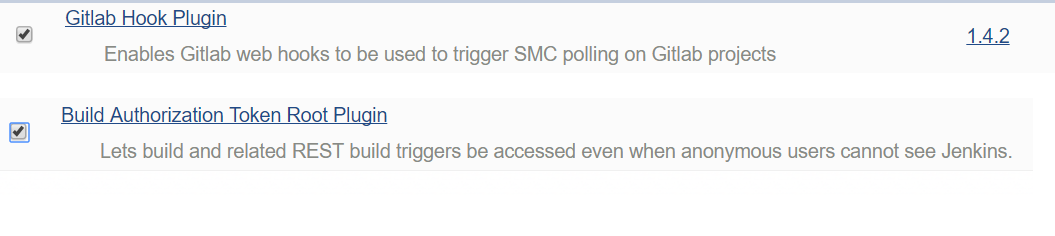
一：目的为在公司的测试环境当中一旦开发向gitlab仓库提交成功代码，gitlab通知jenkins进行构建项目、代码质量测试然后部署至测试环境，注意这只是测试环境，而生产环境依然需要手动部署代码：

1.1：jenkins配置：  
1.1.1:安装Gitlab Hook Plugin插件：  
#系统管理-管理插件-可选插件-Gitlab Hook Plugin和Build Authorization Token Root Plugin

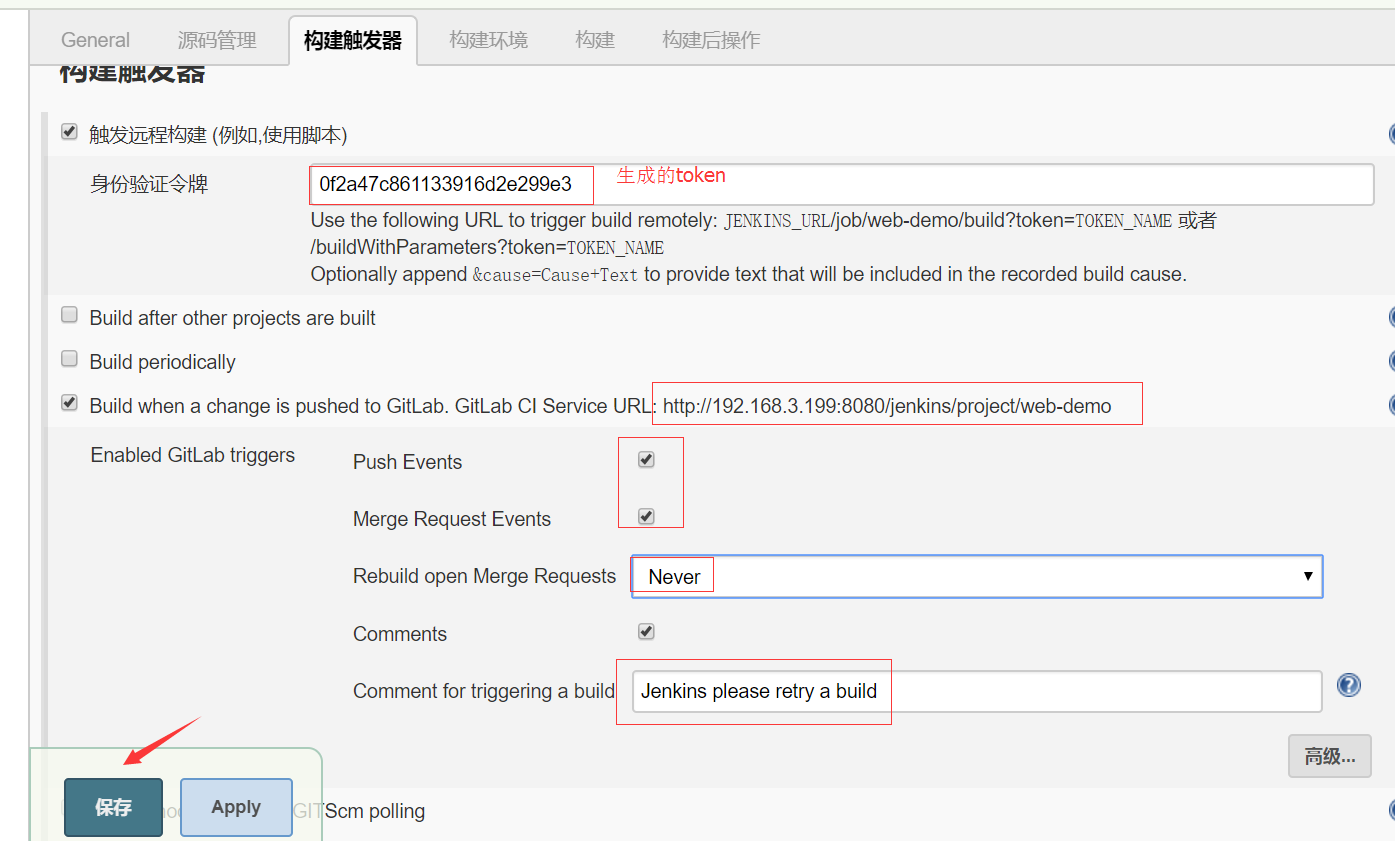


1.1.2：生成随机token：

# openssl rand -hex 12

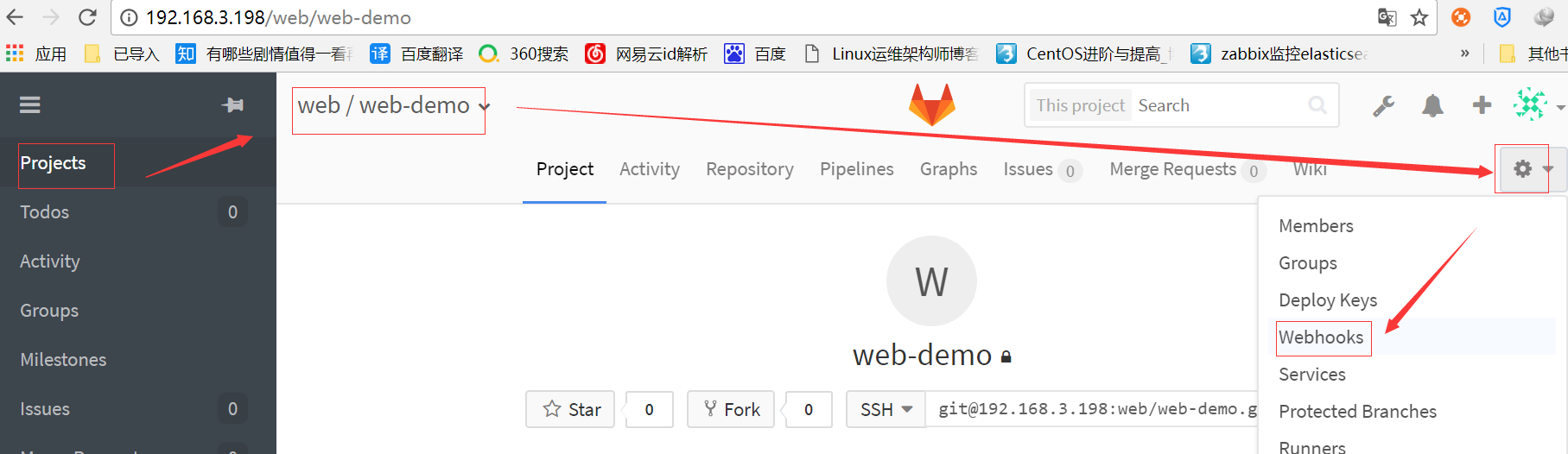
0f2a47c861133916d2e299e3

1.1.3:创建项目触发器：  
#项目-配置-构建触发器：  
http://192.168.3.199:8080/jenkins/project/web-demo

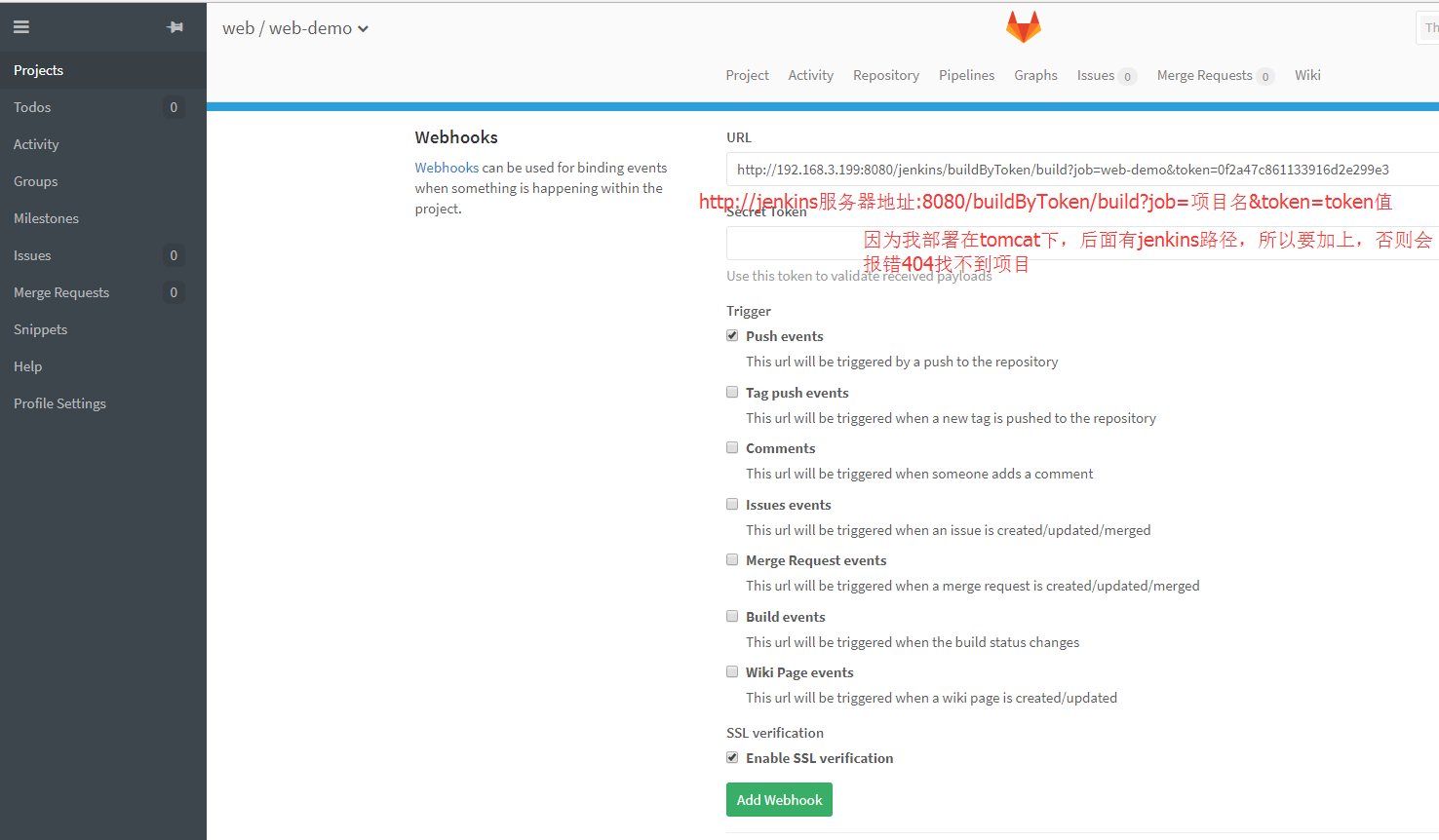


1.2：配置github：  
1.2.1：在git项目配置界面设置链接和token：

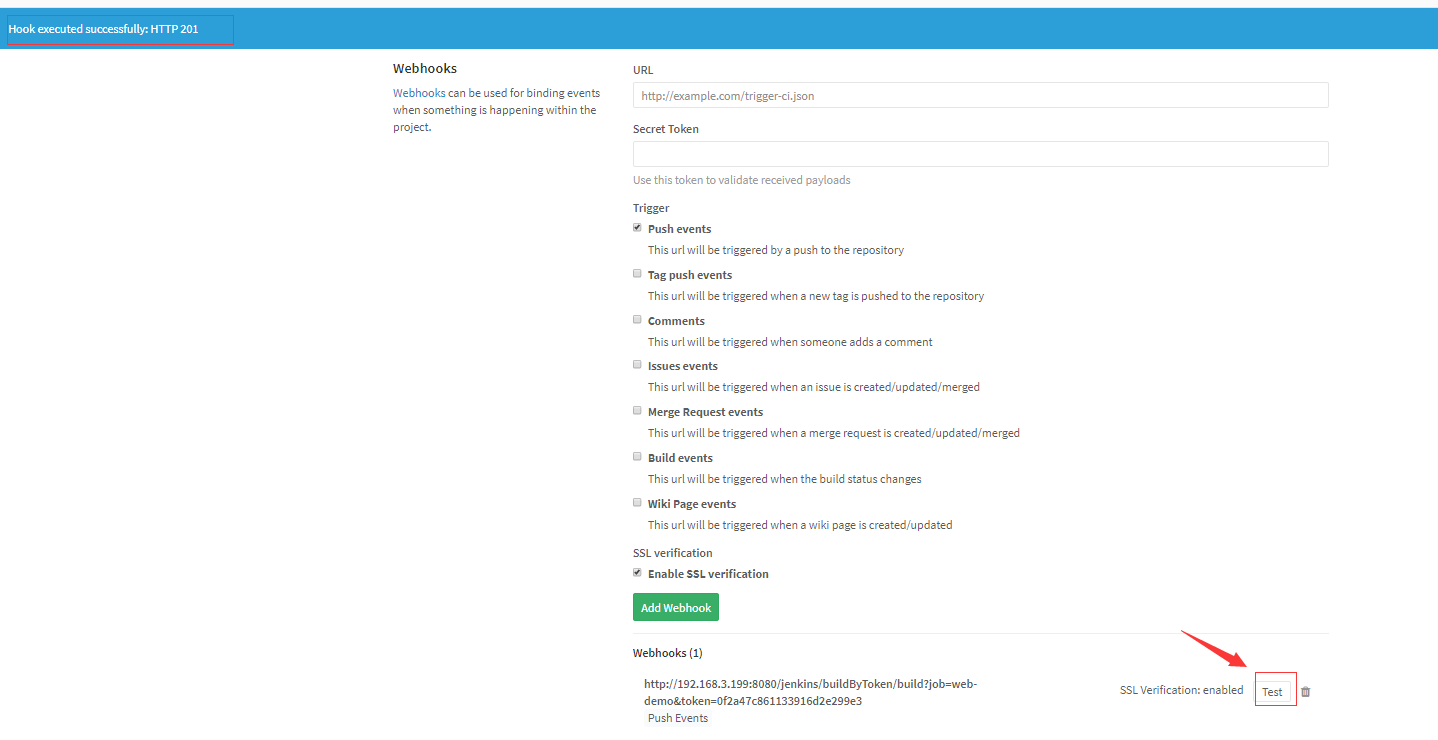
登录gitlab,在这个项目下找到钩子配置的地方

  
#选择项目-设置-webhooks：  
#插件使用介绍，https://wiki.jenkins-ci.org/display/JENKINS/Build+Token+Root+Plugin

http://192.168.3.199:8080/jenkins/buildByToken/build?job=web-demo&token=0f2a47c861133916d2e299e3  
http://jenkins服务器地址:8080/buildByToken/build?job=项目名&token=token值



1.2.2：测试：



报错：  
hook executed successfully but returned http 404

本次是因为没有在项目后面加上jenkins这个路径

1.2.2：测试，看到显示201表示成功

1.3：向git服务器提交代码，验证是否可以自动部署：  
1.3.1：提交代码：

IMG_261

[www@master code]$ git clone git@192.168.3.198:web/web-demo.git

[www@master web-demo]$ echo "Build token root plugin" > index.html

[www@master web-demo]$ git add 'index.html'

[www@master web-demo]$ git commit -m 'build token root plugin test'

[master beb37cb] build token root plugin test1 file changed, 1 insertion(+), 1 deletion(-)

[www@master web-demo]$ git push origin master

Counting objects: 5, done.

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 281 bytes | 0 bytes/s, done.

Total 3 (delta 1), reused 0 (delta 0)

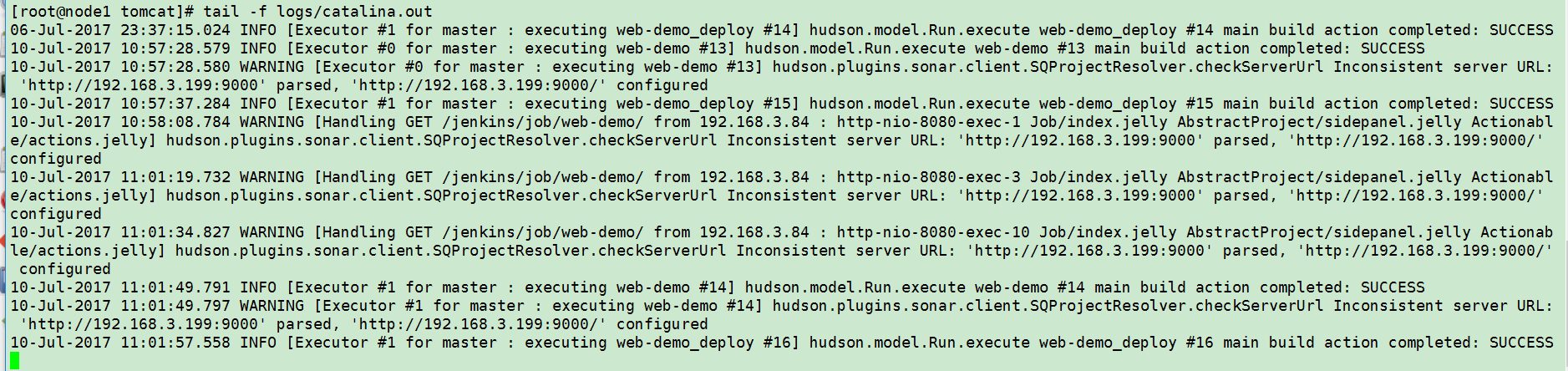
To git@192.168.3.198:web/web-demo.git

c02523b..beb37cb master -> master

IMG_262

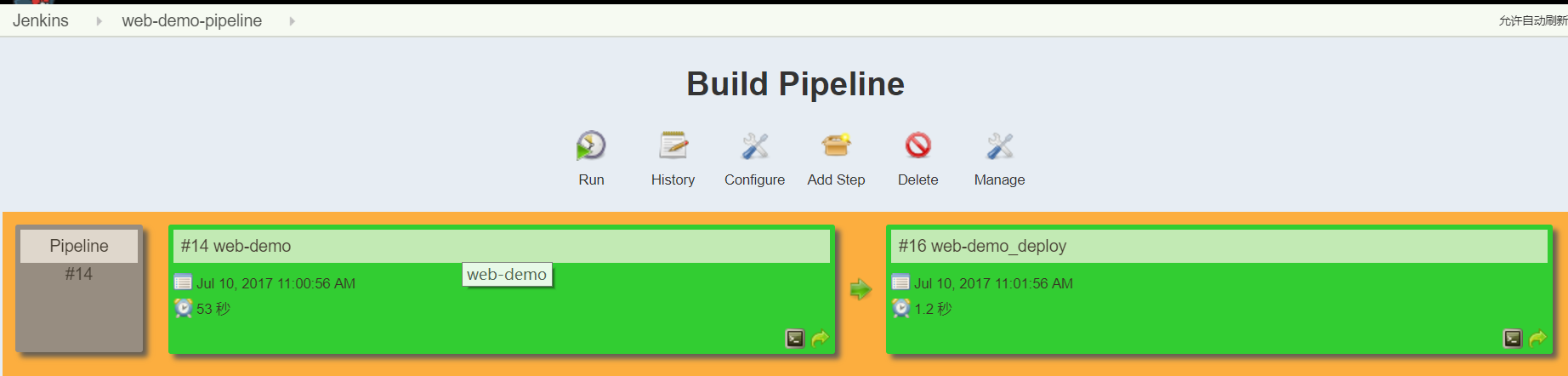
1.3.2：jenkins服务器的日志记录：

[root@node1 tomcat]# tail -f /usr/local/tomcat/logs/catalina.out

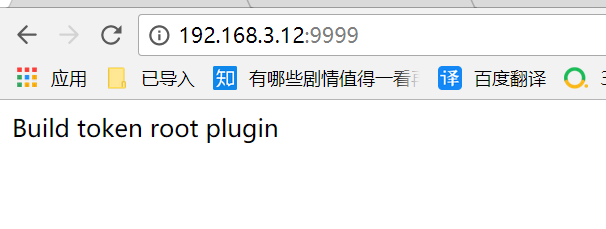


1.3.3：jenkins项目构建：





1.3.4：访问web界面验证代码是否最新的：



1.3.5：jenkins控制台输出信息：

