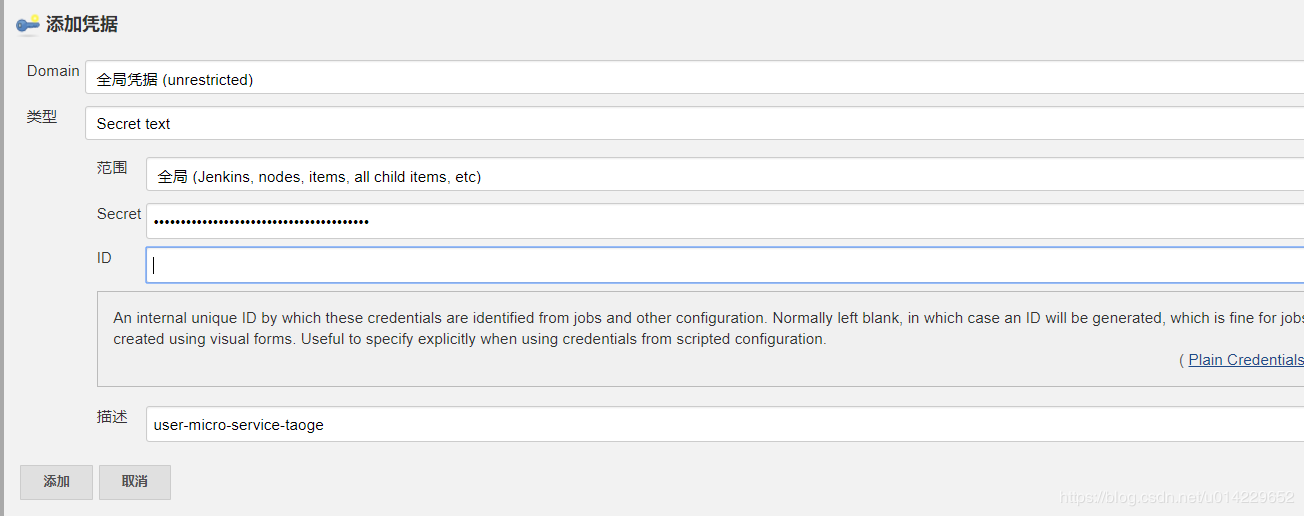


点击Generate token创建一个token

复制token到jenkins

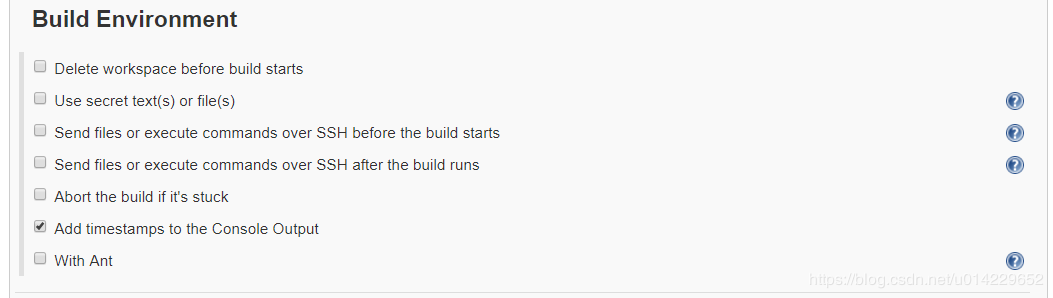


添加token



ID可以为空，点击添加

⑥、构建环境



⑦、Build



下面在Post Steps里添加执行步骤：

⑧、执行shell

BUILD\_DIR=${WORKSPACE}/target

DEPLOY\_DIR=/home/xiuba/taoge/deploy

SHELL\_DIR=/home/xiuba/taoge/shell

SERVICE\_NAME=user-micro-service

PACKAGE\_NAME=${SERVICE\_NAME}-1.0.0-SNAPSHOT.jar

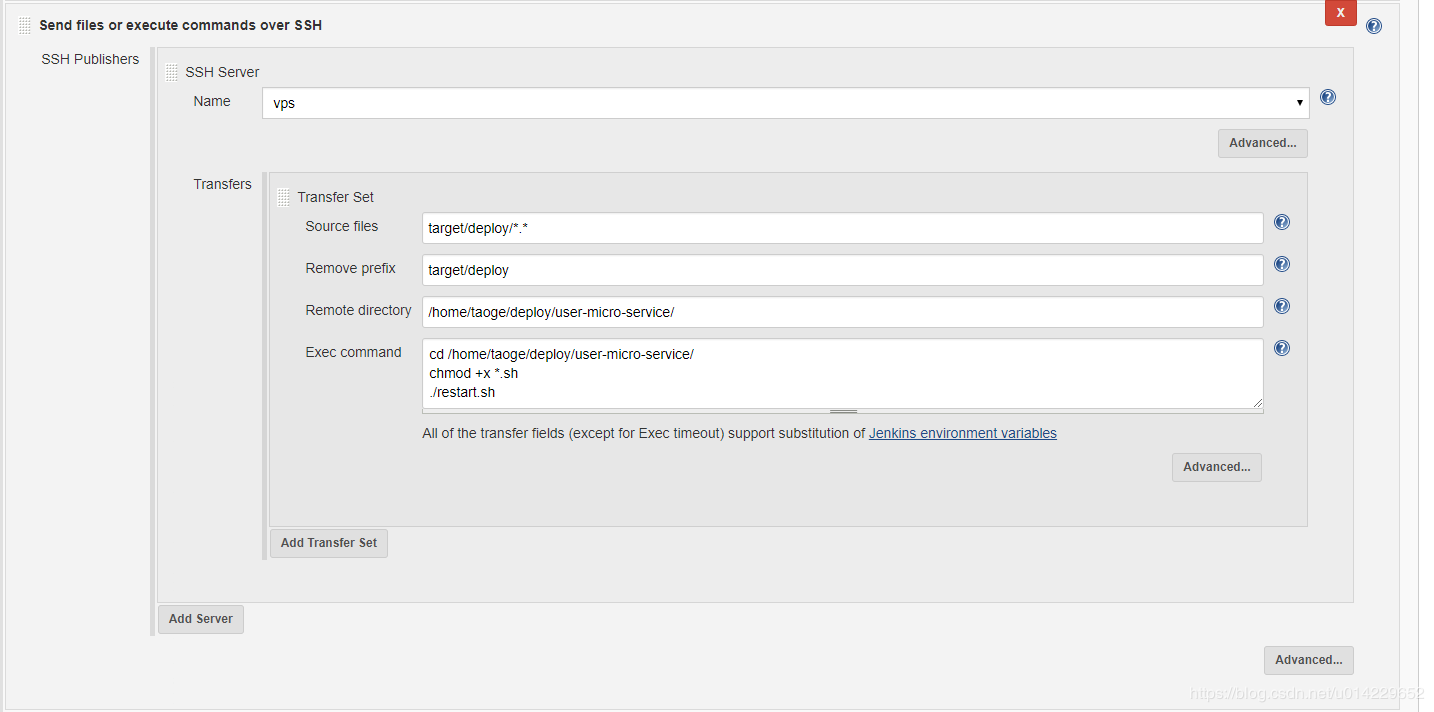
cd ${BUILD\_DIR}

mkdir deploy

cp ${PACKAGE\_NAME} deploy/main.jar

cp /var/lib/jenkins/shell/\* deploy/

⑨、Send files or execute commands over SSH



⑩、sleep



保存

⑪、创建start.sh脚本

cd /var/lib/jenkins

mkdir shell

cd shell

vim restart.sh

#!/bin/bash --login

脚本2

# date:2016-04-18

# 获取进程ID

cur\_dir=$(cd "$(dirname "$0")"; pwd)

echo $cur\_dir

appname=$(basename $cur\_dir)

echo $appname

ProcessID=$(ps -ef |grep $appname|grep -v 'grep'|awk '{print $2}')

echo $ProcessID

RootDir=${cur\_dir}

echo $RootDir

#JVM 参数设置

JAVA\_OPTS=" -server -Xmx512m -Xms256m -XX:+UseConcMarkSweepGC"

#ROCKETMQ 参数设置

#默认把log放到当前工作目录，和其他应用分开，否则默认情况下都输出到1个文件 不利于定位

ROCKETMQ\_OPTS=" -Drocketmq.client.logRoot=$cur\_dir"

#EXEC=nohup java ${JAVA\_OPTS} ${ROCKETMQ\_OPTS} -Dappname=${appname} -jar main.jar > nohup.log 2>&1 &

EXEC=nohup java ${JAVA\_OPTS} -Dappname=${appname} -jar main.jar > nohup.log 2>&1 &

if [[ $ProcessID ]];then # 这里判断进程是否存在

echo "[info]当前通过ps检测进程ID为:$ProcessID"

echo "[error]开始重启tomcat"

kill -9 $ProcessID # 杀掉进程

sleep 3

cd $RootDir

$EXEC

else

echo "[error]进程不存在!开始自动重启..."

cd $RootDir

$EXEC

fi

echo "------------------------------"

31

32

33

34

35

36

好了，现在可以通过jenkins 构建代码并重新启动了，测试。

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

作者：六叔\_

来源：CSDN

原文：https://blog.csdn.net/u014229652/article/details/84867053

版权声明：本文为博主原创文章，转载请附上博文链接！