教程(六)多分支pipeline

什么是多分支pipeline

在jenkins中存在两种类型的pipeline, 之前我们讲述的都是普通类型的pipeline。而多分支pipeline作为一个非常重要的类型重要为我们完成持续集成的第一个步骤—打通gitlab的 通信。 这样研发再push代码后通知jenkins运行我们预先定义的pipeline完成整个持续集成流程。 要做到这样的效果需要分别在jenkins和gitlab中做如下准备工作。

jenkins的配置

配置jenkins (这里已经在咱们的jenkins中配置好,凭据名称为gaofeigitlab账号)

- 1. 安装gitlab插件
- 2. 在安全设置中添加jenkins 凭据
- 3. 类型选择:Gitlab API Token (获取方式:在gitlab中使用自己的账户登录,在User settings中找到Access Tokens。 在这里创建一个token)
- 4. 复制这个token保存到上面说的jenkins 凭据中。

创建多分支pipeline

在jenkins中创建一个job,选择类型为多分支pipeline。在git中填写研发的repo地址,jenkins 凭据以及要监控的分支。如下:



gitlab的配置

到研发的repo中,添加一个跟jenkins通信的webhook。 需要进入settings->integration→添加webhook。 中间要填写jenkins job的url 以及 勾选push event和merge event。如下

🦊 GitLab 🛛 Projects 🗸	Groups - Activity Milestones Snippets	🕒 🕒 🗸 Search or jump to Q D) 🐧 😋 30 Q 🗸 🌐 V
D) Issues 0	Integrations Webhooks can be used for binding events when something is	URL http://auto.4paradigm.com/project/sage-sdk-pipeline
Merge Requests CI / CD	happening within the project.	Secret Token
G Operations		Use this token to validate received payloads. It will be sent with the request in the X-Gitlab-Token HTTP header. Trigger
Registry	General Members	Push events This URL will be triggered by a push to the repository
Packages	Integrations	master
Snippets	Repository	This URL will be triggered when a new tag is pushed to the repository
🏟 Settings	Audit Events	Comments This URL will be triggered when someone adds a comment
✓ Collapse sidebar		Confidential Comments This URL will be triggered when someone adds a comment on a confidential issue
attne://aitlah And in/DHT3/eane-n	vthon_edk/eattinge/integratione	

注意: jenkins job 的url的格式是: http://JENKINS_URL/project/PROJECT_NAME

通过上面的配置,我们就打通了jenkins 与 gitlab的通信。一旦有研发在提交代码和提交merge的时候就会触发这个多分支pipeline运行。

准备jenkinsfile 多分支pipeline的规则是打通了研发repo中所有分支的事件。可以理解为它监控了repo中的所有分支的代码变动。 所以它不准在job的脚本框中编写pipeline,我们需要在研发的分支中添加jenkinsfile来保存我们的pipeline。 注意:多分支pipeline在创建后就会扫描研发repo中所有的分支并寻找jenkinsfile文件。 所以我们要在所有需要执行pipeline的分支中都要编写一份pipeline。如果jenkins找不到jenkinsfile边不会监控此分支。 jenkinsfile中的pipeline脚本与普通的pipeline语法一致,没有区别。 .15. Contractive C White the way of the contraction of the contraction