Jenkins使用总结

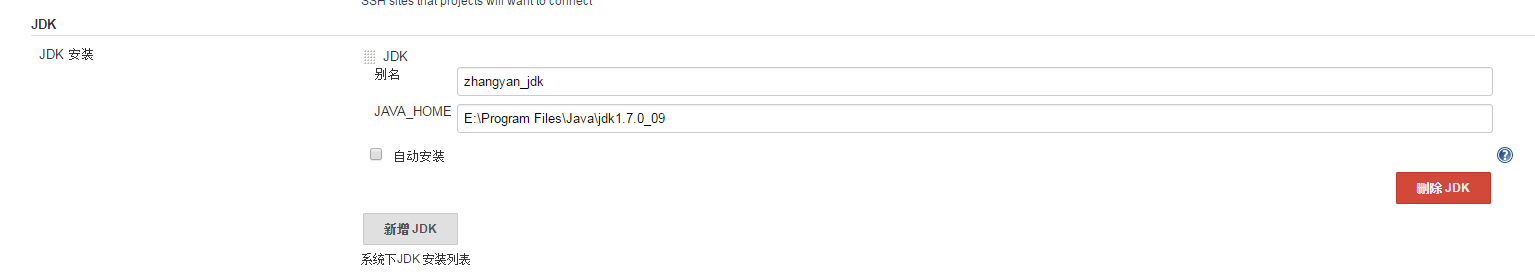
1. **安装jenkins**
2. 从<http://Jenkins-ci.org/上下载最新的jenkins.war>
3. 本地机器安装jdk，tomcat，并且配置jdk的环境变量
4. 本地机器安装git客户端程序，并且配置好环境等
5. 把下载好的jenkins.war放到tomcat中的webapps目录下，然后修改tomcat的conf/server.xm文件，如果Job的控制台中文输出乱码，请将URIEncoding="utf-8"更改为useBodyEncodingForURI="true"

http://images.cnitblog.com/blog/529896/201308/12163628-5e18c80f1c44409fb584a8a618299e12.png

1. 找到tomcat的lib目录，双击运行startup.bat，然后在浏览器中输入localhost:8080/jenkins，回车，就可以看到jenkins的页面了



1. **配置系统常规选项（在jenkins主页面的左侧-系统管理->系统设置中配置）**
2. Jdk安装文件的配置：



1. 自带邮箱配置：



1. **配置用户权限（在jenkins主页面的左侧-系统管理->Configure Global Security中配置）**
2. 注册



设置如上图，保存后系统管理中就出现管理用户的选项。页面右上角也会出现登录/注册的选项。

点击注册：

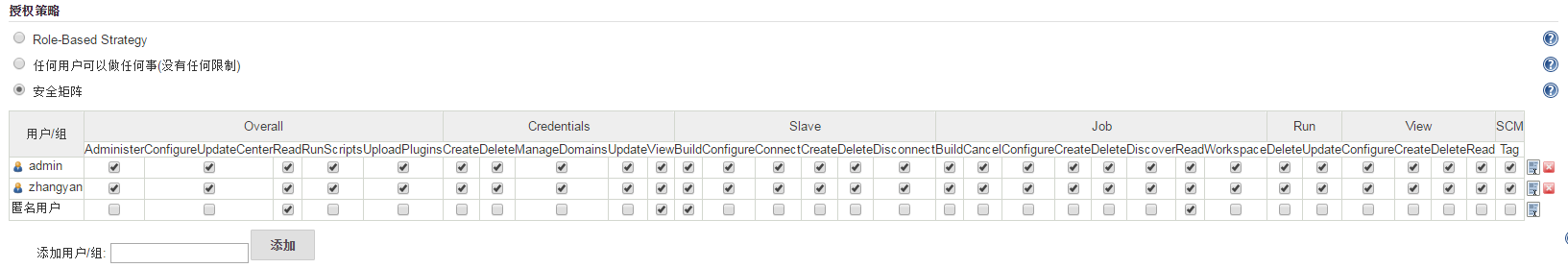


1. 配置用户权限

点击左侧的系统管理—>Configure Global Security ，选择授权策略中的安全矩阵或者项目矩阵授权策略。

说明：安全矩阵和项目矩阵授权策略的配置是一模一样的，唯一的区别是项目矩阵授权策略支持在Job的配置页面再次配置授权策略。

添加用户到矩阵中，如下图。（这里添加的test是注册时候填写的用户名而不是全名和电子邮箱地址！！！在配置权限的时候不要配错）



其中有一些比较特别的权限：

最大的权限是Overall的Administer，拥有该权限可以干任何事情。

最基本的权限是Overall的Read，用户必须赋予阅读的权限，不然什么都看不到。

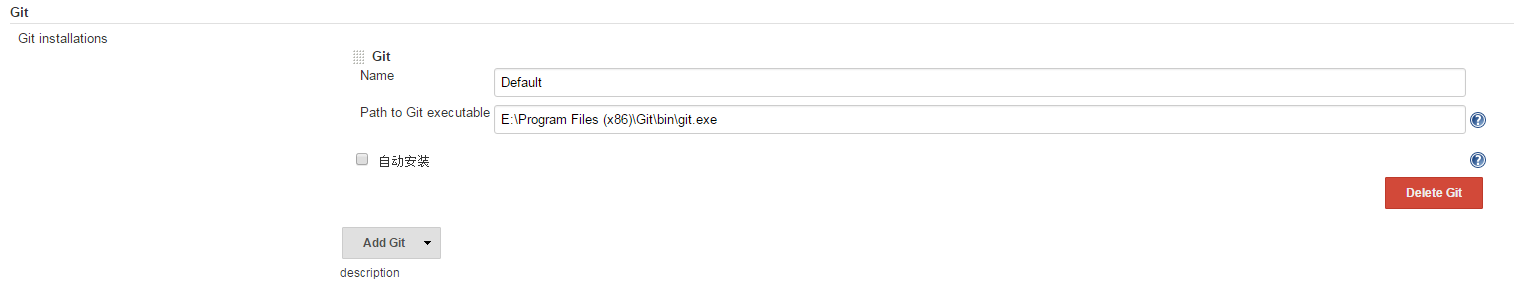
Job的Discover权限是一个奇葩的权限，帮助说Discover比Read的级别更低。如果匿名用户(没有访问job的权限)直接访问一个Job的Url将重定向到登陆页面。(经测试，这个权限应该是被废弃了。)

ps：如果有个用户被赋予了Overall的Read，并没有被赋予Job的Read权限，那么该用户就无法访问job。原因：没有权限。

1. **配置git（使用GIT client plugin、GIT plugin、Git server plugin、GitHub API Plugin、GitHub plugin插件）**

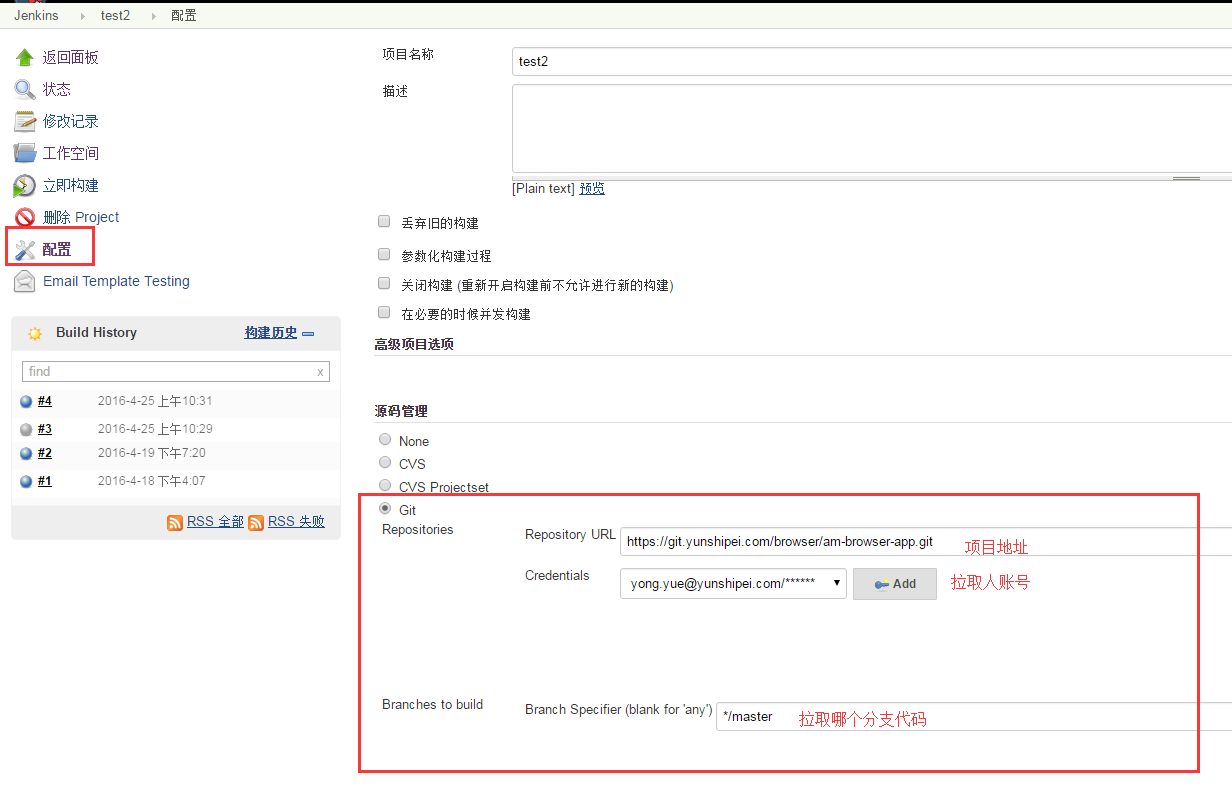
这里插件过多，当时我也忘了具体哪个哪个了

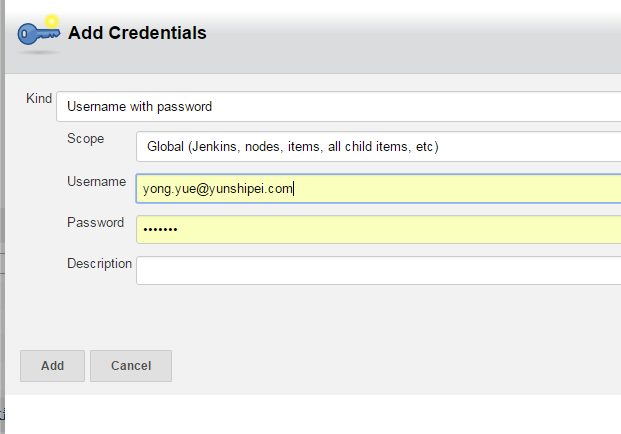
1. 系统管理－系统设置，先设置全局：



配置完git后，每次jenkins从git上拉取代码时候，如果不在项目中配置具体项目的git用户，就会根据本机的git的账号进行拉取代码。

1. 针对具体job名称-配置:



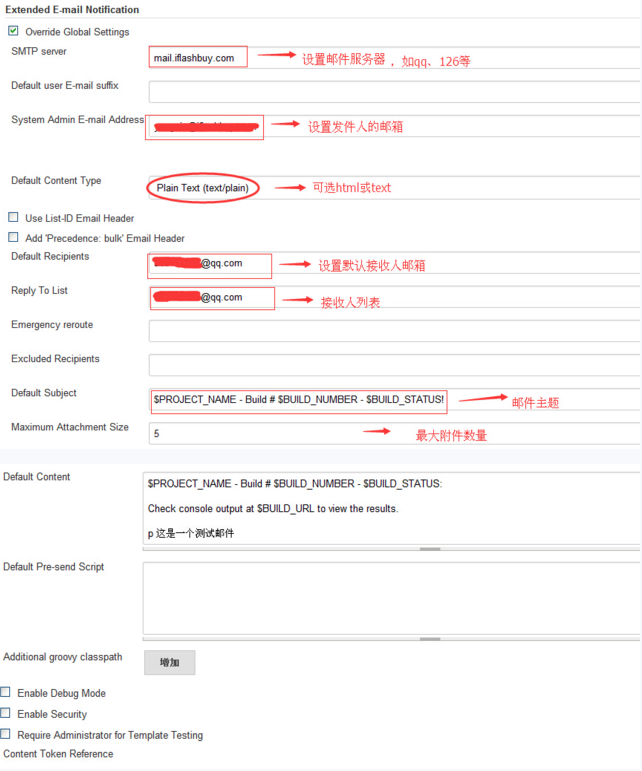


点击拉取人账号的前面的Add时，可以配置具体是哪个账号拉取代码，比如我没有权限获取android的代码，但是我可以在Credentials中配置android开发人员的账号来获取项目权限，这样就可以获取其他人权限的代码了。

1. **配置邮箱（这里可以使用email-ext插件）**

Jenkins默认提供了一个邮件通知，能在构建失败、构建不稳定等状态后发送邮件。但是它本身有很多局限性，比如它的邮件通知无法提供详细的邮件内容、无法定义发送邮件的格式、无法定义灵活的邮件接收配置等等。在这样的情况下，我们找到了Jenkins Email Extension Plugin。该插件能允许你自定义邮件通知的方方面面，比如在发送邮件时你可以自定义发送给谁，发送具体什么内容等等。

1. 系统管理－系统设置，先设置全局：



1. 针对具体job名称-配置：

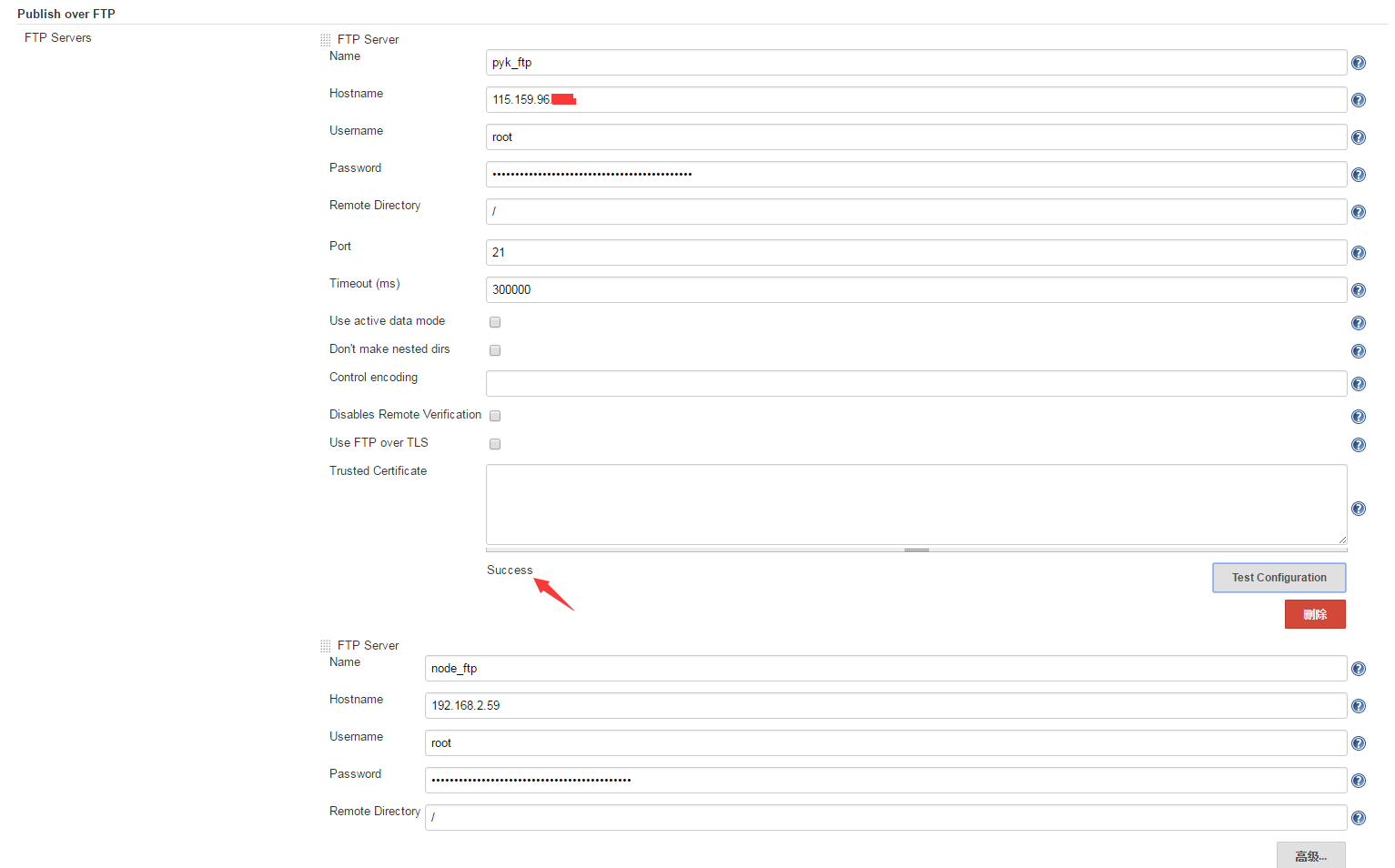


这个插件提供邮箱模板的功能，我们首先要找到C:\Users\a1\.jenkins的下面，如果这个目录有email-templates文件夹，就把模板文件foobar.jelly放到此目录中，如果没有就在此目录中创建一个再存放。

1. **配置ftp（这里用到Publish Over FTP Plugin插件）**

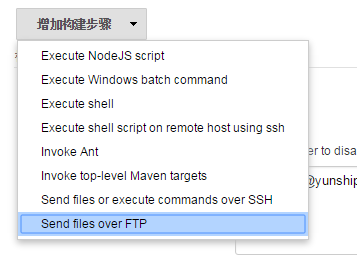
**此种情况是为了应对windows向linux服务器传数据**

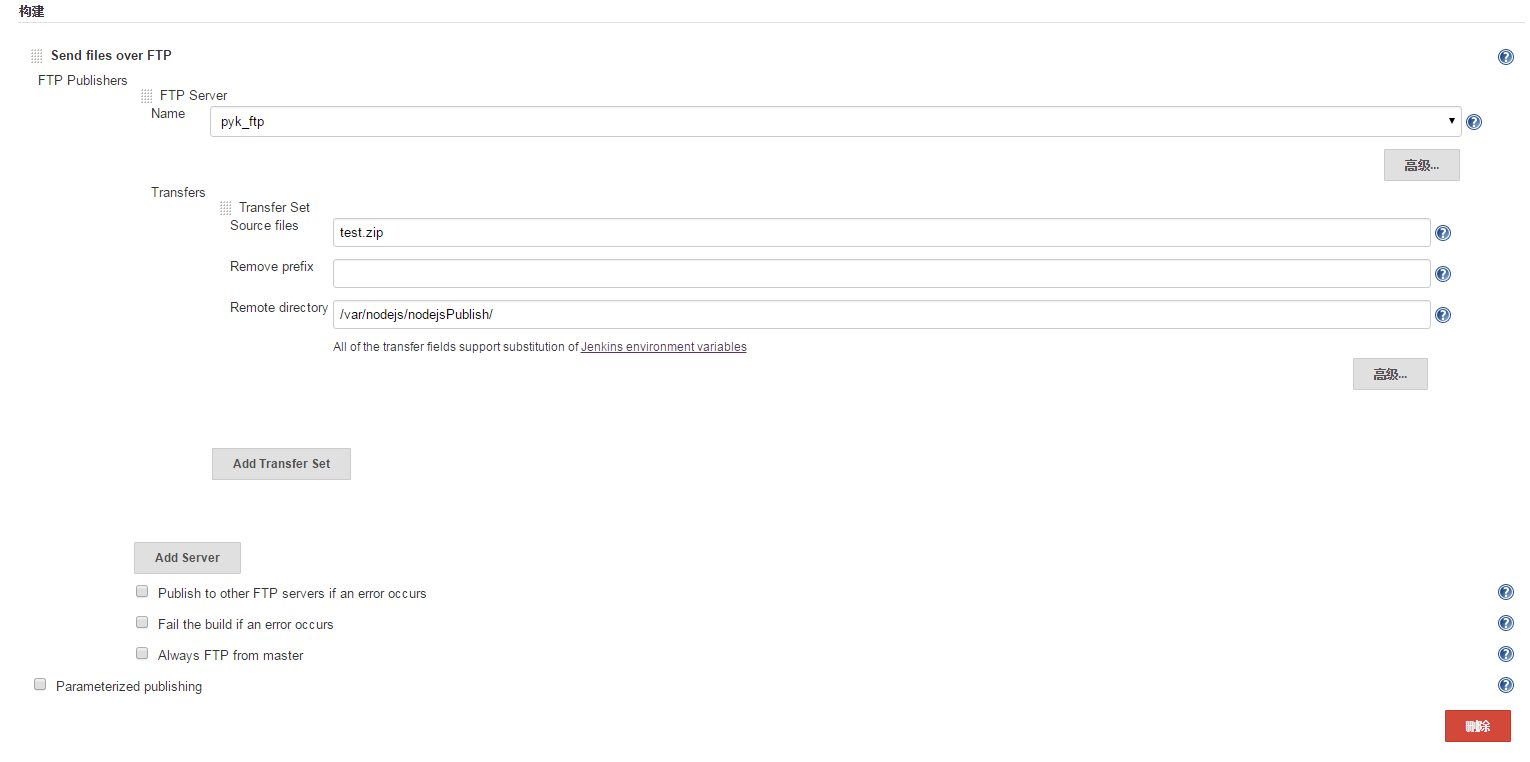
1. 针对全局



配置完成后可以点击Test Configuration进行测试是否可以连接，Success表示可以连接，并且ftp连接可以创建多个。

1. 针对job





在构建的时候可以点击增加构建步骤，添加ftp连接，

注意：这里test.zip的根必须是当前构建项目的文件夹内，比如我当前项目是am-browser-server，那么test.zip的位置是：

C:\Users\a1\.jenkins\workspace\am-browser-server\test.zip

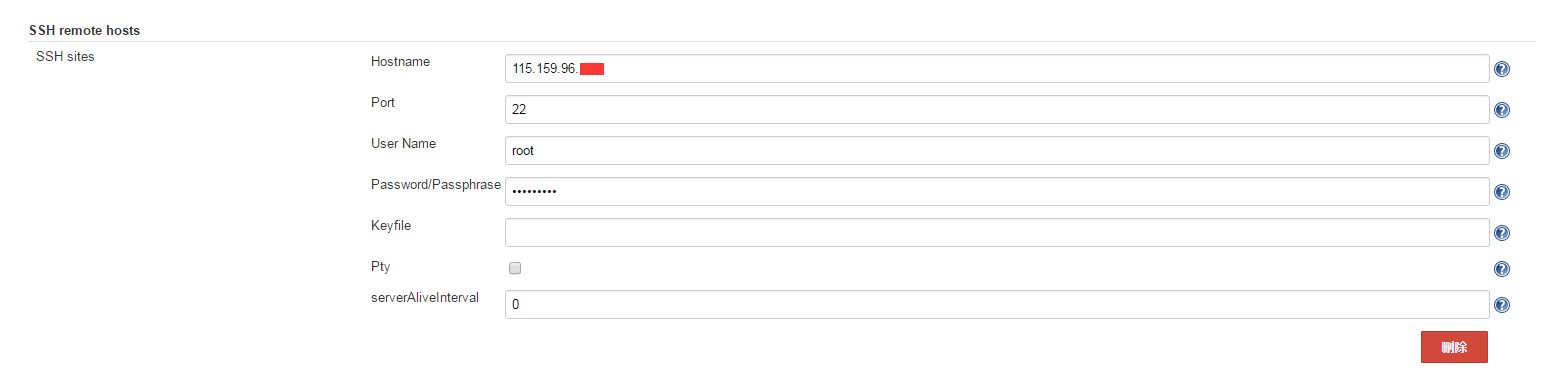
1. **配置ssh（插件：SSH Credentials Plugin、SSH plugin、Publish Over SSH）**

首先，我们可以参照这个教程来安装：

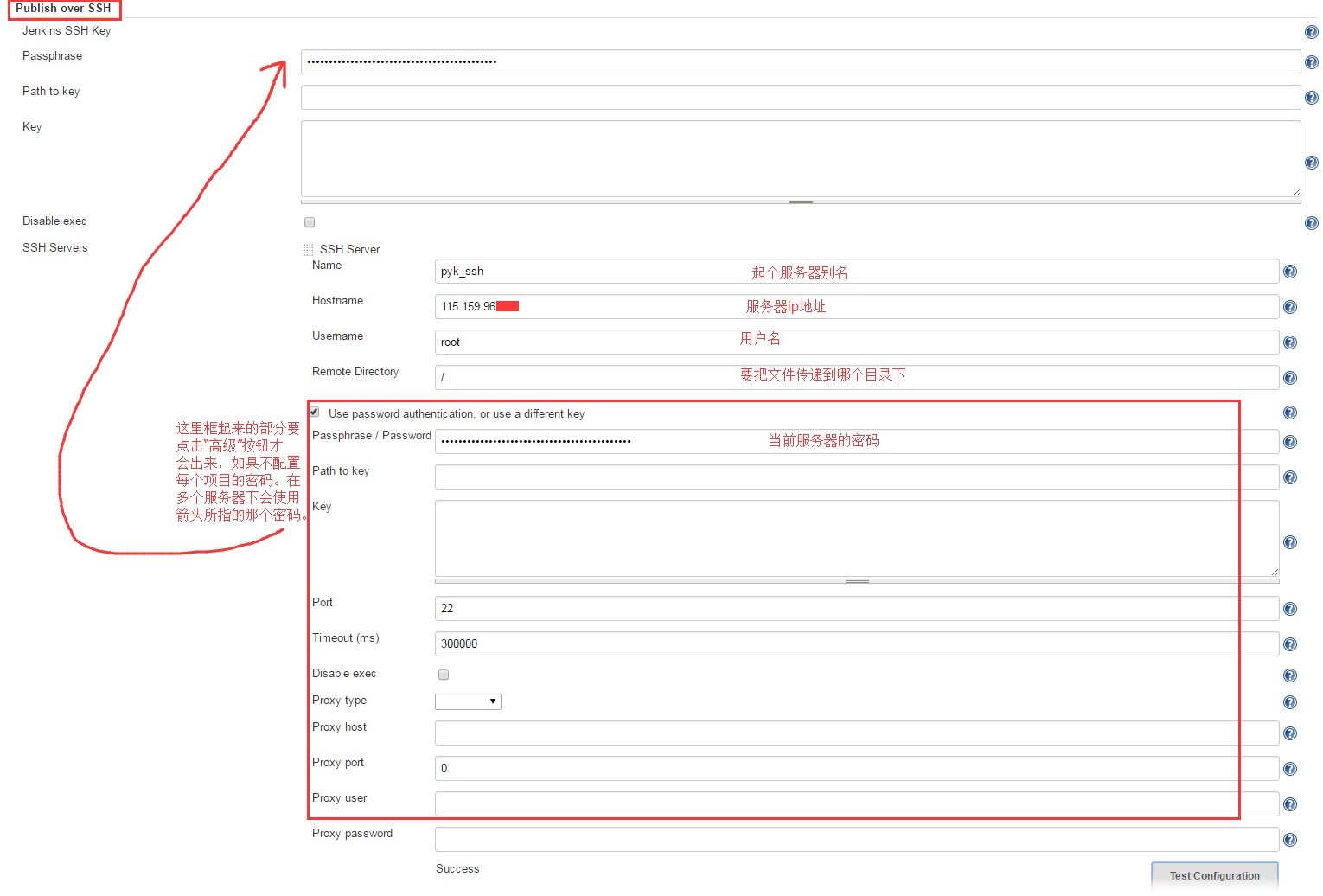
<http://snoopyxdy.blog.163.com/blog/static/601174402014102695720645/>

1. 针对全局

增加远程服务器



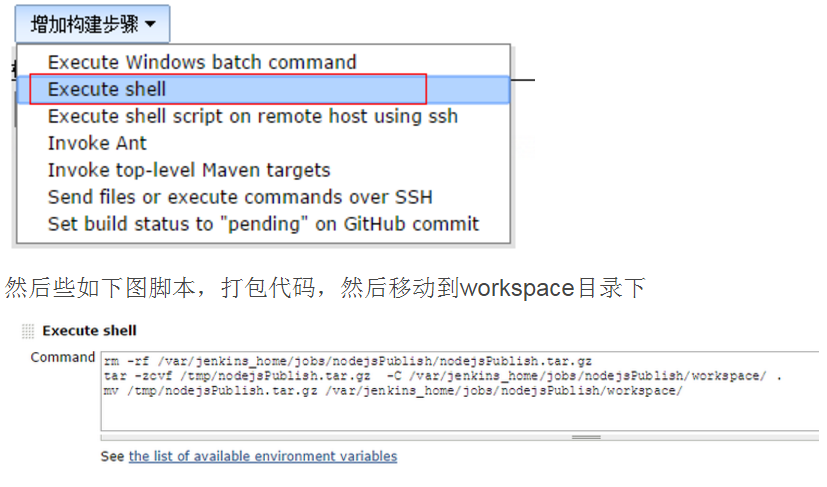
增加远程拷贝文件的远程服务器



1. 针对job

构建完毕之后，将代码打包保存

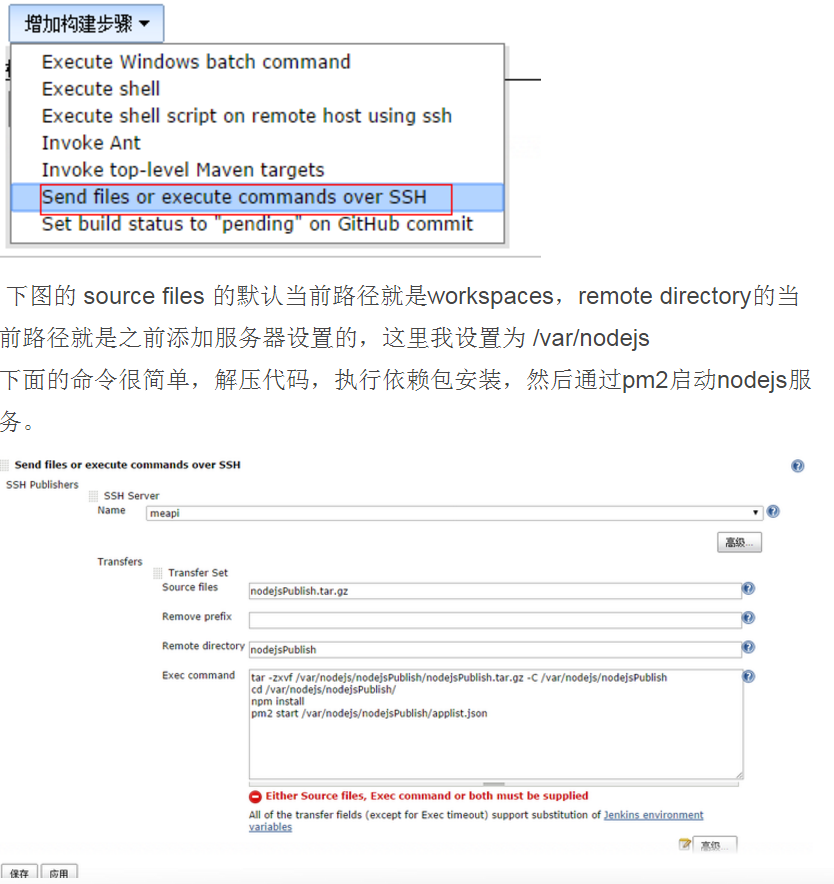
我们先如下图，增加一个构建完毕的shell脚本执行



接着我们需要连接到部署服务器上，将原来的服务暂时暂停，删除上一次部署的压缩包，选择添加远程shell脚本执行，如下图

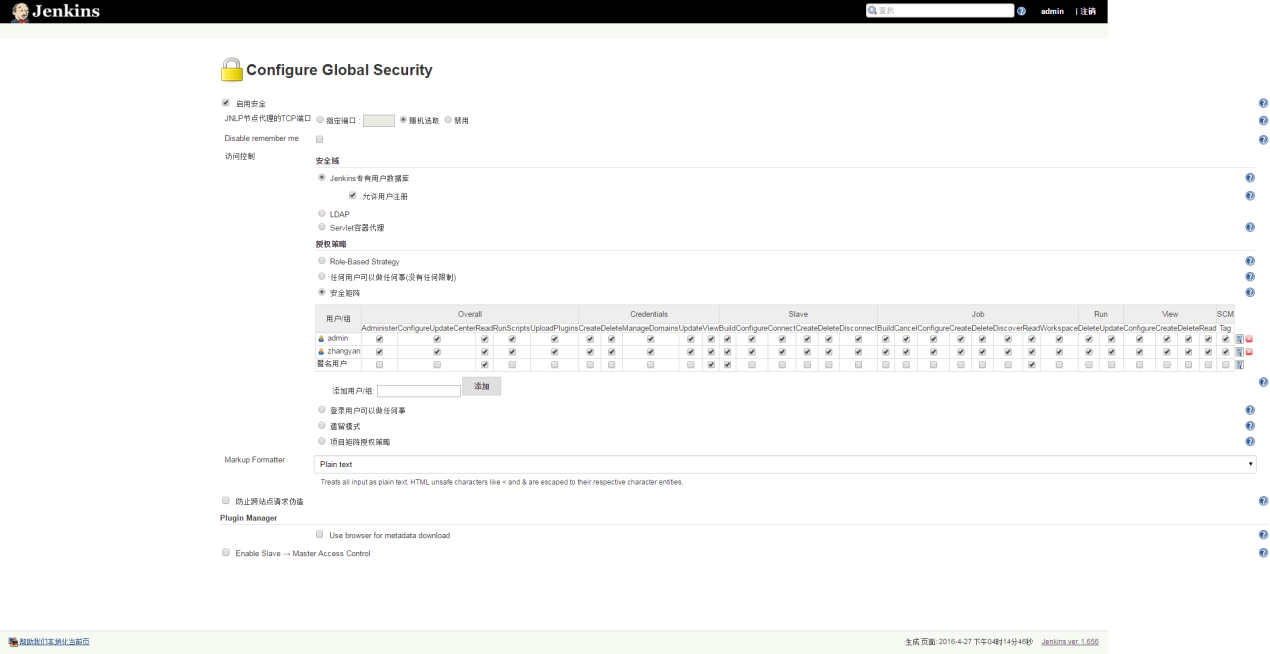


最后我们要把打包好的代码发送到服务器上，然后执行安装依赖包操作，并且启动nodejs服务，选择通过ssh发送文件并执行命令到远程服务器，如图

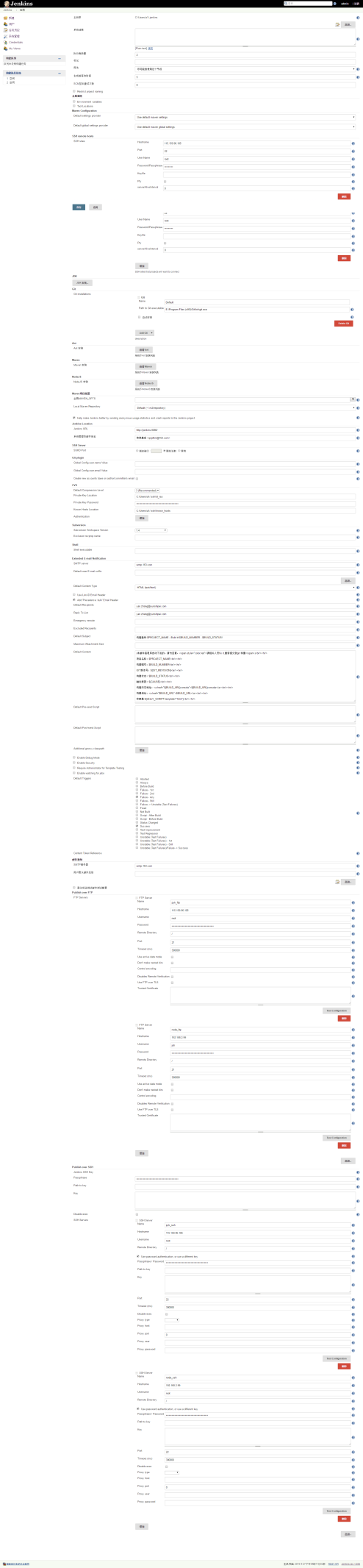


1. **整体走一遍job的配置**

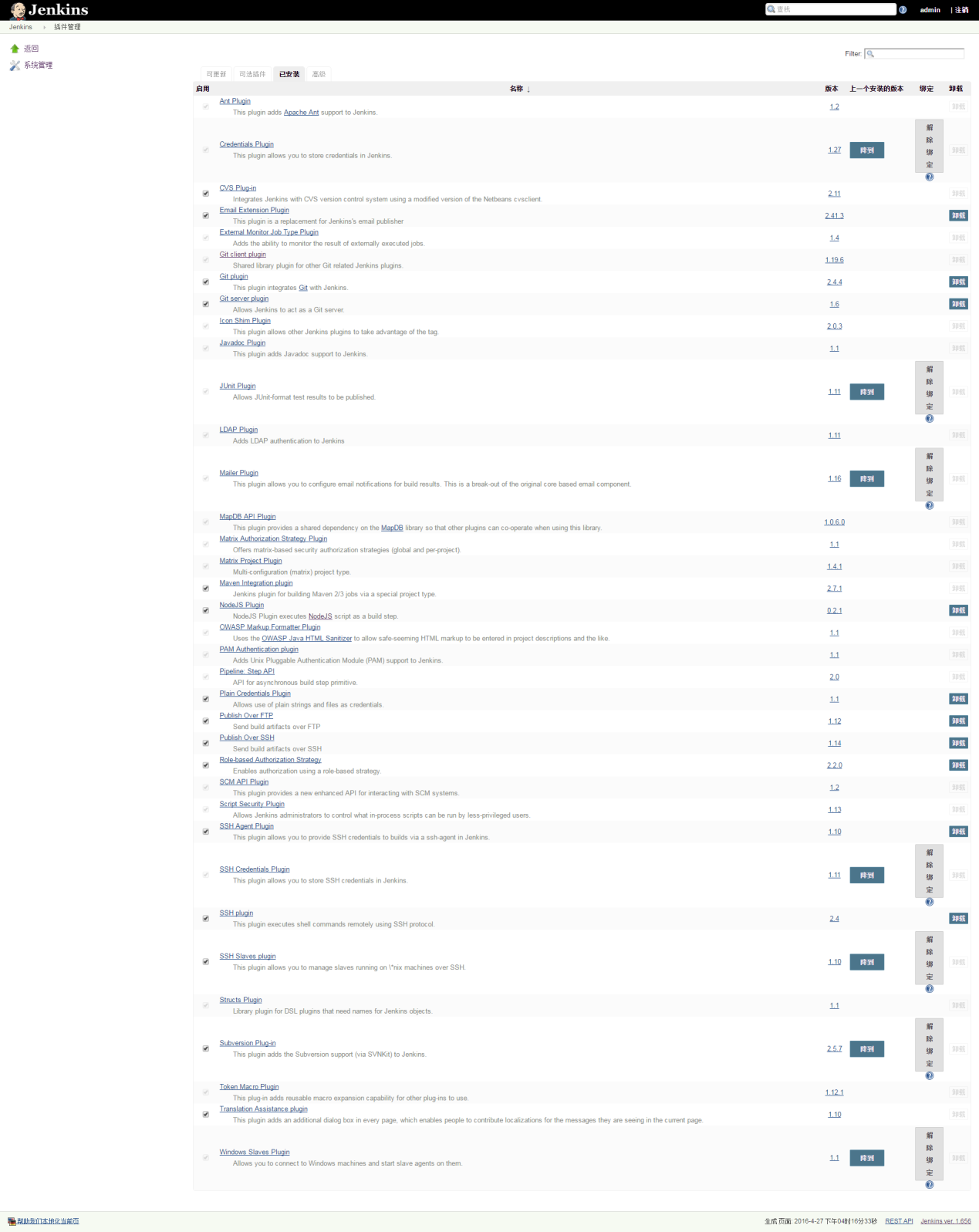
用户：



系统：



插件：



Job：

