## github+Jenkins+codedeploy:

<https://github.com/aws-samples/aws-serverless-workshop-greater-china-region/tree/master/Lab8B-CICD-Jenkins>

步骤：

1.Jenkins server的搭建：在EC2上 (piaozone/Password1)

1）安装aws cli；  
2）安装jenkins，安装推荐的插件（包含github plugin）

3）对Jenkins server运行对应的IAM role进行配置  
因为Jenkins server运行是使用 os层面的jenkins用户，需要保证jenkins用户对应的iam权限。  
如果使用ec2 attache 的role，注意查询.aws目录下的credential文件没有配置。  
如果需要使用iam user，就对jenkins配置aws configure：su到jenkins，su -s /bin/bash jenkins （jenkins为service，无法直接su到该用户）或者使用cd ~jenkins sudo su -jenkins aws configure

2. 为github中使用的repository添加对jenkins的webhook，这样一旦代码被push到github，就会触发jenkins中使用该repository的项目  
1）github-》右上角user-〉setting-》developer setting-〉personal access token-》generate new token （获得代表github用户的access token）

2）repository -》setting-〉webhooks-》payload URL: <http://jenkins> server ip:8080/github-webhook/ 该repository变化时即触发jenkins

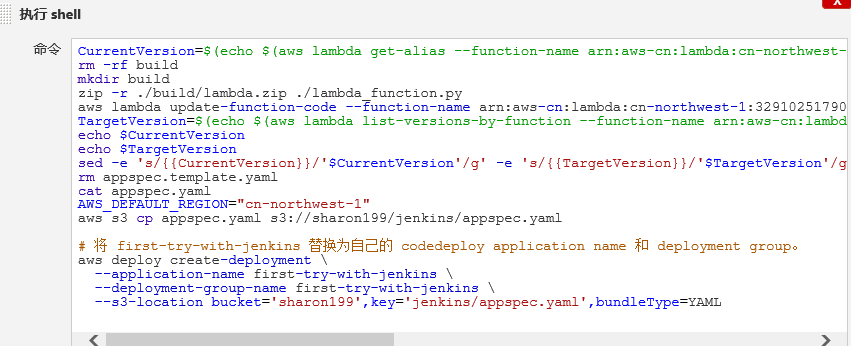
3.Jenkins的配置  
1）系统配置：  
manage Jenkins-》configure system-〉配置jenkins url：<http://jenkins> server:8080/   
配置Github server的credential（选择secret text, secret输入该表github user的access token）

场景一：github上新后，触发jenkins中的项目，执行free style中shell脚本，完成所有的构建任务（该场景使用了lambda zip打包后，上传S3，使用codedeploy部署）

4.Jenkins中创建项目

1）source code management：选择git，配置repository的url（如<https://github.com/1559550282/cicd.git）配置credential>：使用github的user/password  
选择branches to build: \*/main --即repos里面获取代码的分支名称

2）触发器：github hook trigger for GITScm polling -- github上新后通过hook触发该项目执行  
3）构建：执行shell脚本



5.配置codedeploy  
1）create codedeploy使用的role

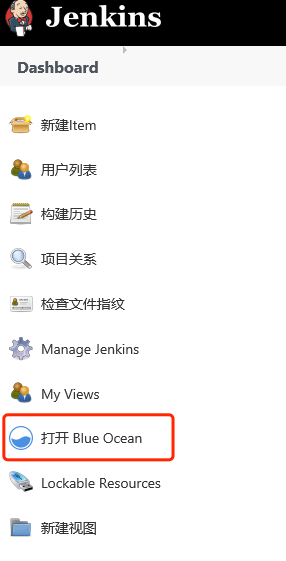
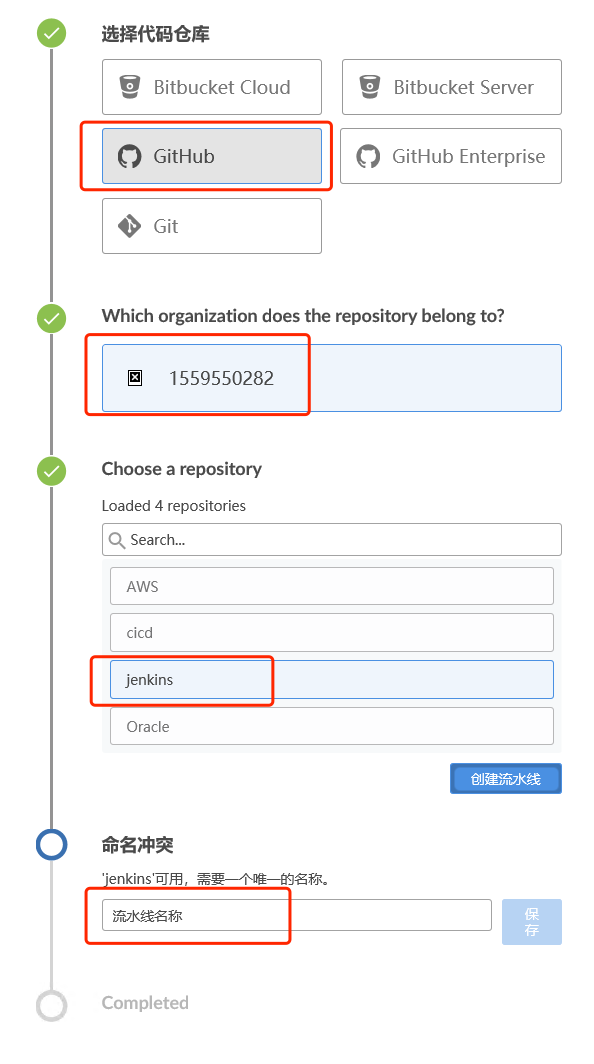
2）create 新application（选计算平台lambda）对应上面用的application\_name

3）对该application 创建部署组（选择role &部署的方式-如allatonce）对应上面的deployment-group

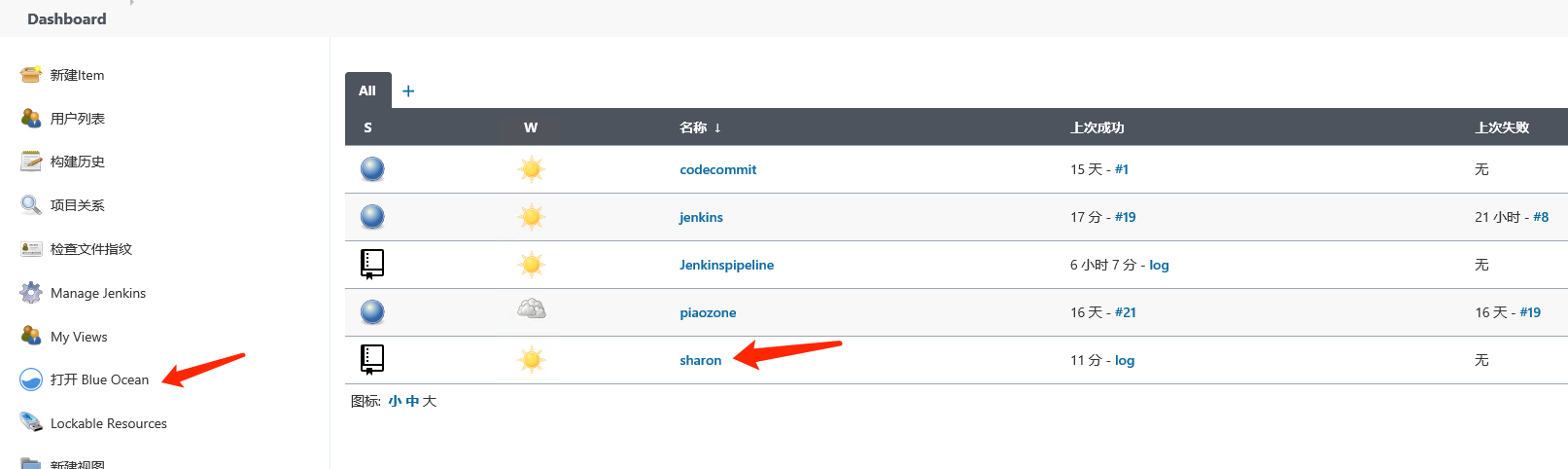
场景一：使用Github + Jenkins pipleline + codebuild + codedeploy + sam部署lambda

前面1、2、3步骤同样配置

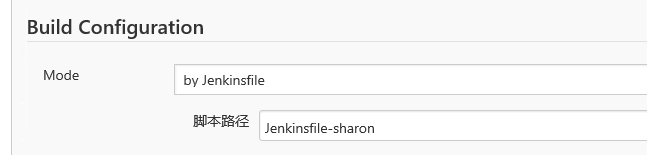
4.为jenkins安装codebuild，codedeploy，blue ocean插件

5.使用blue ocean创建名为sharon的pipeline  
 

创建后：



创建后，进入config，可以修改该pipeline对应的repo中的脚本名称：

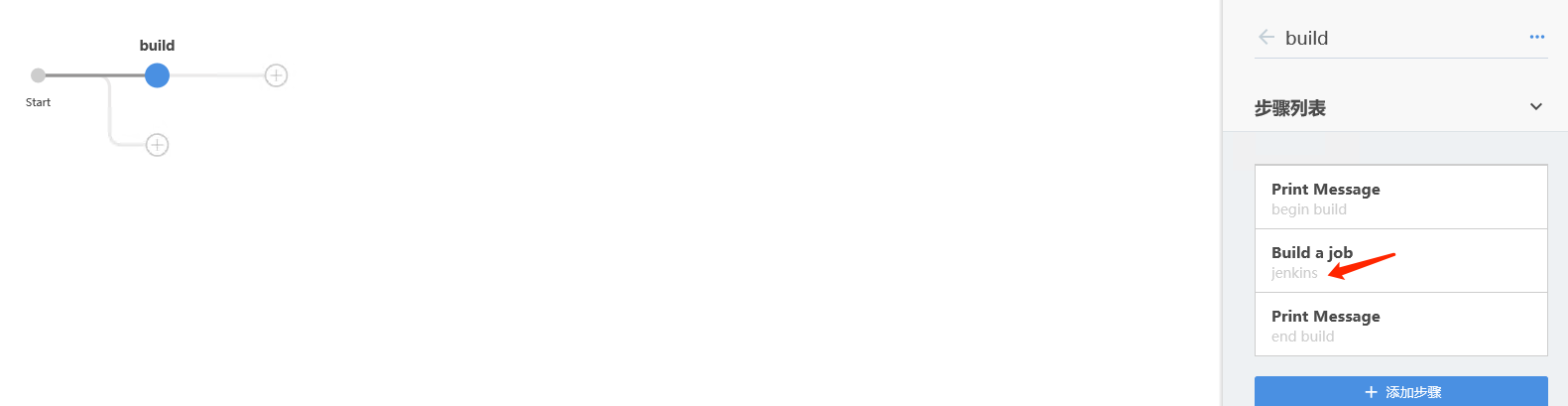


进入到pipeline，再通过blue ocean进入到pipeline的定义



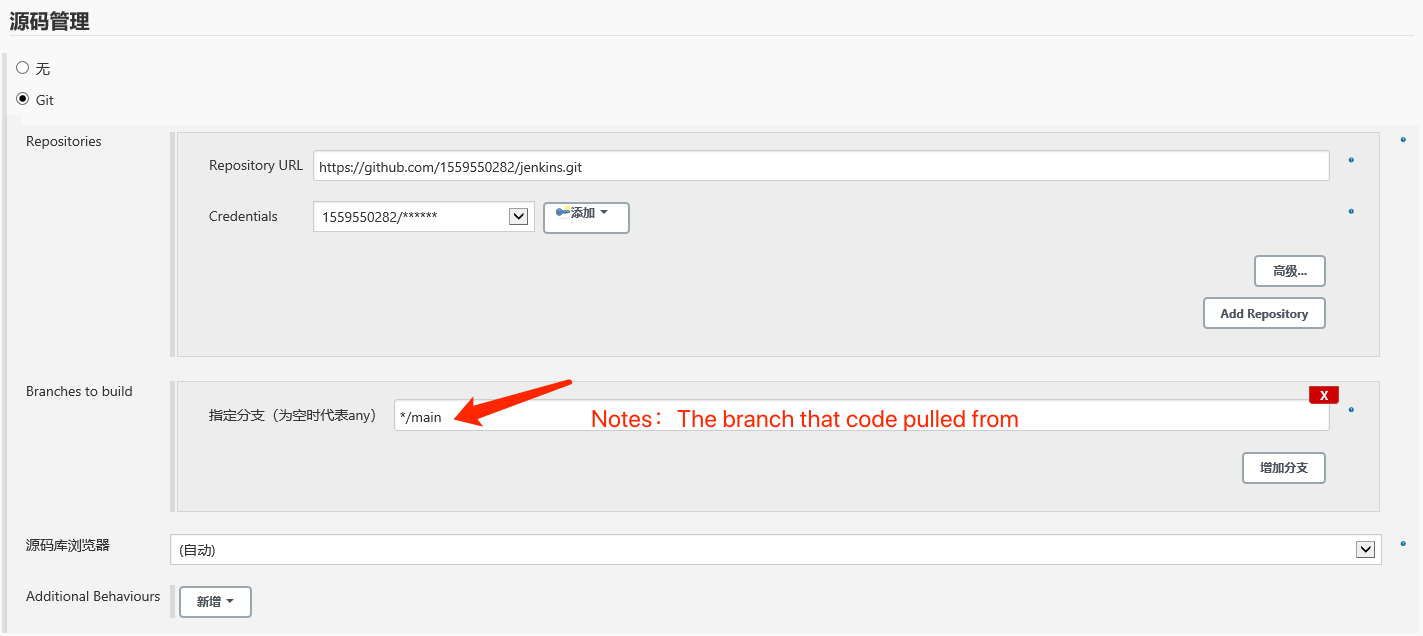
定义pipeline：



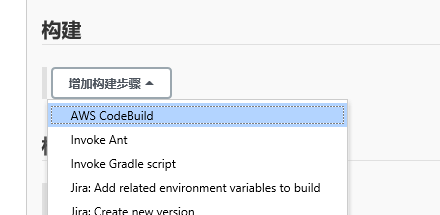


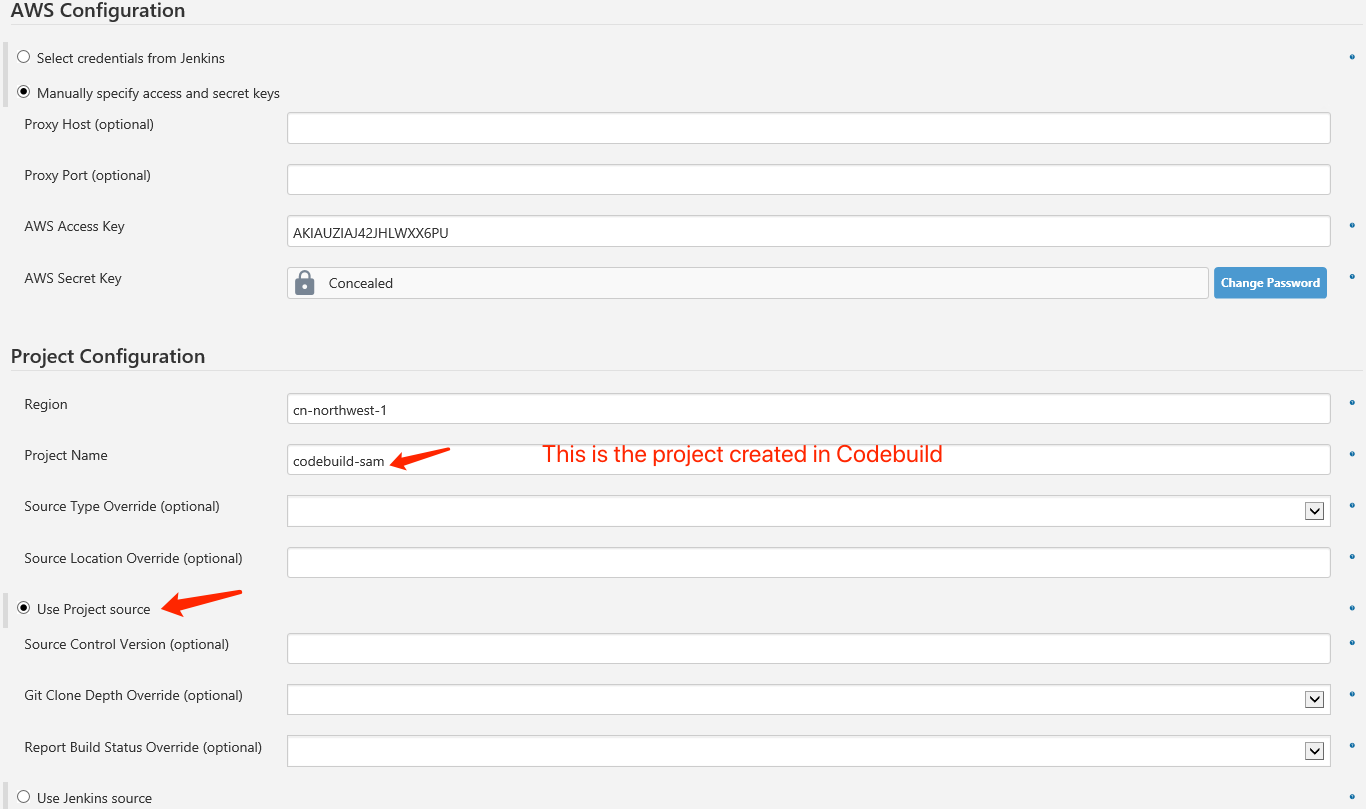
这里指定的build a job - jenkins为jenkins中定义的一个job，如下：

使用free style创建job，指定构建时使用的repo和branch

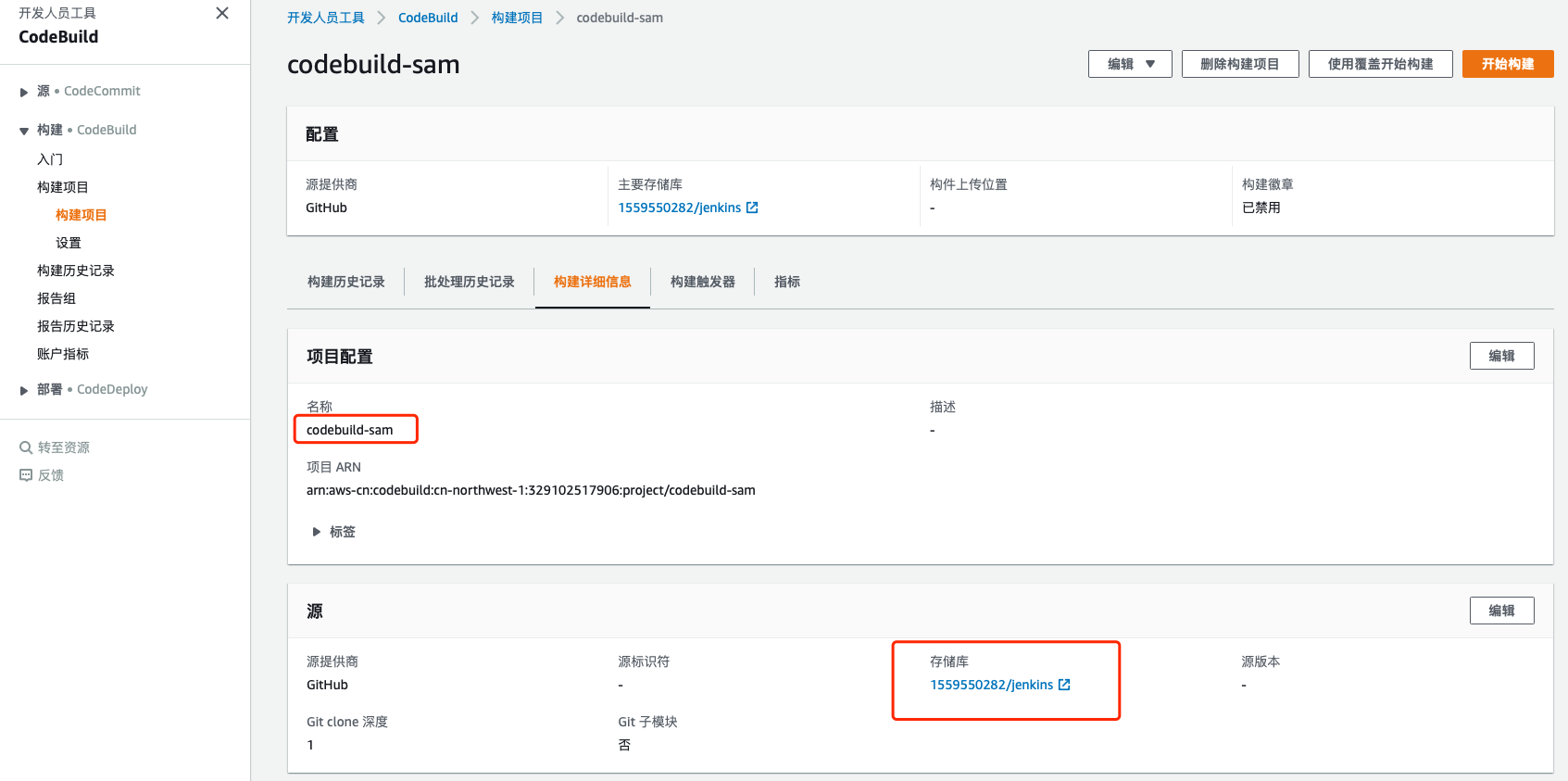


并增加codebuild构建步骤。





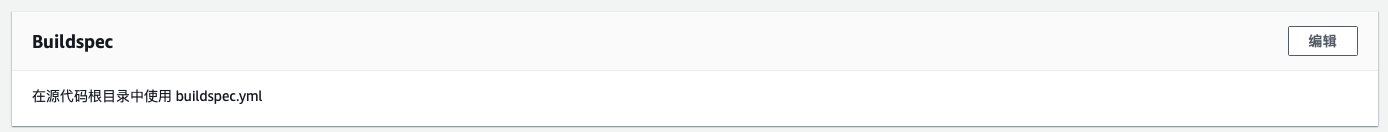
在AWS code build中对codebuild-sam的配置：



指定Codebuild使用的image环境：



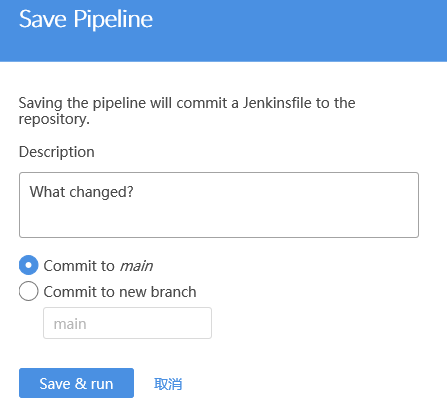
定义Code build使用的脚本(buildspec.yml的定义有模版)，保存在github repo的buildspec.yml文件：



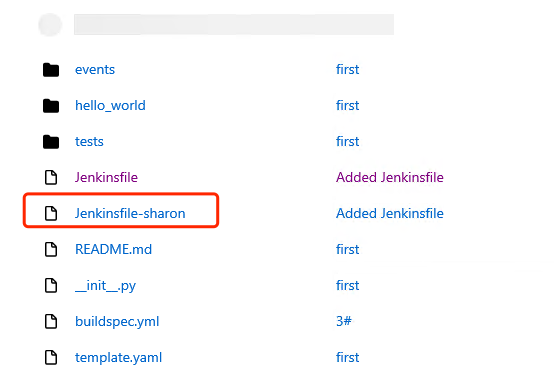




Pipeline定义完成，保存后，将保存在github repo的根目录，



保存后，进入github repos，可以看见该pipeline对应生成的定义文件



文件内容如下：

