

jenkins 实践

作者: [washmore](#)

原文链接: <https://ld246.com/article/1492511359545>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>jenkins 作为一个持续集成(CI)工具,为程序员简化各种繁琐构建,关注编码本身做出极大贡献,本帖前人大神的基础上狗尾续貂,博大家一笑!</p>

<p>准备工作:4 核 2G 云主机一台,部署有 tomcat7.0+maven3.2.2+JDK1.7+git2.6.4;</p>

<p>任务目标:码农只需要老老实实拉代码,写代码,提交代码就好了!!! 程序员只需要在下班前 commit and push 一次自己今天的劳动成果到 git 仓库的指定分支,接下来就可以坐等 jenkins 为您 build and deploy;接下来看我是如何给大家躺雷的!整个过程请结合之前大神发的图示自行脑补!

第一步:前往 https://ld246.com/forward?goto=http%3A%2F%2Fjenkins-ci.org/ 下载 jenkins 工具,我下载的是 war 式的,需要有 tomcat 等容器支撑运行;

war 包下载完成后丢进 tomcat/webapps 目录下,启动容器;

辣么第一个坑来了,之前 我用的是 tomcat7 是基于 JDK1.6 的 在 tomcat 启动过程中 直接报错,看日志输出 提示 jenkins 版本太高,要么升级 jdk,要么降级 jenkins,一开始对于升级 jdk,我是拒绝的,你不让我升我就升级,我还有辣么多小玩意是基于 jdk1.6 开发的,玩意升级 1.7 后跑不了了怎么办?所以我能将 jenkins 逐步往下降低版本,经过一番尝试后发现 1.580 可以在 jdk1.6 下运行,搞起!这次 tomcat 启动了,然而,访问 http://localhost:8080/jenkins 进入主页就有各种黄色感叹号,提示你这个插件不能用那个也不能用,提示你 jenkins 有新的版本啦!!!最关键的是 github 插件用不了了,这还怎么玩??只能忍升级 jdk 咯,一番折腾略过不提,知道使用 jdk1.7 成功访问 1.643 的 jenkins 主页!http://localhost:8000/jenkins</p>

<p>第二步:进入 http://localhost:8080/jenkins 后选择系统管理界面,在选择管理插件,版本控制插件,默认是没有 github 的,选择下载 github 插件,就会自动给你吧依赖的 git gitserver 等有依赖关系的件一股脑下载好,如果下载失败的话,就自己去网站上下载然后上传本地插件文件安装!Deploy to container Plugin 插件用于发布 war 包到指定 tomcat 容器,实际上也是使用的 tomcat7-maven-plugin 插执行了 tomcat7:redploy 命令,如果你需要发布到 tomcat 集群的话,才需要此插件,因为可以添加多个布容器.</p>

<p>第三步:进入 http://localhost:8080/jenkins 后选择系统管理界面,再选择全局系统设置,JDK,MAVEN,GIT 一路设置下去,有不懂的地方就点旁边的小问号,挺管用的!

配置到 Jenkins Location 节点的时候,又一个坑来了,Jenkins URL 请配置当前的访问路径,如果你有多域名指向一个 Ip,那么除了你在此处配置的 URL 外,用其他的域名访问 jenkins 的时候,就会有很醒目提示:跨域风险!直接点掉忽略就行,当时在这个地方纠结了好久,最后在万能的 https://ld246.com/forward?goto=http%3A%2F%2Fstackoverflow.com/ 上得到了解答.Jenkins Location 节点还有另一个需要填写的地也是个坑,说多了都是泪,那就是 系统管理员邮件地址,如果你需要用到 build 成功/失败给指定邮箱送邮件的话!此处就一定要认真填写,要不然后面发送邮件就会报 502,503 等错误,说发送用户邮箱地址用于接收的地址不符什么的!

继续配置 GitHub Plugin Configuration 节点,此处需要用到 github 的 push 钩子,不懂的话请自行百度,大概意思就是当你向指定的仓库执行特定操作比如 push pull merge 等操作的时候,向指定的 URL 送一段此操作的相关

数据包,Credentials 即为凭证,请自行添加一个能正常使用 hook 的票据(一般是 secret text 类型的)应用,Manage hooks 不必勾选,勾选也没关系,意思就是如果你在 jenkins 的 github 身份信息有仓库的 admin 全县的话,就会自动

给你的 github 网站上维护一个钩子上去,如果没有勾选的话 ,就自行在 github 上添加 webhook,URL 是 http://localhost:8080/jenkins/github-webhook/ 注意替换成自己的域名

然后是邮件通知节点,按照你所知道的去配置就好了,毕竟你是用过 foxmail 的人!注意发件邮箱用上面的那个管理员邮件地址就好了!</p>

<p>第四步:进入 http://localhost:8080/jenkins 后选择系统管理界面,再选择 Configure Global Security 界面进行安全性设置,毕竟你也不想谁都能到你这来点点点建立一大堆没用的 project 和 war/jar 吧 一般来讲授权策略选择

安全矩阵就好了,新建一个帐号,勾选上所有的权限,然后把匿名帐号的权限全干掉,注意操作的先后顺序,不然先把匿名用户的权限干掉了然后保存了,再想去新建账户 就又是坑了

第五步:至此,jenkins 的基本就能用了,开始新建一个工程吧!进入新建界面

Item 名称即为工程名,由于我们公司使用的都是 maven 构建工程,so 此处选择构件一个 maven 项目! K 进入下一步,一般没有提到的地方按默认的来就好

源码管理 节点选择 Git,Repository URL 填入仓库地址,Credentials 填写 github 帐号密码添加一个凭证,注意请使用之前配置过 webhook 的帐号.这样才能触发接下来要配置的 Build

hen a change is pushed to GitHub 选项;

Branches to build 为分支号,请自己人填入,

构建触发器节点即什么时候该触发 jenkins 的构建动作,一般勾选 Build whenever a SNAPSHOT dependency is built 和 Build when a change is pushed to GitHub 即可,当然,勾什么看大家的意思自行选择,都是可以的

注意区分 Build periodically 和 Poll SCM 的异同,不懂请自行百度.

Build 节点中 Root POM 填写你要构建工程的顶级父目录中的 pom.xml 文件,Goals and options 填写你希望执行的 maven 命令,我填写的是 clean install;为什么没有其他发布需要的命令 稍后有说;

构建设置节点 E-mail Notification 填写需要邮件通知到的用户列表,用,分割,三选项啥意思自行脑补,里面提到的 module 模块稍后有说;

至此需要执行 jenkins 自动构建功能的配置就完了,此时,你可以提交代码到 jenkins 工程配置的仓库对的分支尝试触发自动构建,或者手动点击立即构建按钮,开始构建!</p>

<p>第六步:假如你一次到位构建成功了,那么恭喜你,github 仓库的代码已经被 build 成对应的 jar 和 war 躺在 jenkins 的目录下了!然而光这样还不够,我还希望 jenkins 能够帮我将一个 maven 父工程中个子模块自动处理,所有 jar 上传到公司 nexus 私服中,所有 war 发布到指定的 tomcat 中,继续往下看

再次选择刚刚新建的工程进入配置界面这时候开始添加构建后操作节点,因为在一次成功构建之前,是没办法执行子模块操作的;

首先添加 Deploy war/ear to a container 操作,如果没有安装 Deploy to container Plugin 的话是没有这一项的,没关系跳过此步好了!WAR/EAR files 填写* /.war 通所有 web 工程子模块,Containers 自己选择容器类型

并填写相应的容器管理信息,如果需要发布到多个容器,继续添加容器即可!

然后添加 Build other projects 操作,意思是父工程 build 成功后会继续执行其的 build,此时要构建的项目就可以填写 jenkins 工程的子模块了一般命名规则是 jenkins 工程名/mavenGroupId:mavenartifactId 的格式,多个子模块之间用,分割,如果之前有进行 Deploy war/ear to a container 操作,则此处不要填写 web 工程模块,只写 jar 即可,

此时选择模块选项进入模块列表,对于 Build other projects 操作列表内的模块,依此点进去填写在配选项里面的 Build 节点中的 Goals,jar 填写 deploy,war 填写 tomcat7:redeploy(注意此处是 tomcat7 maven-plugin 的写法,其他容器 maven 插件自行百度),完成配置后再次尝试手动点击立即构建按钮,始构建!</p>

<p>至此完整的一次流程就走完了,只要有用户提交代码到 github 仓库中,那么 nexus 中的 jar 快照有了,tomcat 下的 war 工程也可以访问了,真是轻松加愉快啊!</p>

<p>如果各位在使用 jenkins 的过程中,有任何疑问或者自己的见解,欢迎探讨!</p>