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<u>Linux</u>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<u>https://bitnami.com/stack/jenkins</u>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inux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/ctlscript.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 □ 表录
	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:



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。



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会自动删除日的构建。



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 O None O CVS O CVS Projectset O Git Subversion Modules Repository URL svn://192.16.1.4/通用版 0 Repository URL is required. Local module directory (optional) Repository depth infinity V Ignore externals Add more locations.. Check-out Strategy Use 'svn update' as much as possible * Use 'syn 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 (自动) 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 O None O CVS O CVS Projectset O Git Subversion Modules Repository URL svn://172.16.1.4/程序代码 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) 0 Repository depth infinity Ignore externals 1 (2) Add more locations. Check-out Strategy * Use 'svn update' as much as possible Use 'swn 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 (自动) v 0

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:

高级...

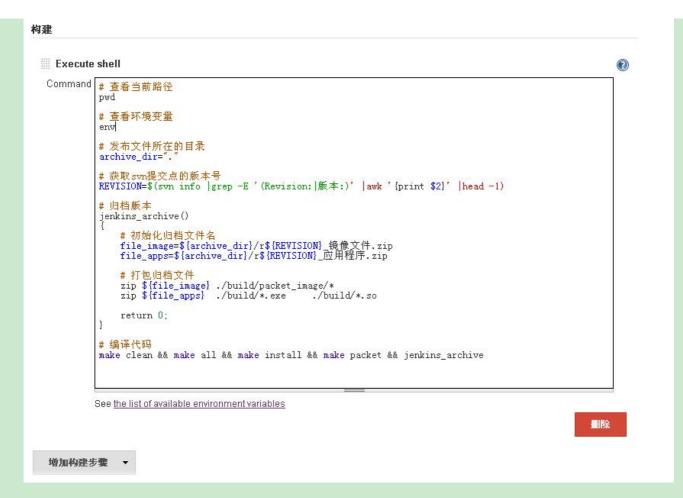
设置构建触发器

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。

Build after other projects are built Build periodically Poll SCM	构建触发器		
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.	■ 触发远程构建 (例如,使)	用脚本)	•
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.	☐ Build after other project	s are built	0
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.	Build periodically		•
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.	Poll SCM		0
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.	日程表	H/5 ****	
2015年7月24日 星期五 上午11时55分46秒 CST.			0
			1
			?

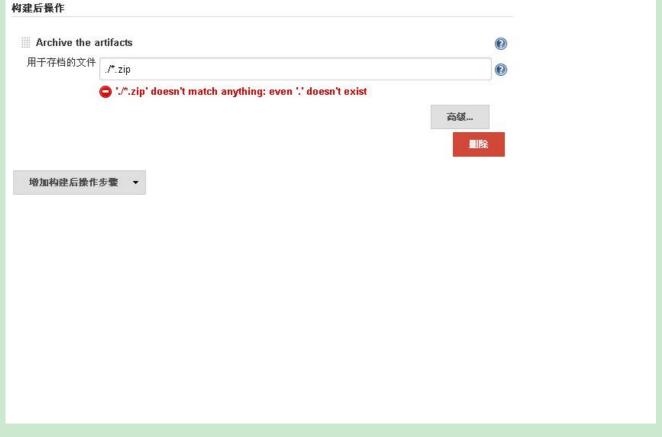
配置如何构建

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。



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。



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。



a Bui	ld History	构建历史 —
<u>#21</u>	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。