

# 持续集成 之 Jenkins - 许振坪的专栏 - 博客频道 - CSDN

作者：许振坪，<http://blog.csdn.net/benkaoya>

## 持续集成

有关持续集成的简介，可参考我此前的文章：<http://blog.csdn.net/benkaoya/article/details/44993583>

## Jenkins简介

官网：<http://jenkins-ci.org/>

### 功能与优点

Jenkins是基于Java开发的一种持续集成工具，用于监控持续重复的工作，功能包括：

- 持续的软件版本发布/测试项目。
- 监控外部调用执行的工作。

跟其他持续集成相比，它的主要优点有：

- 开源，即免费。
- 支持多种平台（windows、linux、os x都支持）。
- 安装、配置简单。
- Web可视化管理界面，并且有丰富的tips帮助信息。

### Jenkins和Hudson的关系

Jenkins的前身是Hudson（Sun开发），2010年从Hudson分支出来。由于Sun被Oracle收购，后者声称拥有Hudson的商标所有权，前者被迫这样做的。分支出来的Jenkins将继续走open source道路，而Hudson可能会用于商业用途，二者都在继续开发中

## 我的环境

如果对在Linux下安装有恐惧症的，BitNami提供了jenkins的一键安装程序，简单、易用、方便，可在<https://bitnami.com/stack/jenkins>中下载，windows、linux、OS X平台都有提供。我是在linux环境下安装的Jenkins，环境如下：

所需软件	版本
服务器操作系统	Debian
Jenkins软件版本	bitnami-jenkins-1.617-0-linux-installer.run

## 安装

### 安装Jenkins

由于用的是bitnami-jenkins，所以安装过程很简单。将bitnami-jenkins-1.606-0-linux-installer.run拷贝至linux中，增加可执行权限，然后直接运行该安装文件即可。

安装向导很简单，这里不介绍。需要注意的是：务必记录下安装过程中设置的管理员账户和密码，等下要用到。

### 启动Jenkins服务

安装完bitnami-jenkins，最后一步就会提示你启动Jenkins服务。除了可以用这种“图形界面”的方式来启动Jenkins服务之外，还可以通过“命令行”方式来启动/停止Jenkins服务。

```
#/home/ci/jenkins-1.617-0/ctlscript.sh status          /*查看Jenkins服务状态*/    #/home/ci/jenkins-1.617-0/c
tlscript.sh start          /*启动Jenkins服务*/    #/home/ci/jenkins-1.617-0/ctlscript.sh stop          /*停
止Jenkins服务*/
```

## 开始体验

### 管理员登陆

安装完，在浏览器中输入<http://localhost:8080/jenkins>，就能看到效果了，如下图所示，如果无法打开，就要查看日志了，日志输出到安装路径的log目录下。

 **Jenkins**

Jenkins ▶

用户名:

密码:

☐ 在这台计算机上保持登录状态

登录

[创建一个用户账号](#) 如果你没有注册用户.

输入管理员用户名和密码即可登录：

 **Jenkins**

? Username LastName | 注销

Jenkins ▶ [关闭自动刷新](#) [添加说明](#)

 新建

 用户

 任务历史

 系统管理

 Credentials

 My Views

## 欢迎使用Jenkins!

开始 **创建一个新任务**

**构建队列** —

队列中没有构建任务

**构建执行状态** —

1 空闲

2 空闲

## 新建一个任务

点击“新建”，新建一个任务。我要通过shell进行编译、构建，所以新建任务时选择的类型是“构建一个多配置项目”，如下图所示。



The image shows the Jenkins 'New Item' dialog. On the left is a sidebar with navigation links: '新建' (New), '用户' (Users), '任务历史' (Task History), '系统管理' (System Management), 'Credentials', and 'My Views'. Below these are two expandable sections: '构建队列' (Build Queue) showing '队列中没有构建任务' (No build tasks in queue) and '构建执行状态' (Build Execution Status) showing two '空闲' (Idle) items. The main area has a form for 'Item名称' (Item Name) and four radio button options: '构建一个自由风格的软件项目' (Build a free-style software project), '构建一个maven项目' (Build a maven project), 'External Job', and '构建一个多配置项目' (Build a multi-configuration project). Each option has a brief description. The '构建一个多配置项目' option is selected. An 'OK' button is at the bottom right.

## 输入项目名称

输入项目名称，我这里命名为ProjTest。同时我也勾选了“丢弃旧的构建”，这项功能主要是为了节省服务器空间，我设置为“保持构建的最大个数”为30，超过的Jenkins会自动删除旧的构建。



The image shows the Jenkins 'Configure' page for a project named 'ProjTest'. The left sidebar has links: '返回面板' (Return to dashboard), '状态' (Status), '修改记录' (Edit history), '工作空间' (Workspace), '立即构建' (Build now), '删除 Multi-configuration project', and '配置' (Configure). Below these is a 'Build History' section with '构建历史' (Build history) and 'RSS 全部' / 'RSS 失败' links. The main area has a form for '项目名称' (Project Name) with 'ProjTest' entered, a '描述' (Description) text area, and a '[Escaped HTML] 预览' (Preview) button. Below the description is a checked checkbox for '丢弃旧的构建' (Discard old builds) and a '策略' (Strategy) dropdown set to 'Log Rotation'. There are two numeric input fields: '保持构建的天数' (Keep builds for days) and '保持构建的最大个数' (Keep maximum number of builds), with the latter set to '30'. Below these are three checkboxes: '启用项目安全' (Enable project security), '参数化构建过程' (Parameterize build process), and '关闭构建 (重新开启构建前不允许进行新的构建)' (Disable build (no new builds allowed until build is restarted)). A '高级...' (Advanced...) button is at the bottom right.

## 与代码版本管理工具集成

Jenkins支持常用的代码版本管理软件。通过此项设置，Jenkins会自动从代码版本管理的服务器下载源码，并自动构建。以下以SVN为例进行说明。

在源码管理中选择“Subversion”，在“Repository URL”中输入版本库地址，如果你输入的地址有误，会提示“Repository URL is required.”，如下图所示。

源码管理

☐ None  
☐ CVS  
☐ CVS Projectset  
☐ Git  
☒ Subversion

Modules

Repository URL:  ?

**Repository URL is required.**

Local module directory (optional):  ?

Repository depth:  ?

Ignore externals: ☐ ?

[Add more locations...](#)

Check-out Strategy:  ?

Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

源码库浏览器:  ?

输入正确的版本库地址，结果还是提示有误，如下所示。

源码管理

☐ None  
☐ CVS  
☐ CVS Projectset  
☐ Git  
☒ Subversion

Modules

Repository URL:  ?

**Unable to access svn://172.16.1.4/项目管理/C 行业GPS项目区/两标视频终端项目/A 程序代码: svn: E200015: No credential to try. Authentication failed (show details) (Maybe you need to enter credential?)**

Local module directory (optional):  ?

Repository depth:  ?

Ignore externals: ☐ ?

[Add more locations...](#)

Check-out Strategy:  ?

Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

源码库浏览器:  ?

[高级...](#)

错误提示信息为：

Unable to access svn://172.16.1.4/项目管理/C 行业GPS项目区/两标视频终端项目/A 程序代码: svn: E200015: No credential to try. Authentication failed (show details)  
(Maybe you need to enter credential?)

这是因为Jenkins所在的服务器没有访问过SVN版本库，无法通过身份验证，解决办法就是在Jenkins所在服务器访问一次SVN版本库，如下所示：

```
$ ssh ci@172.16.1.4ci@172.16.1.4's password:Last login: Fri Jul 24 10:13:39 2015[ci@localhost:/home/ci]#[ci@localhost:/home/ci]#svn info "svn://172.16.1.4/项目管理/C 行业GPS项目区/两标视频终端项目/A 程序代码"认证领域: This is YF1's Source SVN"ci"的密码:认证领域: This is YF1's Source SVN用户名: xzp"xpz"的密码:-----注意! 你的密码, 对于认证域: This is YF1's Sou
```

rce SVN只能明文保存在磁盘上！ 如果可能的话，请考虑配置你的系统，让 Subversion可以保存加密后的密码。请参阅文档以获得详细信息。你可以通过在“/home/ci/.subversion/servers”中设置选项“store-plaintext-passwords”为“yes”或“no”，来避免再次出现此警告。-----保存未加密的密码 (yes/no)  
?yes路径：A 程序代码URL：svn://172.16.1.4/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86/C%20%E8%A1%8C%E4%B8%9AGPS%E9%A1%B9%E7%9B%AE%E5%8C%BA/%E4%B8%BA%8F%E4%BB%A3%E7%A0%81版本库根：svn://172.16.1.4/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86版本库 UUID：c69c5511-93e5-fc4f-99f1-8de8aae1ea61版本：7026节点种类：目录最后修改的作者：rf最后修改的版本：7005最后修改的时间：2015-07-16 10:24:45 +0800 (四, 2015-07-16) [ci@localhost:/home/ci]#

## 设置构建触发器

也就是什么时候开始自动构建。这里可以有不同方案，大家一看就知道是什么意思。我这里选择Poll SCM，该方案能定时检代码版本库是否有新的提交，如果有则自动构建。我设置的日程表是“H/5 \* \* \* \*”，即5分钟自动检查一次代码版本库是否有新的提交。

### 构建触发器

☐ 触发远程构建 (例如,使用脚本) ?

☐ Build after other projects are built ?

☐ Build periodically ?

☒ Poll SCM ?

日程表

H/5 \* \* \* \*

?

Would last have run at 2015年7月24日 星期五 上午11时50分46秒 CST; would next run at 2015年7月24日 星期五 上午11时55分46秒 CST.

Ignore post-commit hooks

☐

?

## 配置如何构建

构建方式有很多种，我这里选择shell，因为我要用makefile来编译源码。需要提醒的是这里的SHELL是由Jenkins启动的，Jenkins设置了很多自己的环境变量，可以通过env打印出来研究下，里面很多信息是可以使用的。

## Execute shell

Command

```
# 查看当前路径
pwd

# 查看环境变量
env

# 发布文件所在的目录
archive_dir="."

# 获取svn提交点的版本号
REVISION=$(svn info |grep -E '(Revision:|版本:)' |awk '{print $2}' |head -1)

# 归档版本
jenkins_archive()
{
    # 初始化归档文件名
    file_image=${archive_dir}/r${REVISION}_镜像文件.zip
    file_apps=${archive_dir}/r${REVISION}_应用程序.zip

    # 打包归档文件
    zip ${file_image} ./build/packet_image/*
    zip ${file_apps} ./build/*.exe ./build/*.so

    return 0;
}

# 编译代码
make clean && make all && make install && make packet && jenkins_archive
```

[See the list of available environment variables](#)

删除

增加构建步骤

## 发布构建结果

Jenkins自动构建完后，可以将构建结果存档（如程序版本），以方便大家下载使用，可以通过“构建后操作”的“Archive the artifacts”来实现。

## 构建后操作

## Archive the artifacts

用于存档的文件

./\*.zip

 './\*.zip' doesn't match anything: even '.' doesn't exist

高级...

删除

增加构建后操作步骤

## 其他说明

从下图的构建历史可以看出，蓝色表明构建成功，红色表明构建失败。在测试的过程中，代码版本库都没有新的提价，如何触发构建？可以通过“立即构建”来触发。

-  返回面板
-  状态
-  修改记录
-  工作空间
-  立即构建
-  删除 Project
-  配置
-  Git Polling Log

Build History		构建历史
	#21	2015-9-21 下午5:07
	#20	2015-9-21 下午4:59
	#19	2015-9-21 下午4:32
	#18	2015-9-21 下午4:27
	#17	2015-9-21 下午4:17
	#16	2015-9-21 下午4:16
	#15	2015-9-18 下午4:05
	#14	2015-9-18 下午2:31
	#13	2015-9-18 下午2:30
	#12	2015-9-18 下午2:22

## 授权管理

一切都整好后，发现匿名用户竟然能访问“工作区”，能看到源码。这应该要杜绝。好在Jenkins有丰富的“项目矩阵授权策略”能达到我们要的效果。

用管理员登陆Jenkins，进入“系统管理 > Configure Global Security”，这里有个“授权策略”，选择“项目矩阵授权策略”，在这里就能限制匿名用户的权限了。我对匿名用户的限制是：不允许访问工作区，不允许访问源码，可以下载构建后的归档文件。

## 总结

本次介绍Jenkins只是个入门级别的，想要深入的修行，还是得靠个人。对本次使用Jenkins做个简要总结：

1. 安装简单，Web可视化管理界面，容易上手。
2. 能与代码版本管理软件结合，自动检测是否有新提交，并自动构建。
3. 每次构建结果能自动归档，以便回溯，或供他人下载。