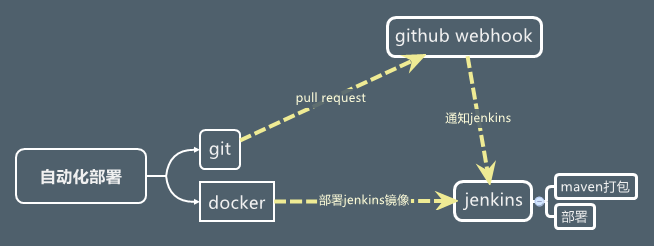
### 转自：http://m.blog.csdn.net/ggjlvzjy/article/details/51151591

### 写在前面

以前搭建spring-cloud的时候 看到别人代码里面总有Dockerfile，然后了解到docker容器的概念，玩了两个星期玩不下去了，就放下了，无意间在网上看到一篇文章Docker持续部署，就想着自己也搭建一套玩玩。

### 流程

网上那篇文章介绍的流程涉及到多个docker镜像，我这里为了入门就简单化了。



自动化部署效果就是： 你修改了本地的代码 然后push到git上面，git的webhook功能会给jenkins post一个请求，jenkins接收到请求后，就进行打包部署操作，细节下面会说。

#### docker部署jenkins

docker 怎么安装使用可以参考以前博客。   
docker启动：

sudo route del -net 172.16.0.0 netmask 255.240.0.0

sudo service docker start

镜像地址：<https://hub.docker.com/jenkins/>   
jenkins docker官网有提供image的 直接pull下来就可以了。   
jenkisn安装步骤：   
拉取镜像

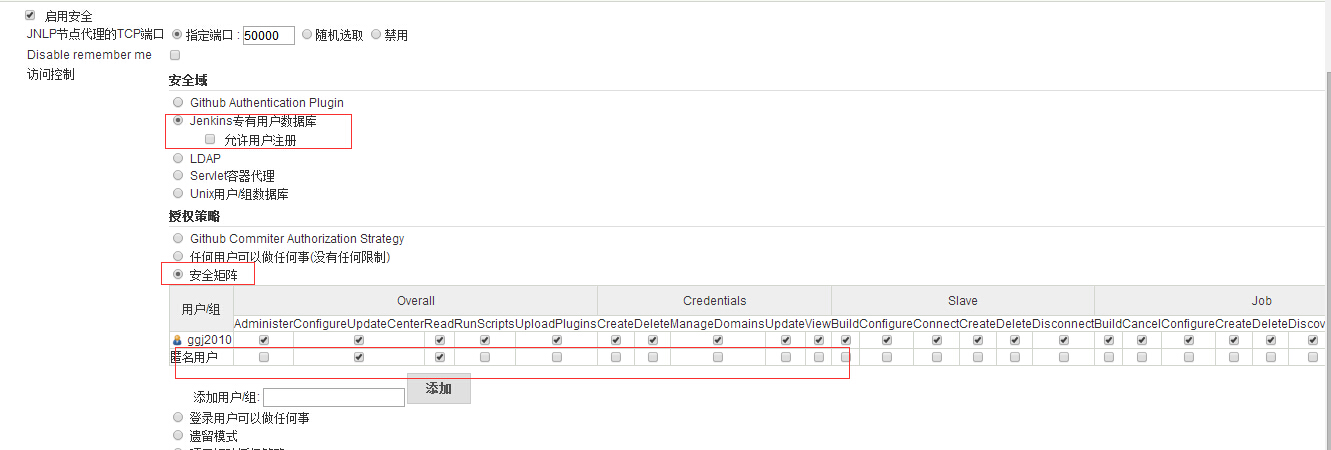
docker pull jenkins:latest

这种方式启动会存储数据，-t 标示以后台模式运行，-p标示容器和宿主服务器之间的开放端口号 -v表示需要将本地哪个目录挂载到容器中，格式：-v <宿主机目录>:<容器目录>   
运行容器

docker run -t -p 8080:8080 -p 8083:8083 -p 50000:50000 -v /var/jenkins\_home:/var/jenkins\_home jenkins

运行成功之后，我们输入<http://ip:8080/> 就可以访问到了

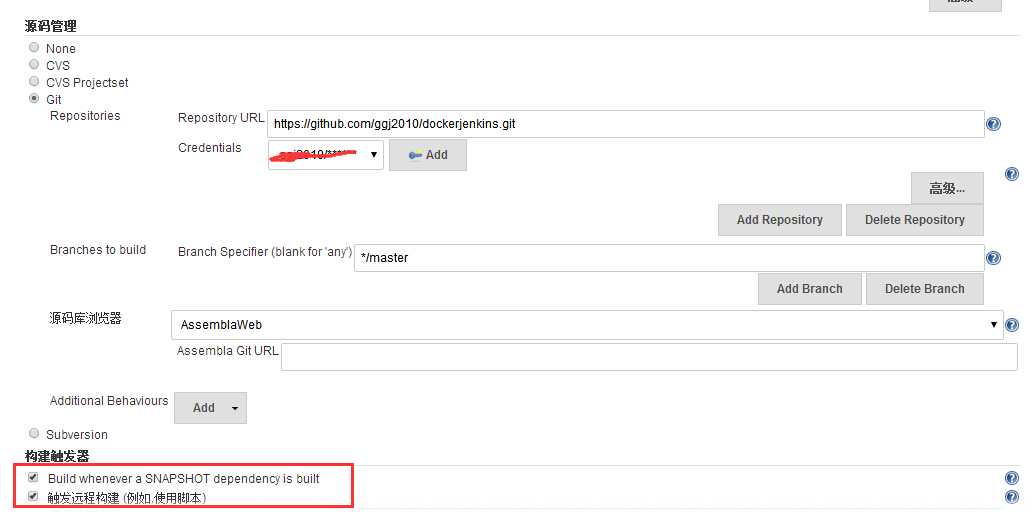
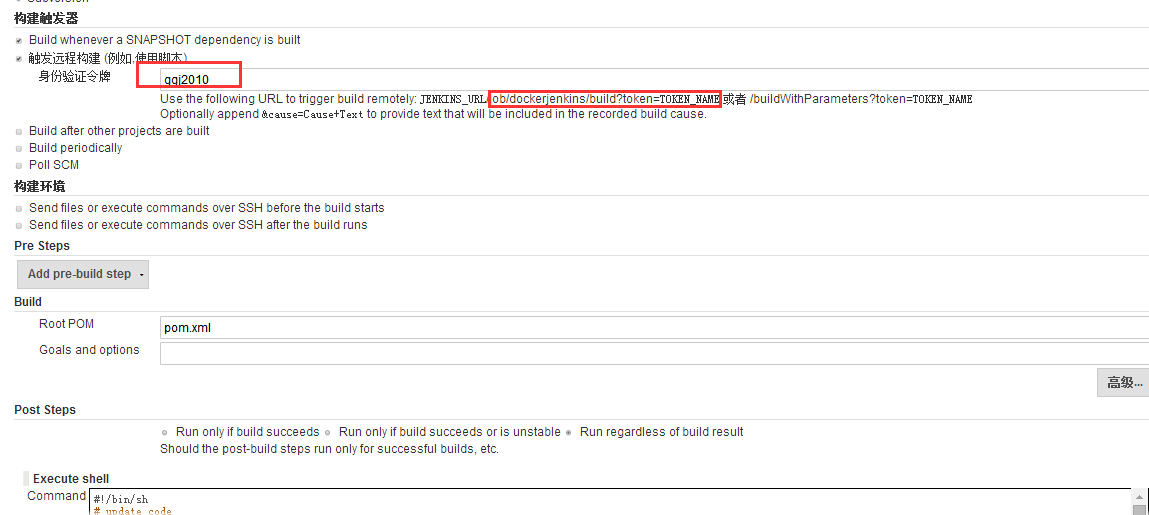
#### jenkins的注意事项

权限问题，具体细节可以百度，我们可以先开发用户注册功能，然后注册一个用户，然后给注册的用户开放所有的权限，匿名的用户权限勾掉。需要注意一点的就是 git的webhok功能也需要权限，所以我们配置安全矩阵的时候要开发匿名用户的访问job的权限，不然git的webhok功能无法使用   


系统配置   
在全局属性里面新增：BUILD\_ID=DONTKILLME 这个键值对非常重要，如果不配置 无论你怎么运行java -jar *\**.jar & 项目都无法后台运行，因为一旦jenkins 部署成功执行完shell 就会把其他的进程都干掉。   
maven jdk都要配置下

插件安装   
git/Publish over SSH 都要按照然后配置

项目配置

1. shell 主要实现kill 原有的java进程和运行新的jar

projectName="dockerjenkins.jar"#kill

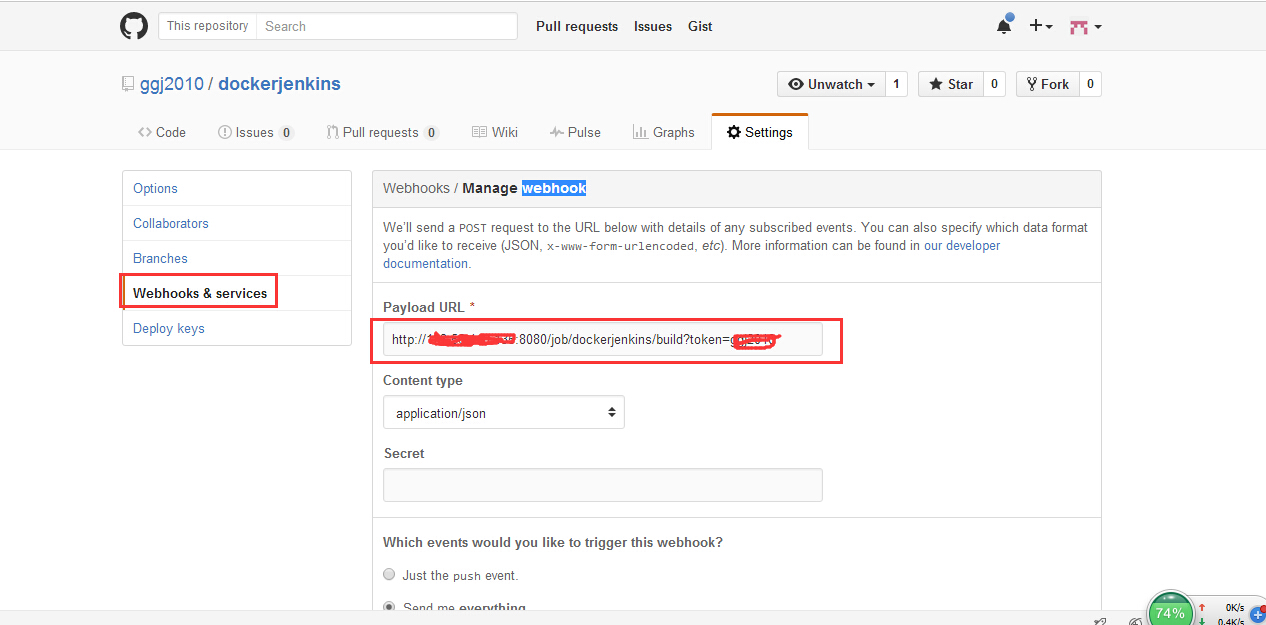
ps -ef | grep -v 'grep' | grep 'dockerjenkins.jar' | awk '{print $2}'| xargs kill -9

ps -ef|grep java

echo "start ${projectName}"

java -jar $WORKSPACE/target/${projectName} &

#### git webhoks

git项目地址：<https://github.com/ggj2010/dockerjenkins.git>   
webhoks配置：   
这个url 和jenkins配置项目的地址一致 可以看前面截图触发远程构建   


参考网址：<http://www.thinksaas.cn/group/topic/378471/>