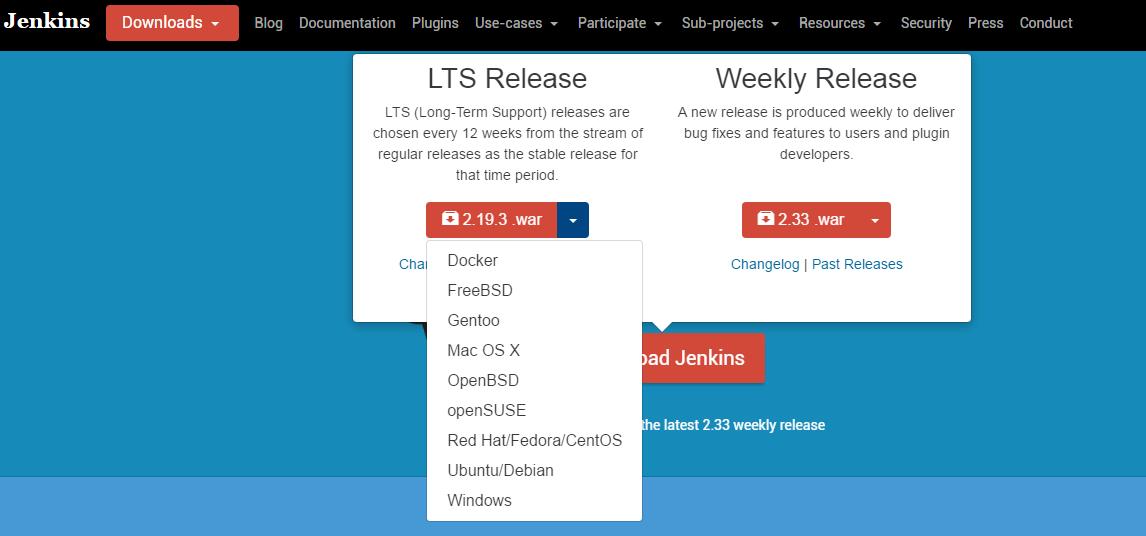
Jenkins环境搭建、配置及使用说明

Jenkins 官网地址：<https://jenkins.io/index.html>

最新的稳定发布版本为2.19.3（本说明基于本版本）



需要的环境：

1. JDK，Jenkins是java程序，因此需要安装JDK；
2. MAVEN，想要在Jenkins中使用maven就必须安装。

# 安装Jenkisn

下载Jenkins的Windows版本，直接运行msi文件，即可完成安装，Jenkins以后端服务的形式运行。

安装完成以后，浏览器输入：<http://localhost:8080> 就可以看到Jenkins的界面了。

# 安装插件

常用插件说明：

1. subversion插件（从svn检出项目必备）

subversion插件地址：

<https://wiki.jenkins-ci.org/display/JENKINS/Subversion+Plugin>

1. Deploy to container Plugin 插件（部署war/ear必备）
2. Publish over ssh插件（通过ssh传输文件以及执行远程shell命令）
3. Maven插件（创建maven构建的项目必备）
4. JUnit Plugin 插件（生成测试报告）
5. 其他……

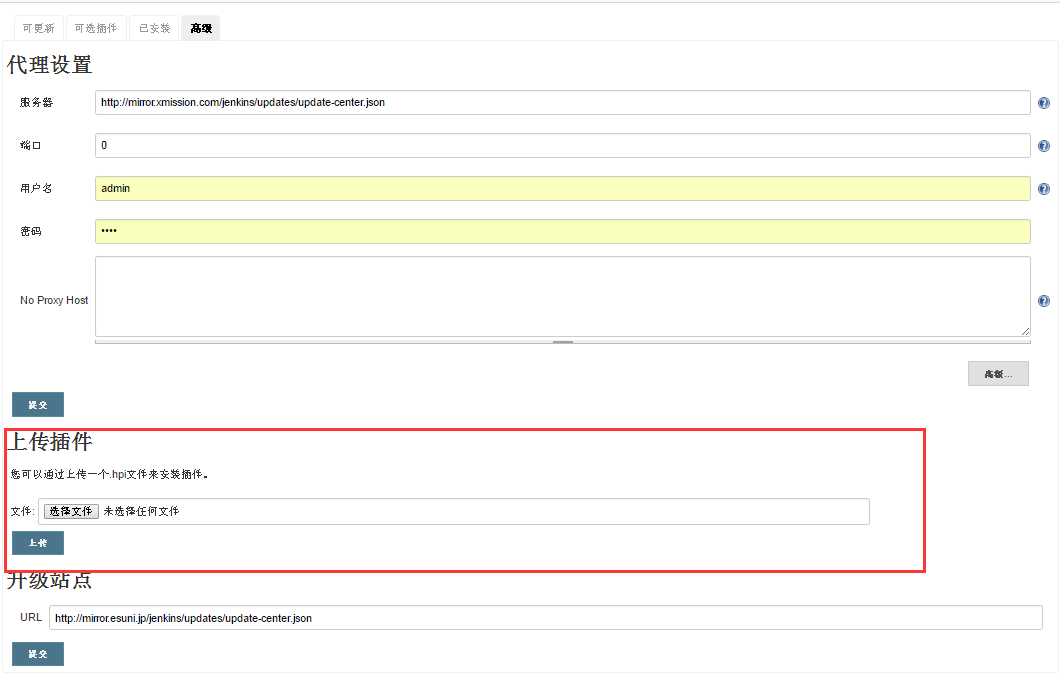
## 安装步骤

1、点击主界面的“系统管理”进入系统管理页面，选择“管理插件”



默认安装完Jenkins以后，无法读取到可用的安装插件，这里选择手动上传安装的方式进行安装，请先将需要安装的hpi插件文件下载到本地。

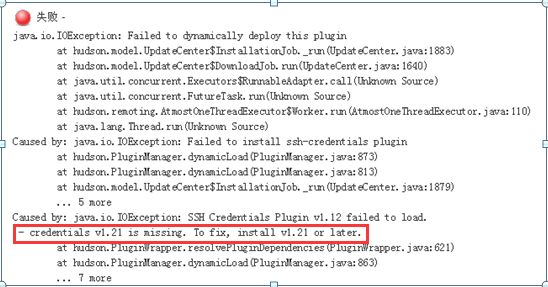
2、在管理插件页面选择“高级”tab选项，如下图所示，点击“选择文件”然后点击“上传”即可进行插件安装。



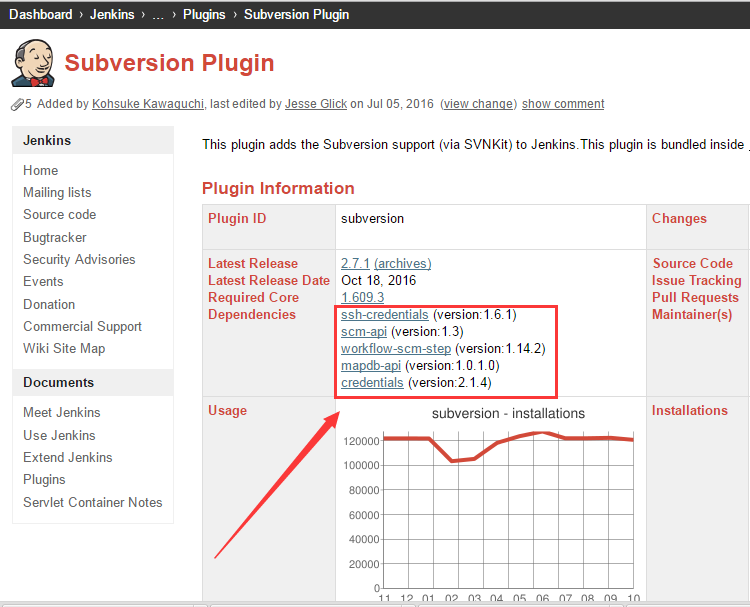
在插件的安装过程中，可能会出现错误，如下图在安装时出现了credential v1.21 is missing的错误，这是因为subversion插件需要依赖credentail插件，所以我们在安装插件时，需要先将依赖到的插件安装好 。

插件的依赖是传递的（如安装插件a，依赖了插件b，而插件b又依赖插件c，安装顺序为先安装c，再安装 b，再安装a）。

如下图所示，如果没有预先安装红色框内的依赖插件及子插件，subversion插件是无法安装成功。



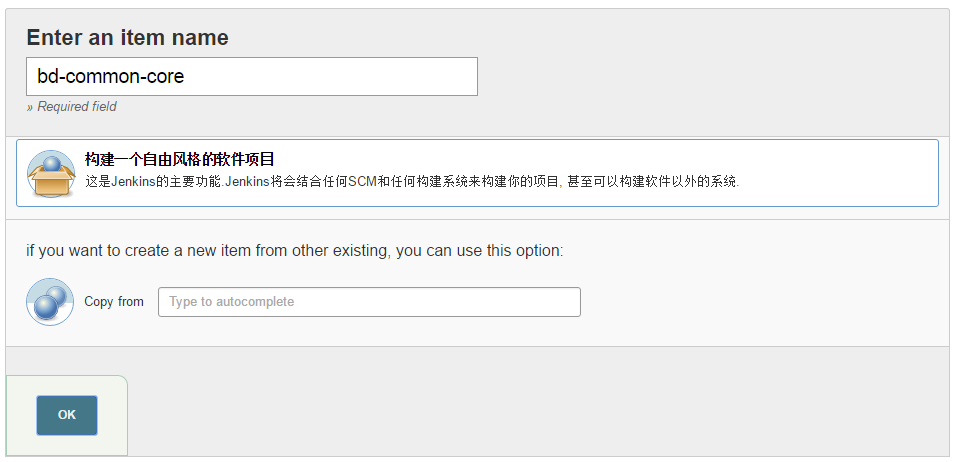
安装SVN插件



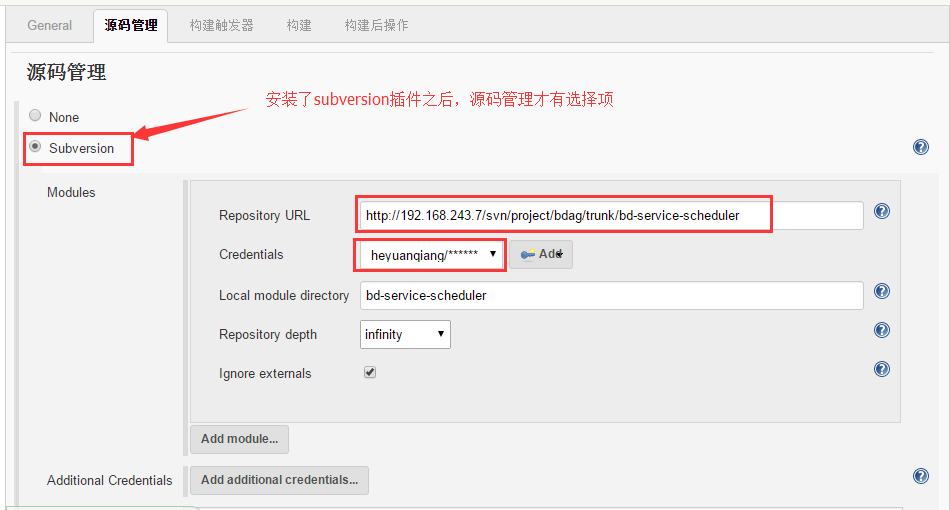
# 创建项目

## 创建一个自由风格的项目

1、主界面，点击“新建”，在创建页面填写项目名称，选择自由风格的项目，然后点击“OK”



2、源码管理中，有以下参数需要设置

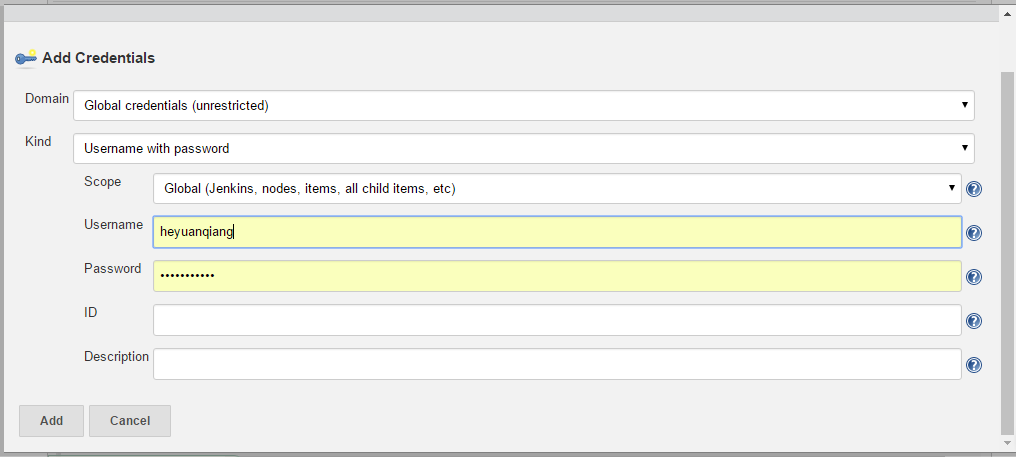


1. 代码的仓库地址RepositoryURL，这里直接填上代码的svn地址即可；

如本示例图填写的是项目bd-service-scheduler的svn地址。

1. 登录证书Credentials，一般来讲采用用“户名和密码”方式登录；

点击上图中Credentails后面的“Add”按钮即弹出登录方式方式的配置框，白色框中的项采用默认即可，黄色的用户名和密码框填入访问第一步填写的svn对应的用户名和密码即可。



1. 本地module directory(本地代码目录)。

这里只能填写相对目录，实际的绝对目录地址是：\*\*/Jenkins/workspace/*Local modue directory*

Jenkins会从第一步填写的URL将代码check下来，存放到本目录。

1. 点击保存回到Jenkins主界面就可以对刚创建的项目进行手动编译了。



鼠标移到name列的项目标题上，标题后面会出现一个下拉的小三角，点击可选择相应的快捷操作，如下图：

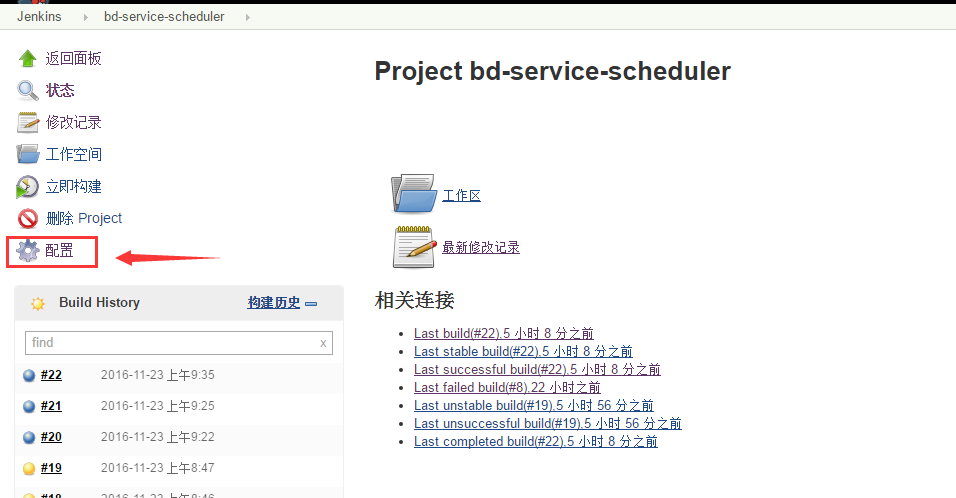


图例表示的意思如下（其中浅色透明的球呈现时是一闪一闪的）：

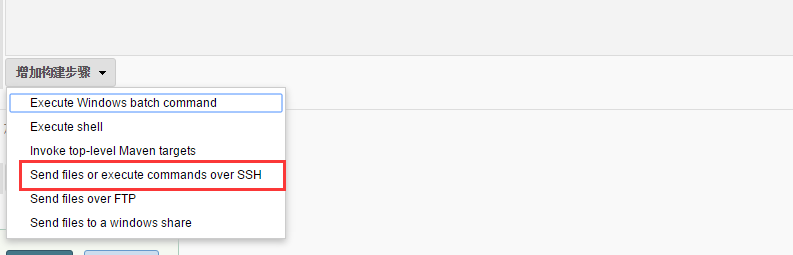


## 创建构建动作

构建后，将编译的jar包通过SSH打包到远程服务器，步骤如下：

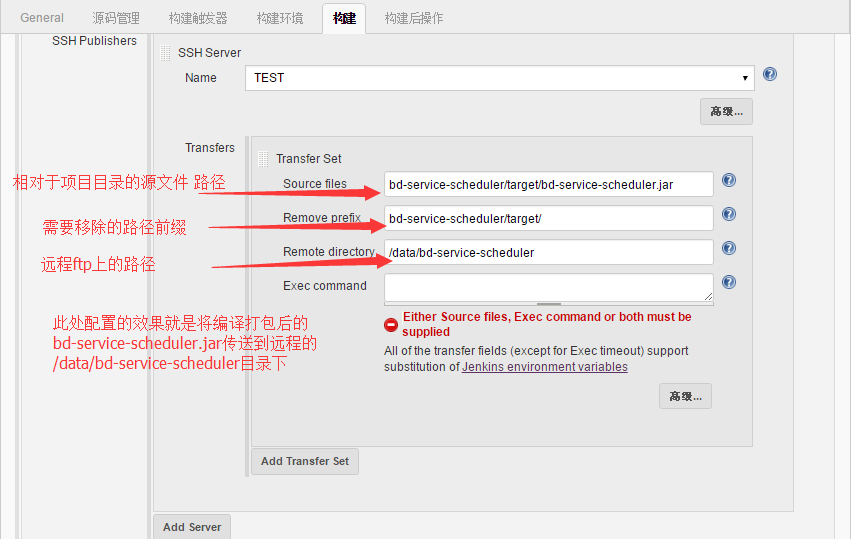
1、Jenkins主界面点击对应的项目，进入项目主界面，在项目主界面左侧点击“配置”

2、进入项目配置页面，点击增加构建步骤



3、在弹出框中加入如下配置：

下图中的配置实现将编译打包生成的bd-service-scheduler.jar 上传动SSH SERVER（TEST）。



4、保存，回到面板页面，执行编译，完成之后，在控制台的看输出如下输出表示构建成功：

