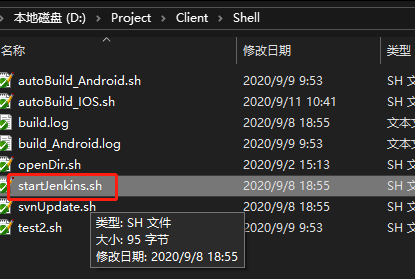
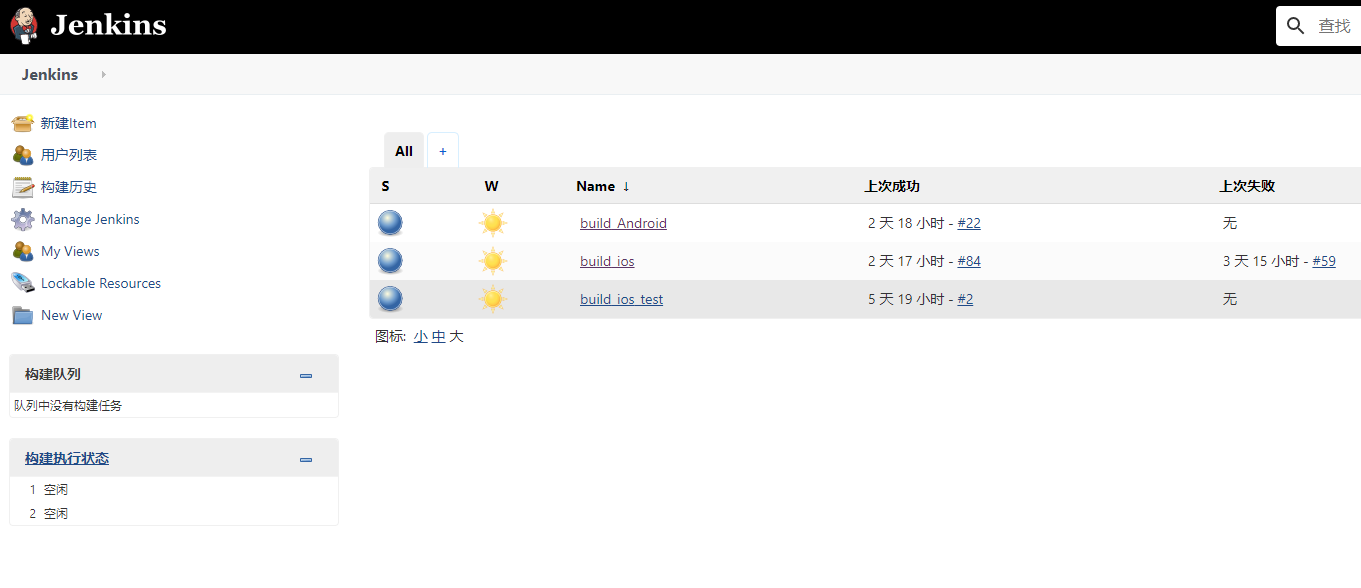
第一步,执行startJenkins.sh shell脚本开启Jenkins服务



第二步,登录管理员账号

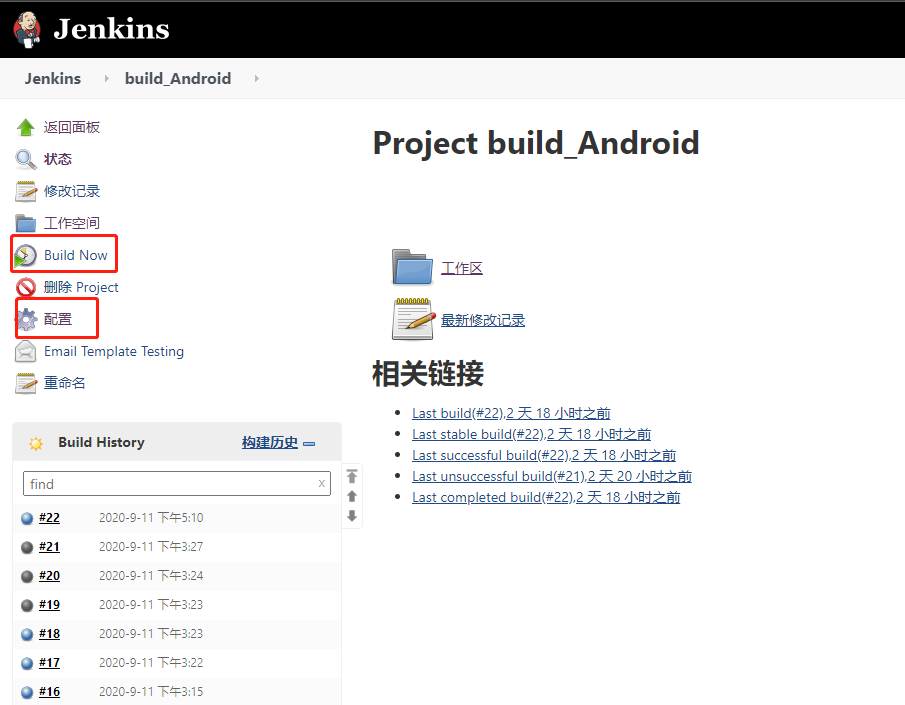
账号: admin

密码: admin



进入到主界面后,如图所示

第三步:点击build Android 的项目



Build Now是进行项目的构建

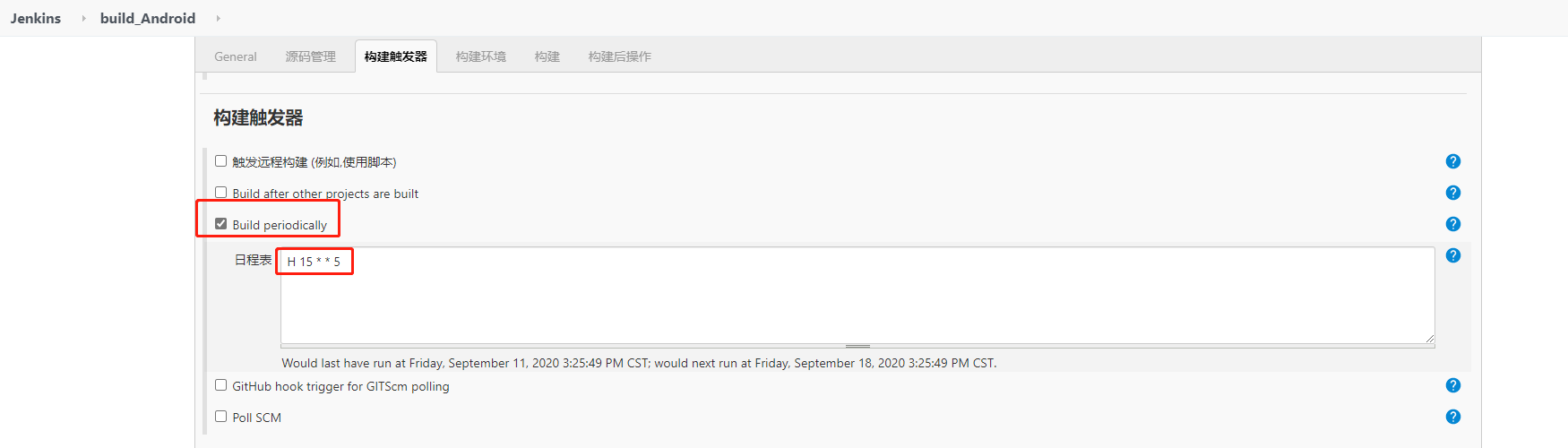
配置里面可以设置构建时,执行的功能

Build History时历史构建记录

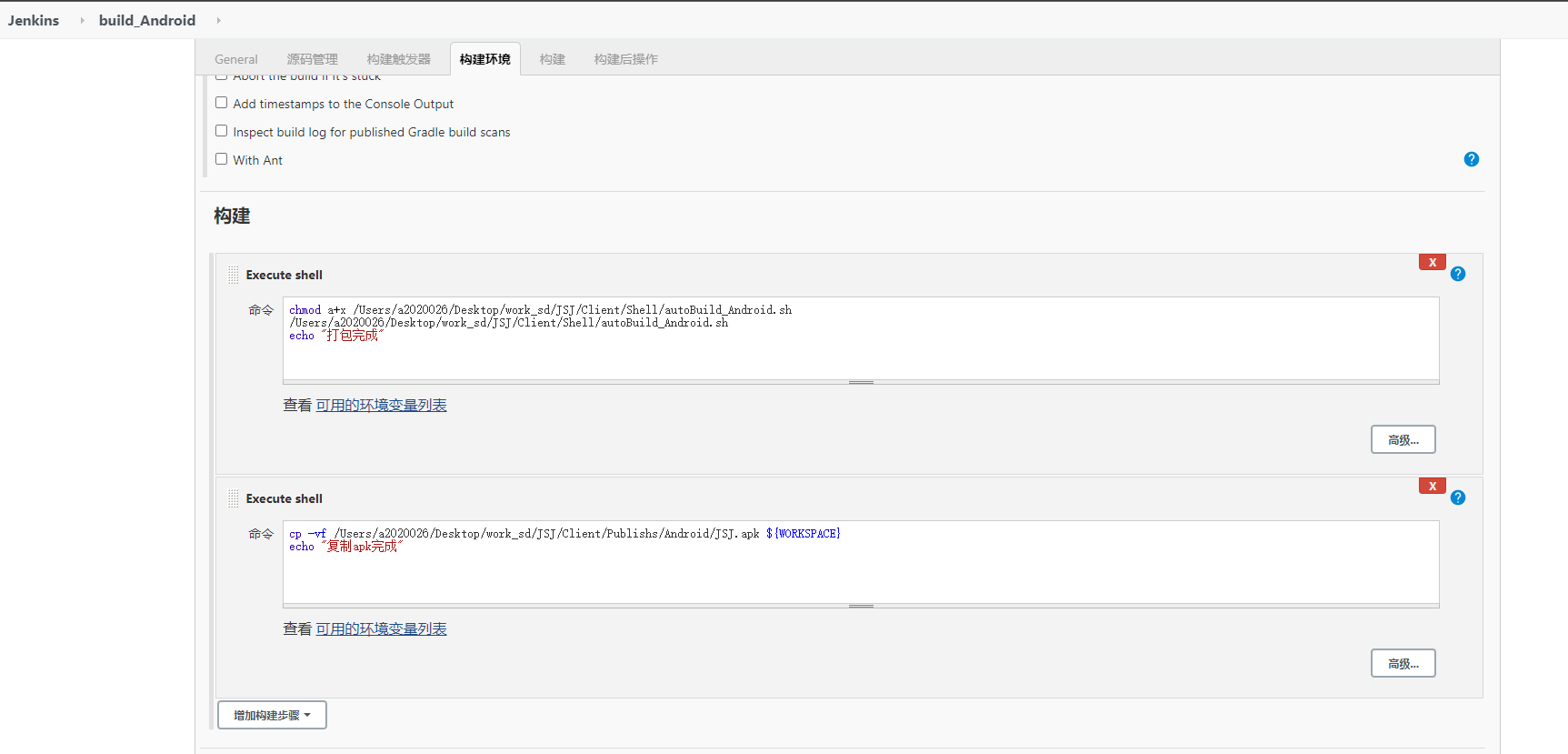
灰色的图标表示构建失败或者取消

第四步:进入到配置里面

在Build periodically 设置自动构建的时间,目前Android在每周五下午15点进行一次自动构建 H 15 \* \* 5



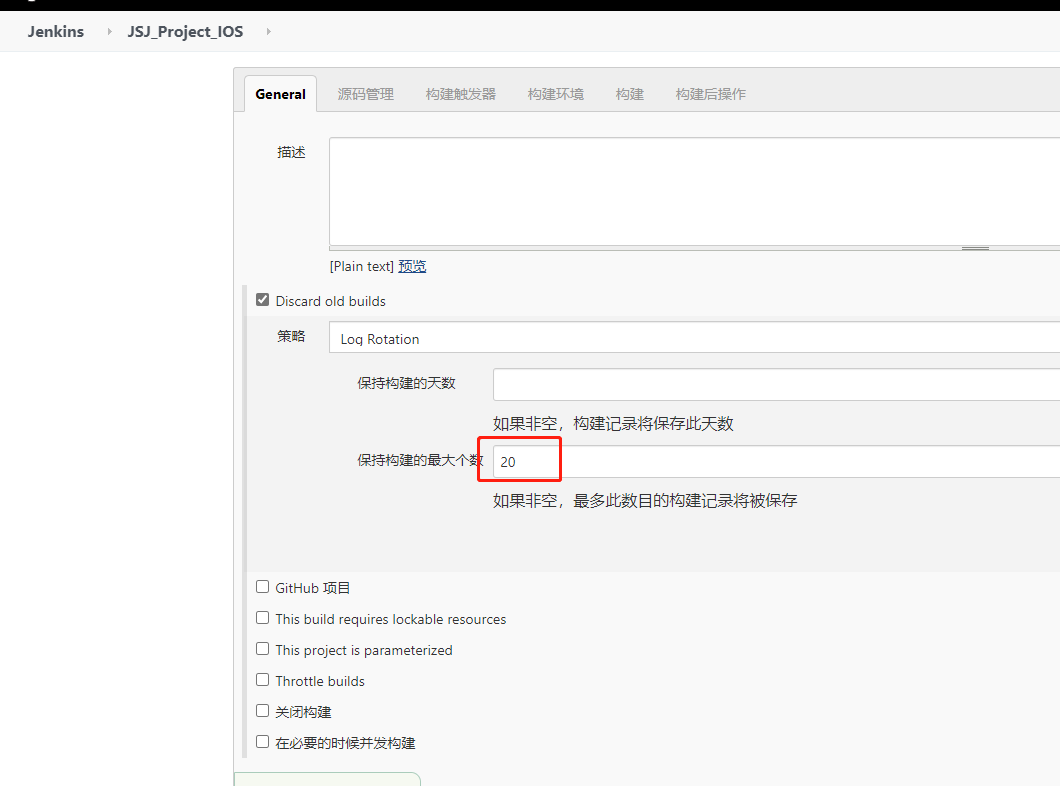
点击增加构建步骤,执行shell脚本.



首先是给执行脚本文件设置权限为可执行,然后执行autoBuild\_Android.sh脚本进行自动打包

打包完后,将apk复制到工作空间下

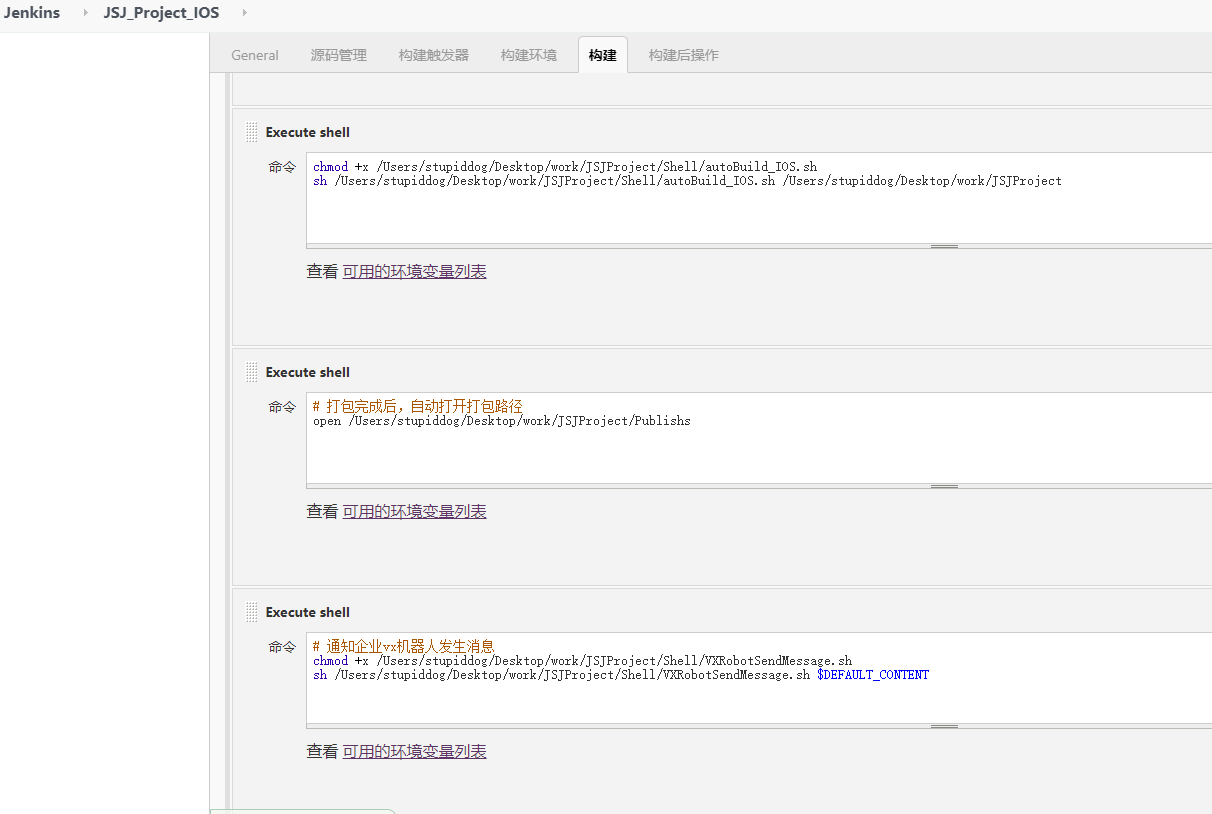
设置保留最新的20次构建记录



设置周3和周5下午15点自动打包

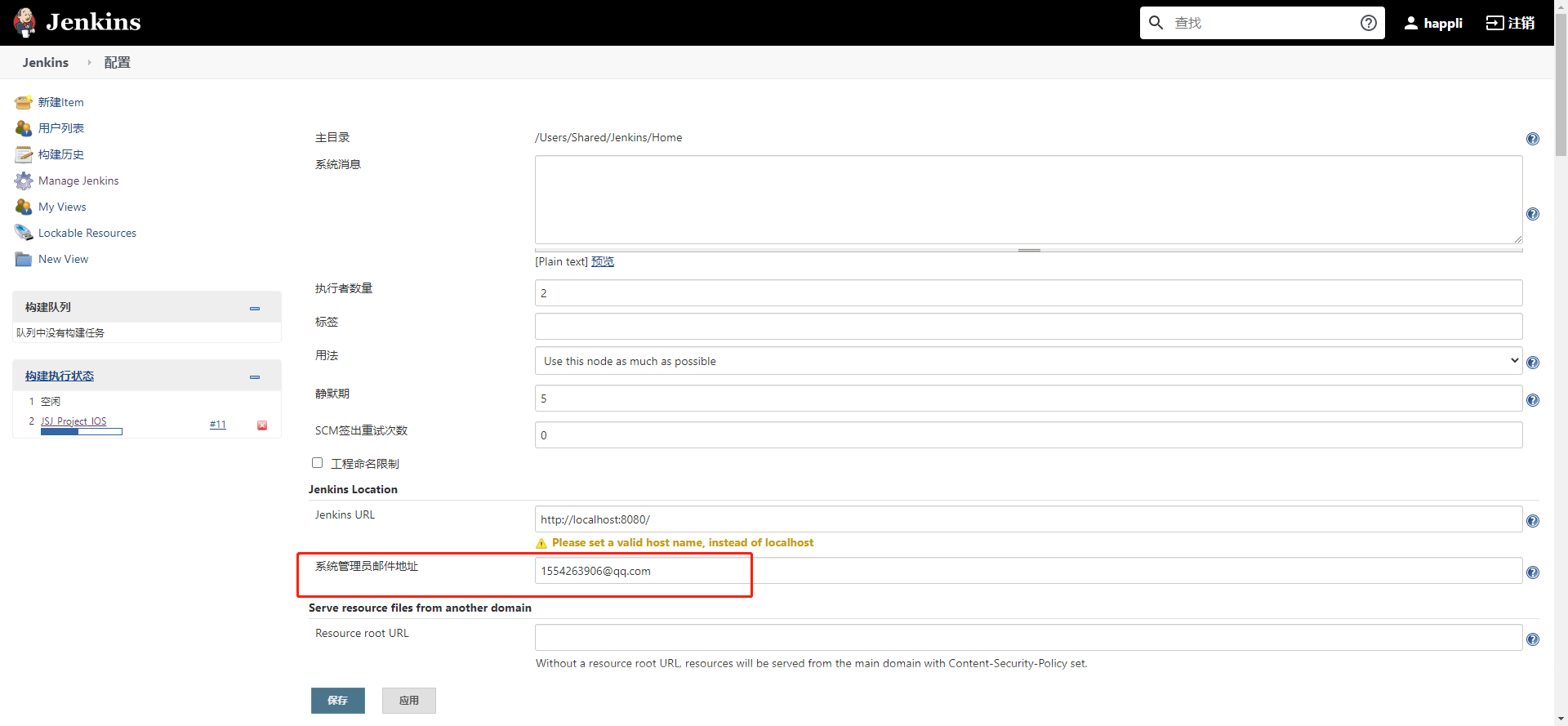


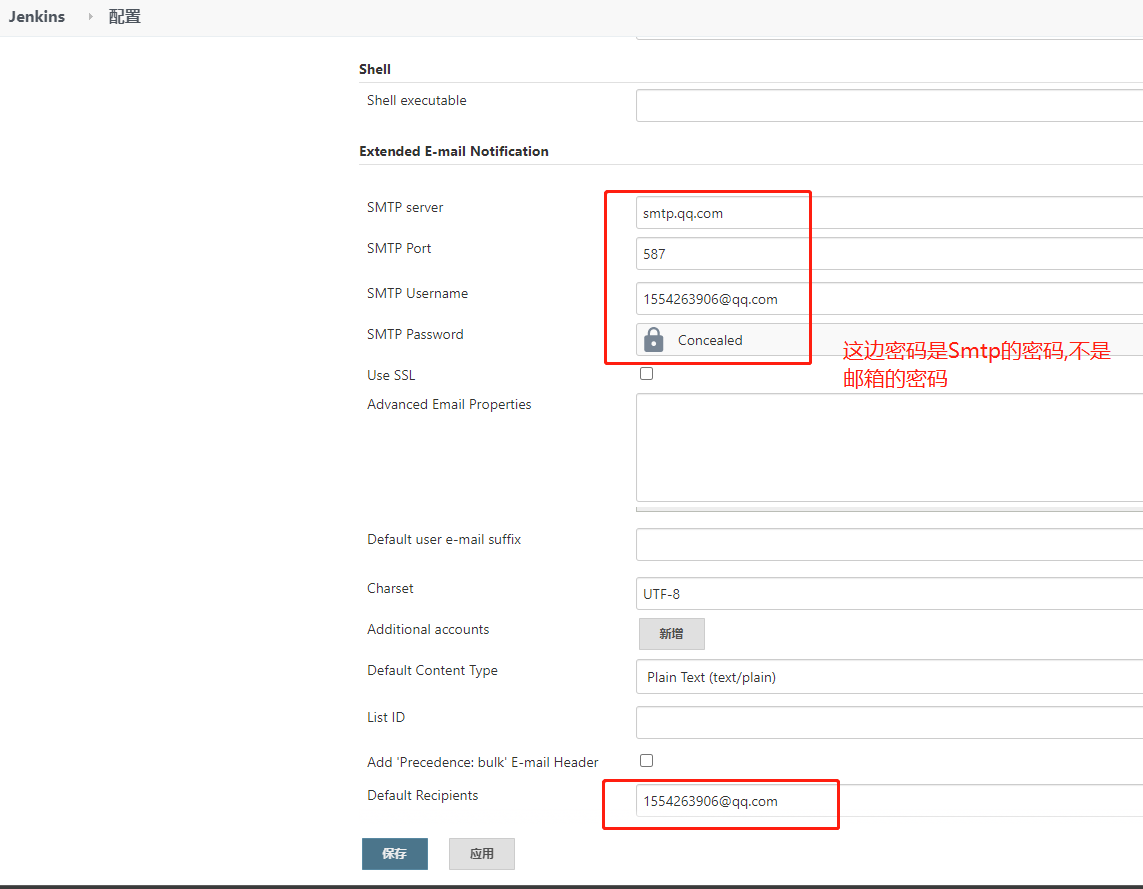
构建时,调用多个shell脚本进行操作

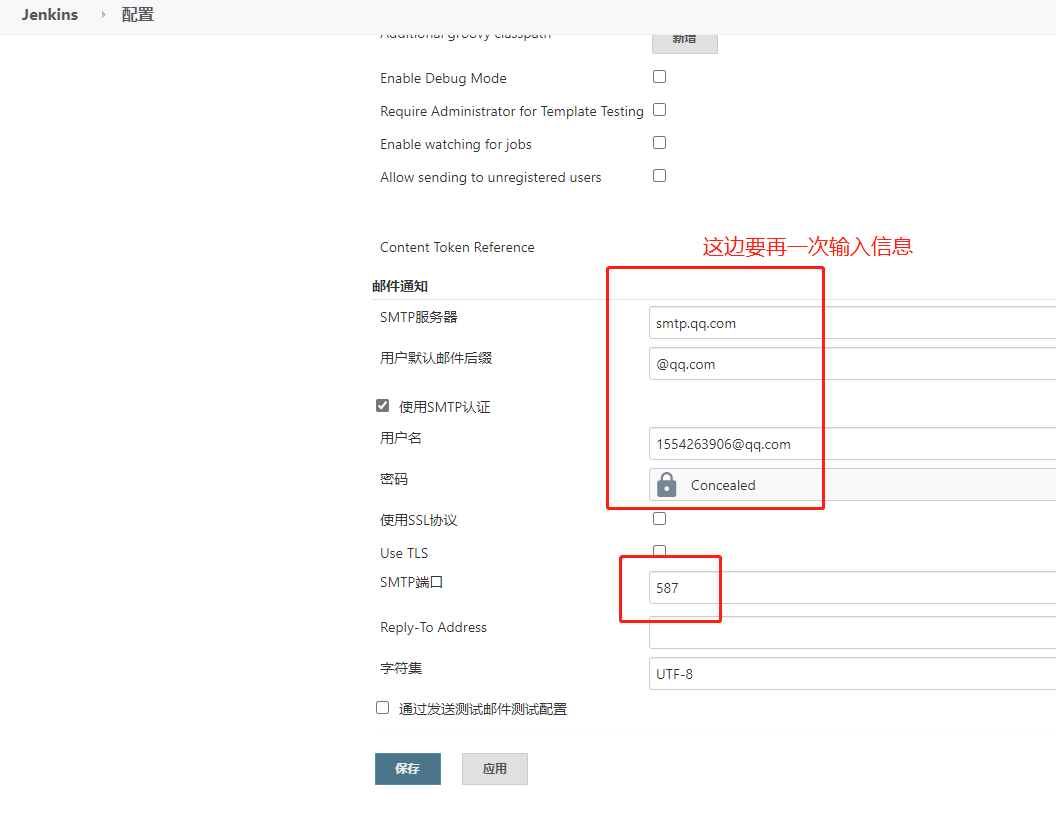


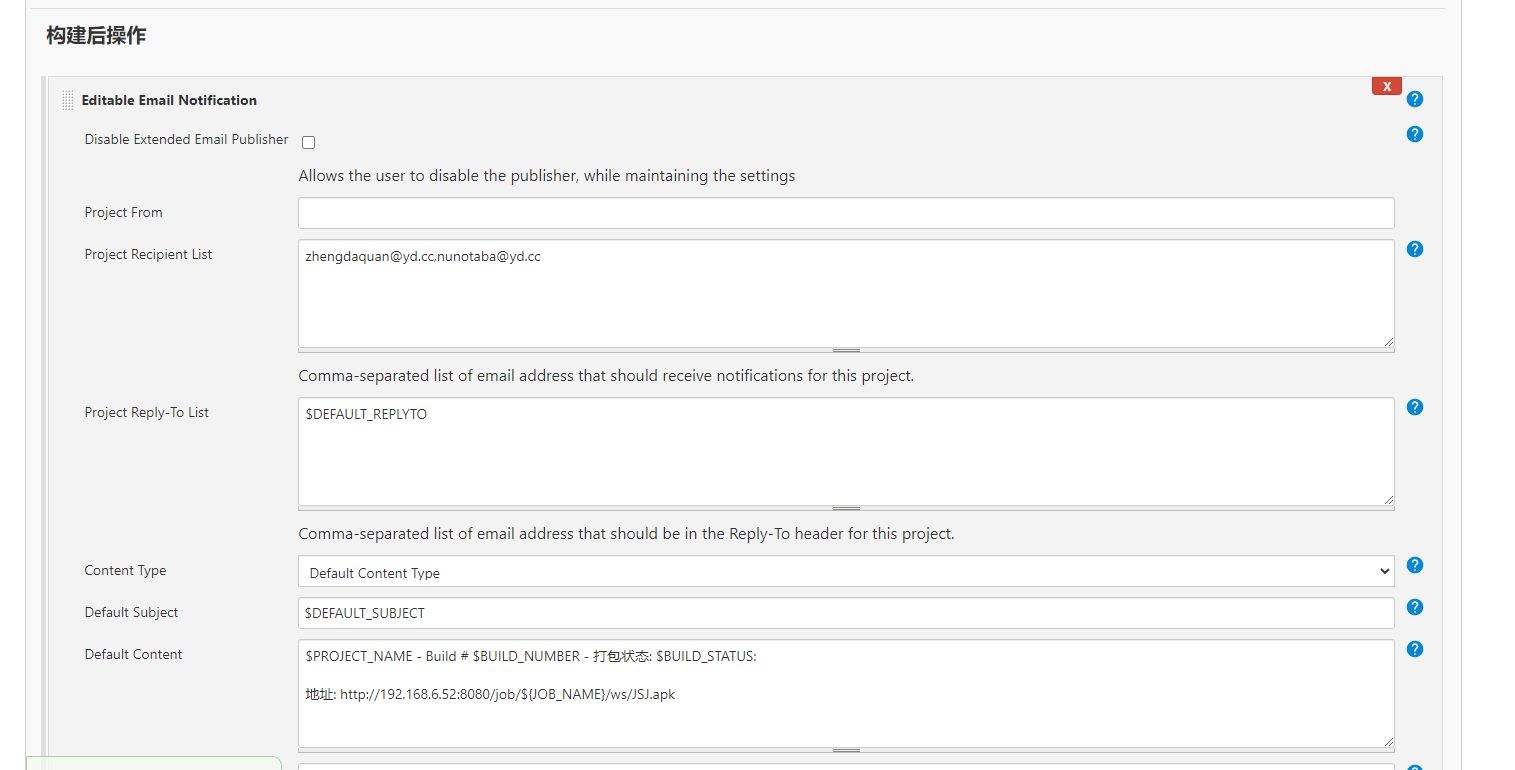
之后就是用邮件通知收件人,打包完成,附件带上打包log和文件路径方便查看和下载

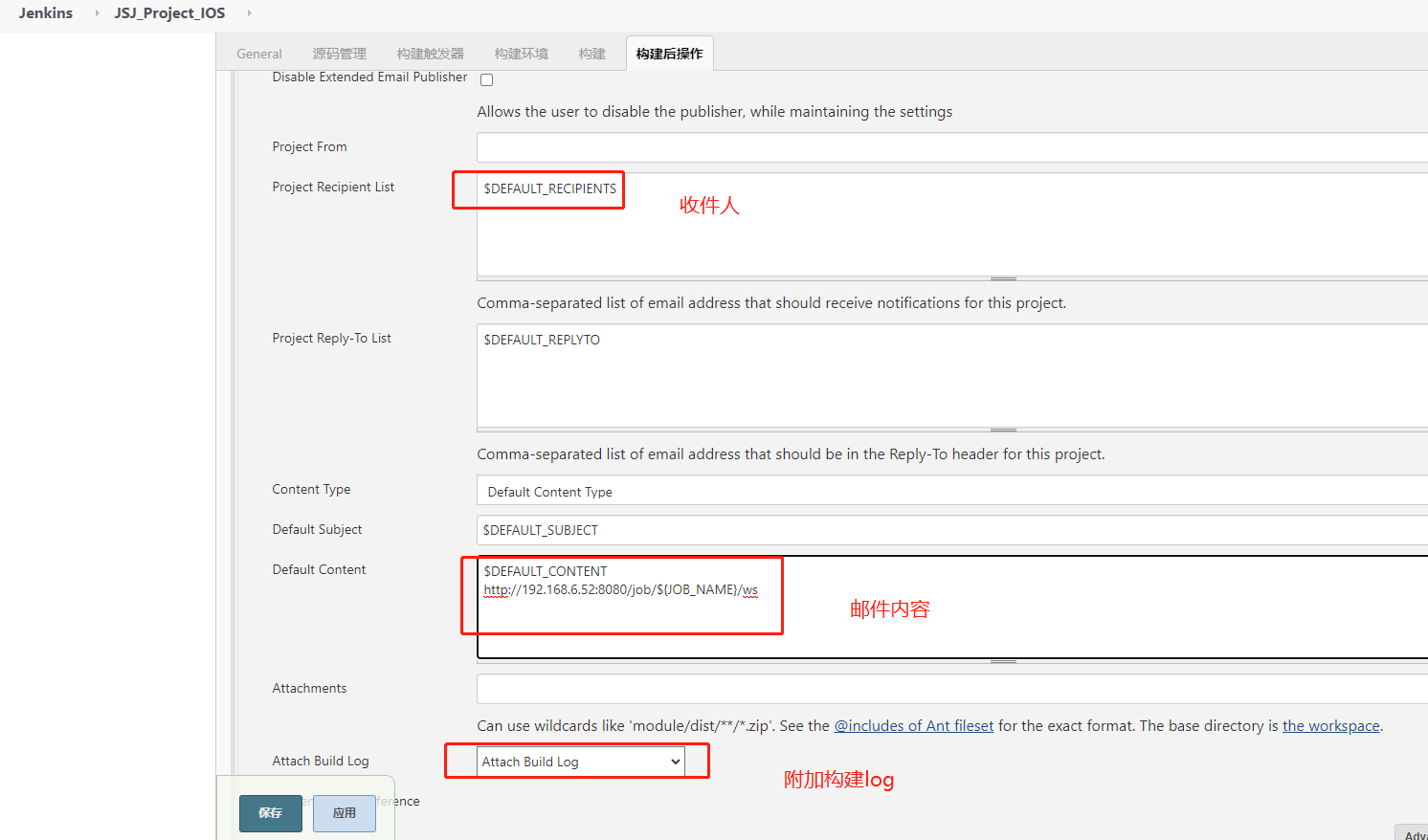
1. 首先需要到Jenkins的系统设置里面进行设置通知邮箱



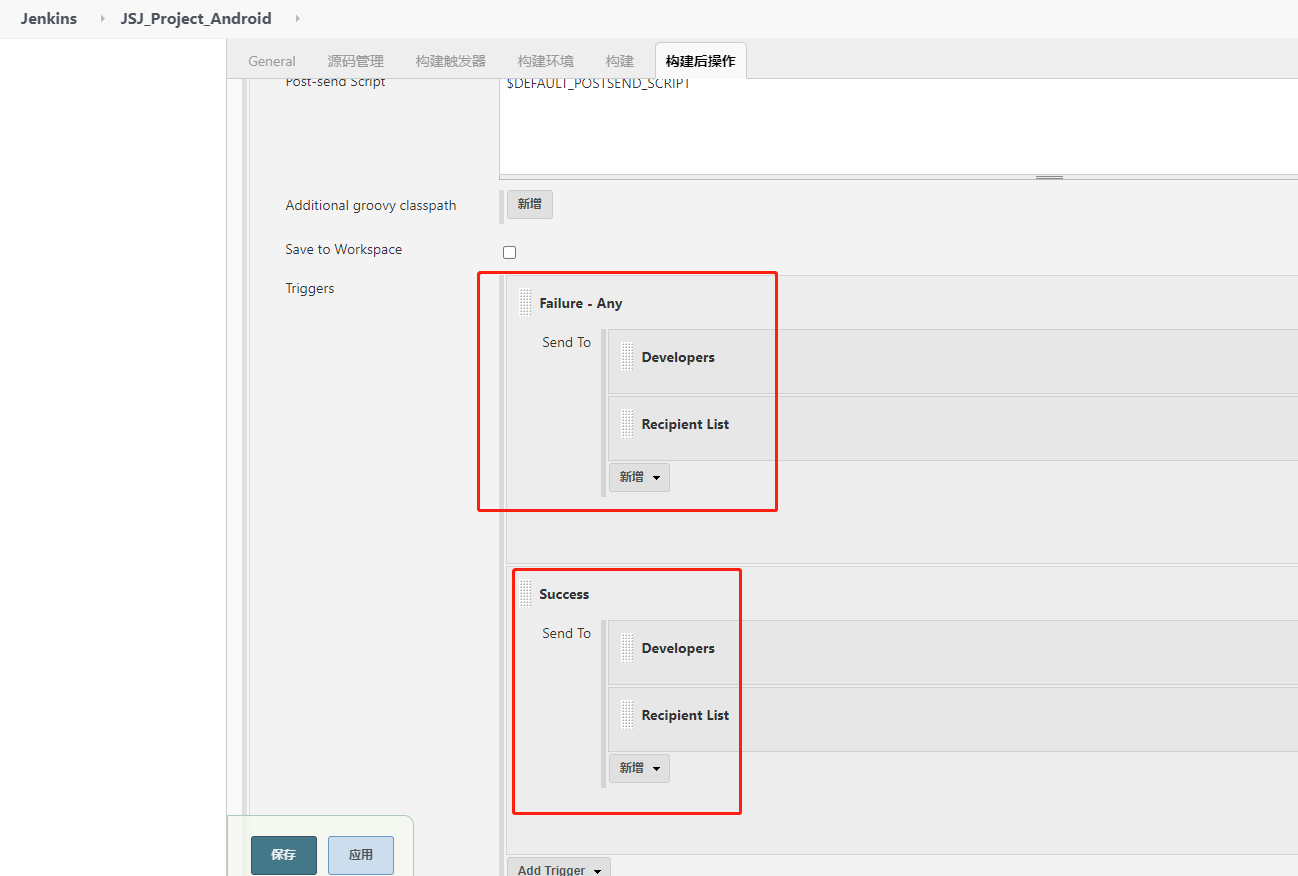








然后就是项目中需要添加触发条件,再 打包成功和失败后,都通知发送列表中的邮件



Project Recipient List 是收信人的邮件列表,用逗号隔开

Default Content 是邮件内容,

$PROJECT\_NAME 是项目的名字 这边就是 build\_Android

$BUILD\_NUMBER 是编译次数

$BUILD\_STATUS 是本次编译的结果,成功了还是失败了

由于邮件附件大小不能太大,所以apk就用web的形式进行通知,直接从web上进行下载

Mac苹果的电脑ip 192.168.6.52

Jenkins端口是8080

地址: [http://192.168.6.52:8080/job/${JOB\_NAME}/ws/JSJ.apk](http://192.168.6.52:8080/job/$%7bJOB_NAME%7d/ws/JSJ.apk)

这个是直接访问当前项目的工作空间底下的apk文件

基本上步骤就是这些,打IOS的时候,只是执行的shell脚本改成autoBuild\_IOS.sh

然后unity打包log文件会生成到和shell文件同层级中,

Android是 build\_Android.log

IOS是 build\_IOS.log

然后在执行shell脚本进行自动打包前,先调用svnUpdate.sh 的脚本进行更新,

然后将一下账号管理,目前Jenkins设置允许注册账号,如果不允许注册的话,要到系统设置里面进行设置.

还有就是Mac电脑一段时间不操作会自动待机.会导致Jenkins不能访问.所以要么设置为不待机,