**Jenkins+macos(虚拟机)自动化构建**

**虚拟机工具**：VMware Workstation Pro v15.0.0

**解锁工具**： Unlocker v3.0.0

**系统**： macOS Mojave 10.14懒人版

**软件下载**：https://pan.baidu.com/s/1mvDbcvaLtUz3eXQ6tYVRrg 提取码: 8tc8

**Vmware安装**

双击vm安装包开始安装vm15

虚拟机软件的安装向导界面单击“下一步”按钮

在最终用户许可协议界面选中“我接受许可协议中的条款”复选框，然后单击“下一步”按钮，

选择虚拟机软件的安装位置（可选择默认位置），选中“增强型键盘驱动程序”复选框后单击“下一步”按钮，

根据自身情况适当选择“启动时检查产品更新”与“帮助完善VMware Workstation Pro”复选框，然后单击“下一步”按钮

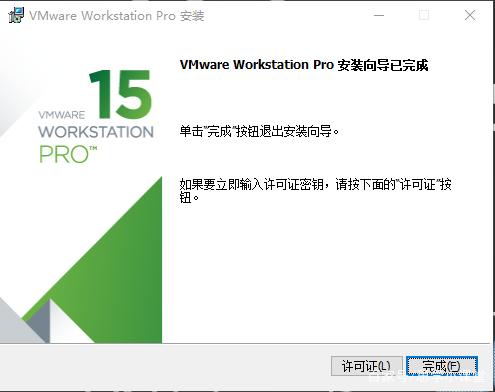
选中“桌面”和“开始菜单程序文件夹”复选框，然后单击“下一步”按钮

一切准备就绪后，单击“安装”按钮

进入安装过程，此时要做的就是耐心等待虚拟机软件的安装过程结束

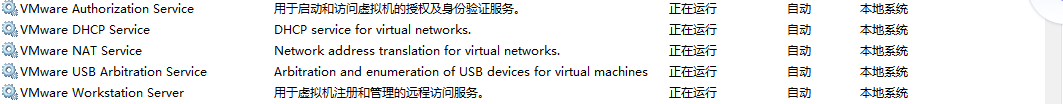
虚拟机软件便会安装完成，点击许可证，输入注册机产生的许可证编码即可自动激活vm

（注册机同vm安装包在同一文件夹，下载时直接下载到本地即可）



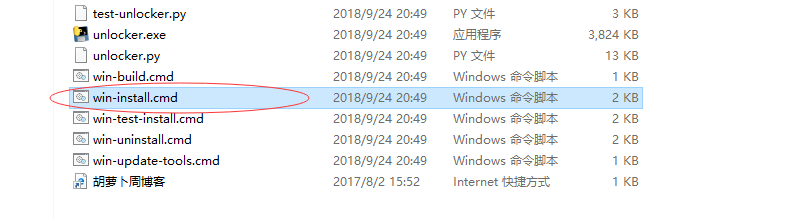
**Unlocker安装**

1：关闭vm服务



2：安装unlocker

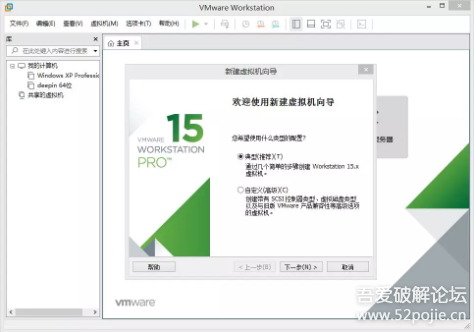
找到解锁工具，右键-以管理员身份运行win-install.cmd,脚本运行完毕会自动关闭



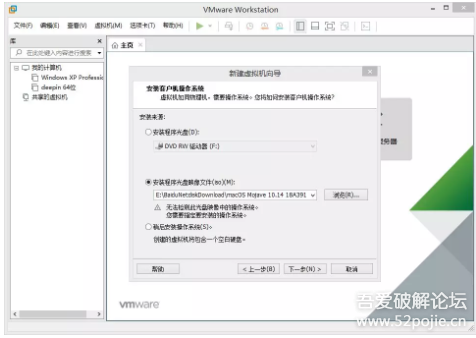
3:将之前关闭的vm服务全部启动

**创建虚拟机**

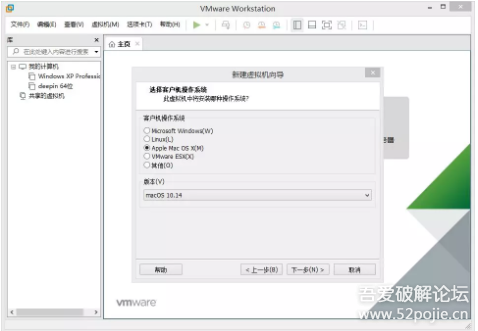
打开VMware，创建新的虚拟机，选择“典型”



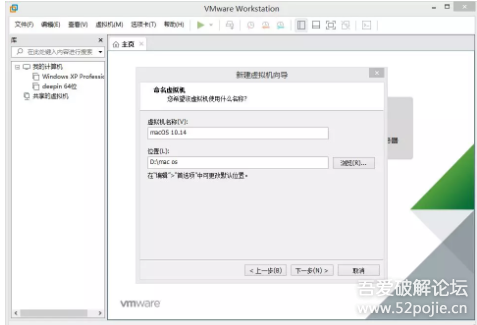
安装来源是macOS Mojave 10.14懒人版文件，注意浏览的时候文件类型要选择所有文件，不然找不到macOS Mojave 10.14 18A391 Lazy Installer.cdr



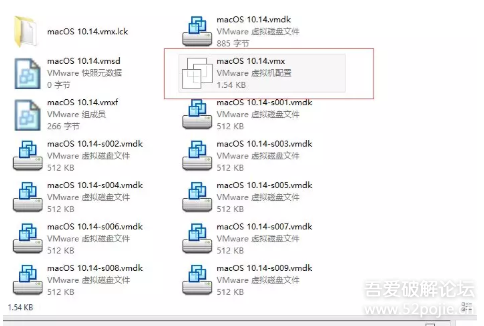
系统类型是MAC OS 10.14,如果没有以上的解锁操作，选项里是没有MAC OS可选的

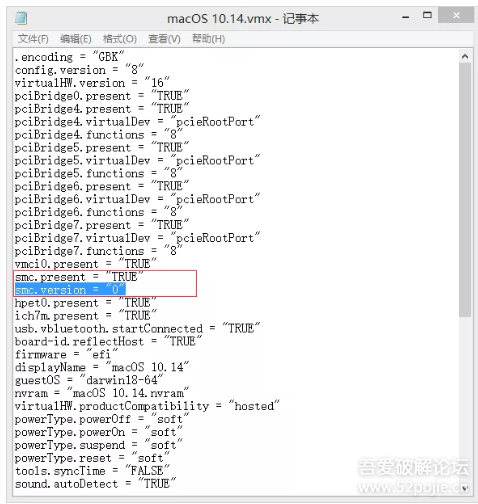


根据实际情况选择安装位置

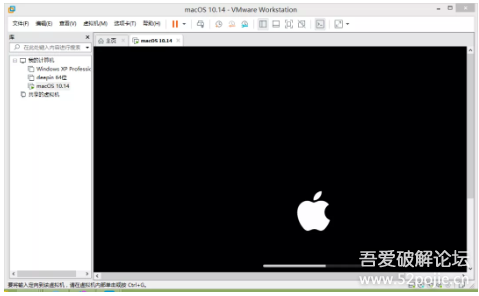


后续操作直接下一步，安装完成后先不要打开虚拟机，找到刚才虚拟机系统文件路径下的macOS 10.14.vmx，用记事本打开，在 smc.present = "TRUE" 后添加（smc.version = "0"）(建议您复制，不包括括号) 后保存。

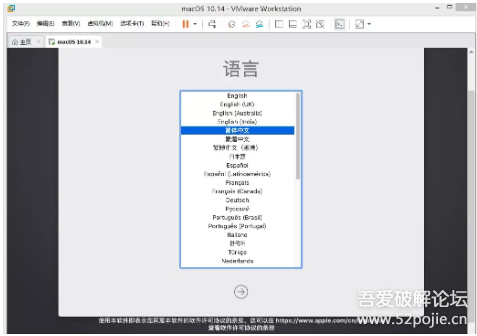




打开虚拟机，等待进度条加载完毕



语言选择简体中文，同意条款，继续安装

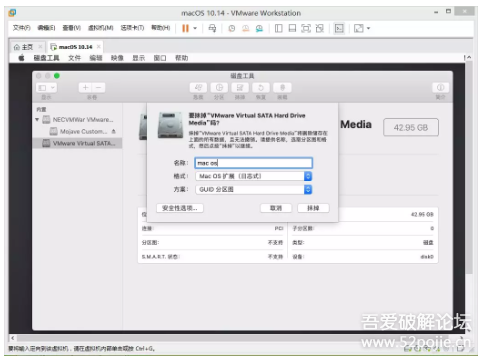


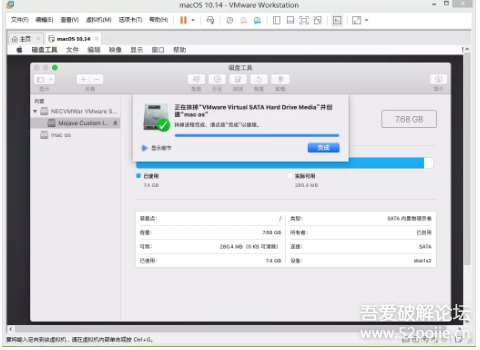


到这一步点击上方“实用工具”里的磁盘工具

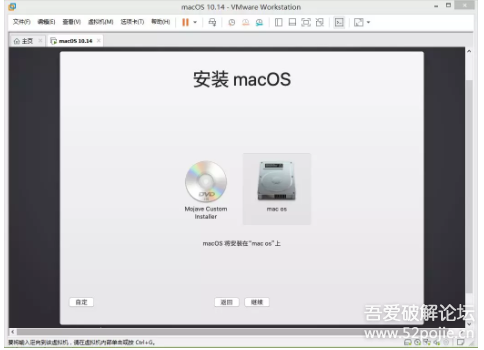


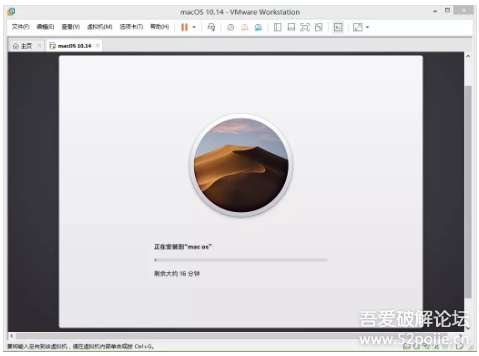
单击左侧的vmware虚拟硬盘，然后找到上方的“编辑”-“抹掉”，名称就叫mac os吧



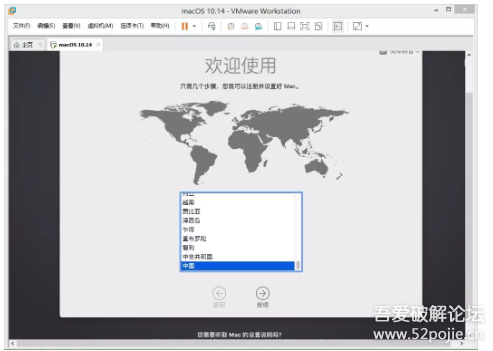


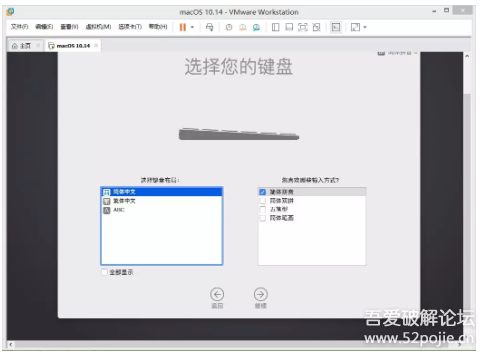
关闭磁盘工具，右侧会多出我们刚才分出来的一个磁盘，选择这个磁盘并继续。





待安装完成，国家选择中国，键盘选择简体中文，不传输信息，apple id稍后设置，创建用户名和密码。







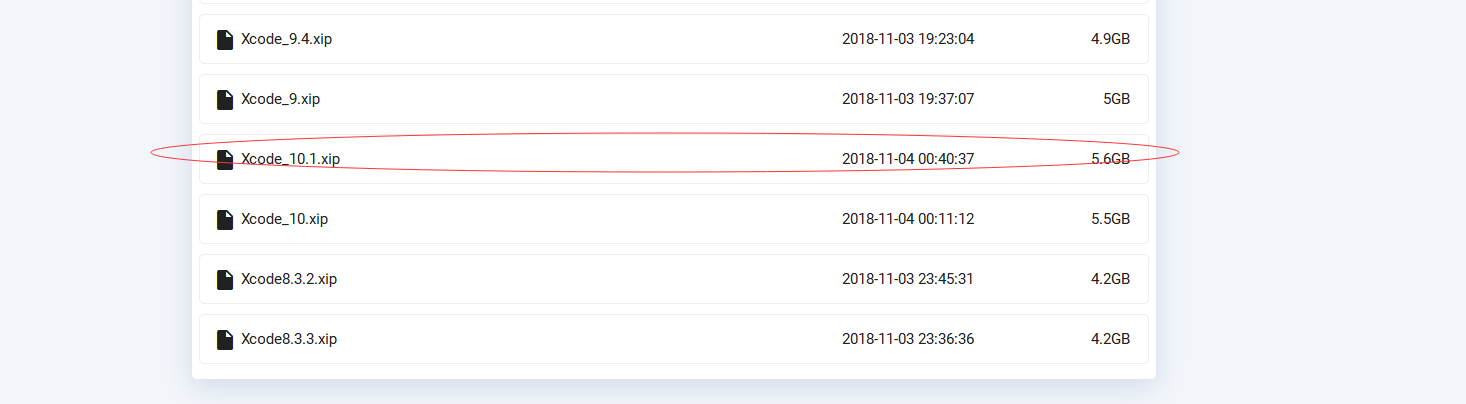


全部设置完毕即可进入系统，然后推出安装磁盘，至此macos系统安装完成

**安装xcode**

下载链接：<https://pan.macxin.com/Software/Xcode/>

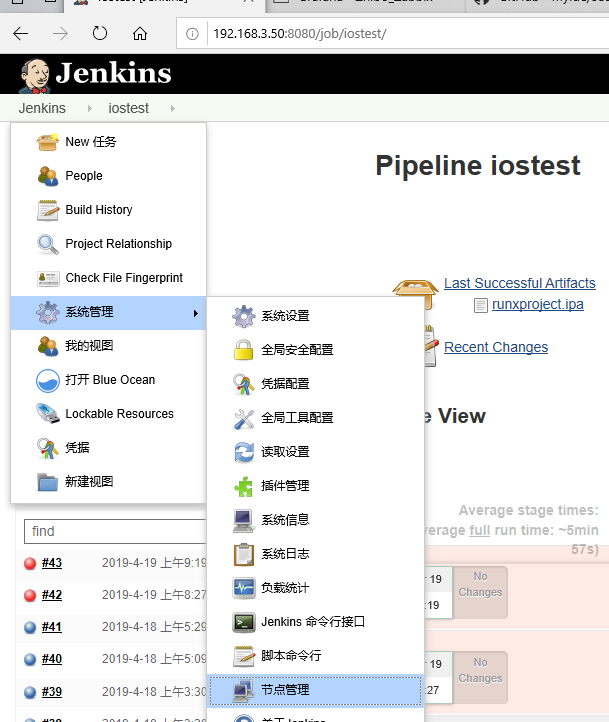
下载所需版本xcode10.1

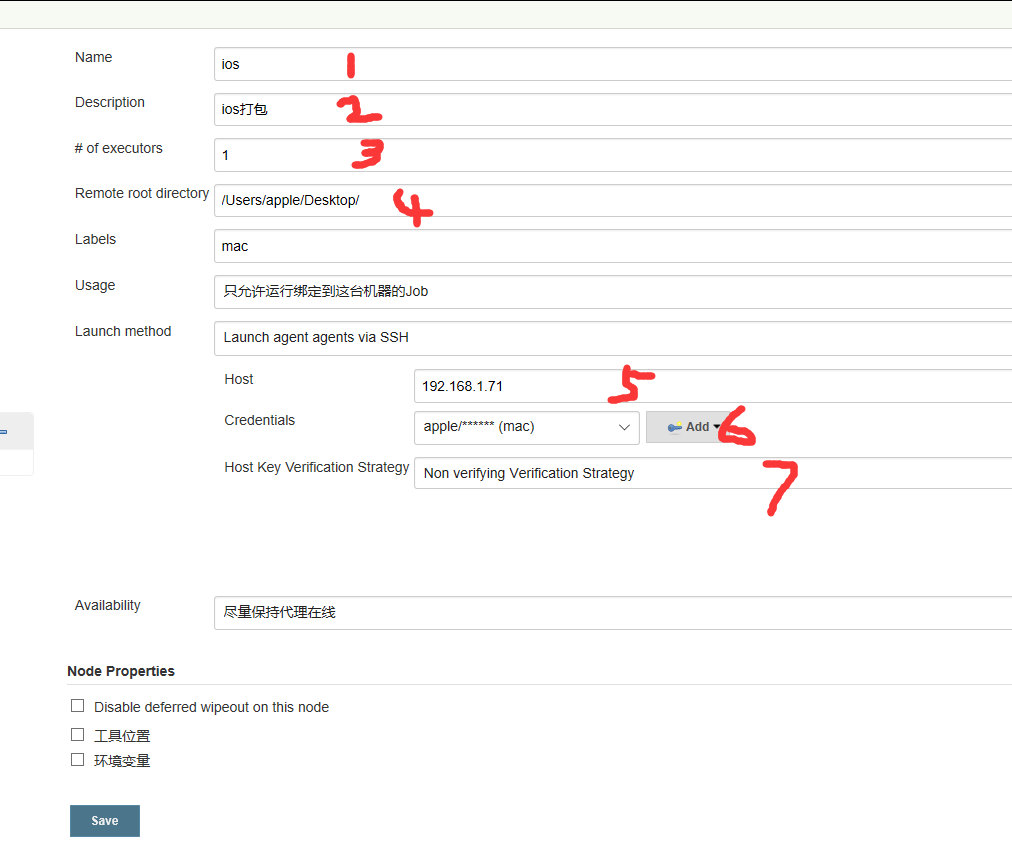


找到下载位置 双击安装包即可安装xcode

Jenkins设置mac节点

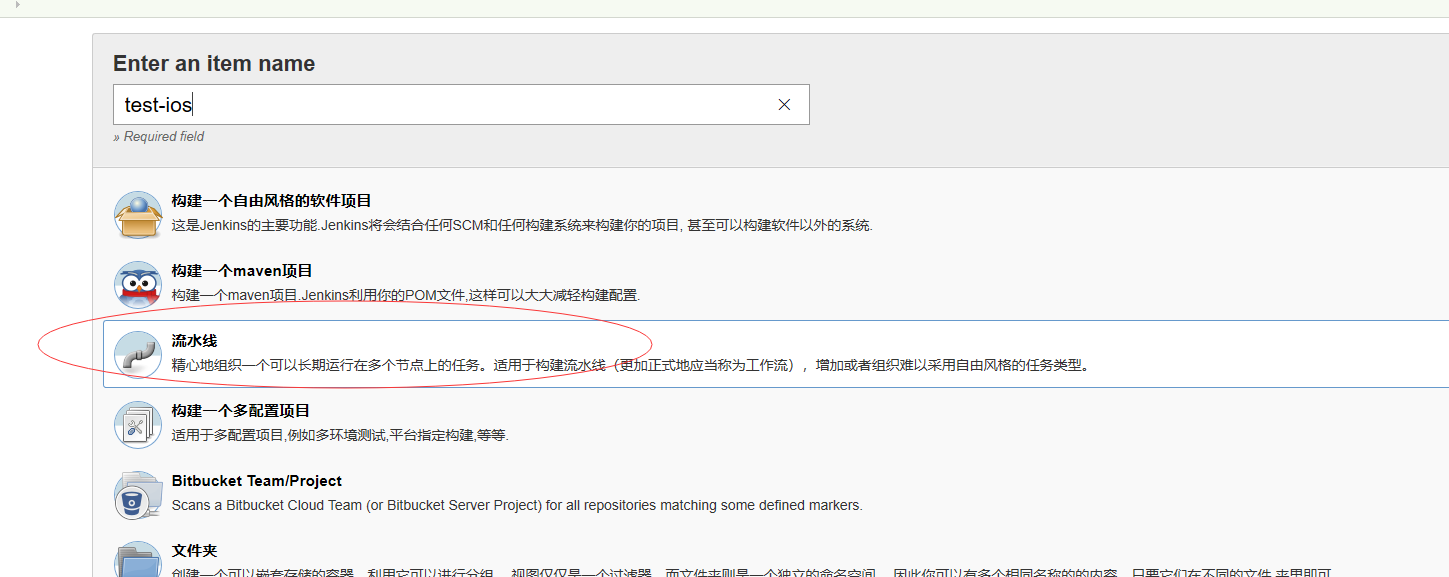
点解jenkins-系统管理-节点管理 配置mac节点





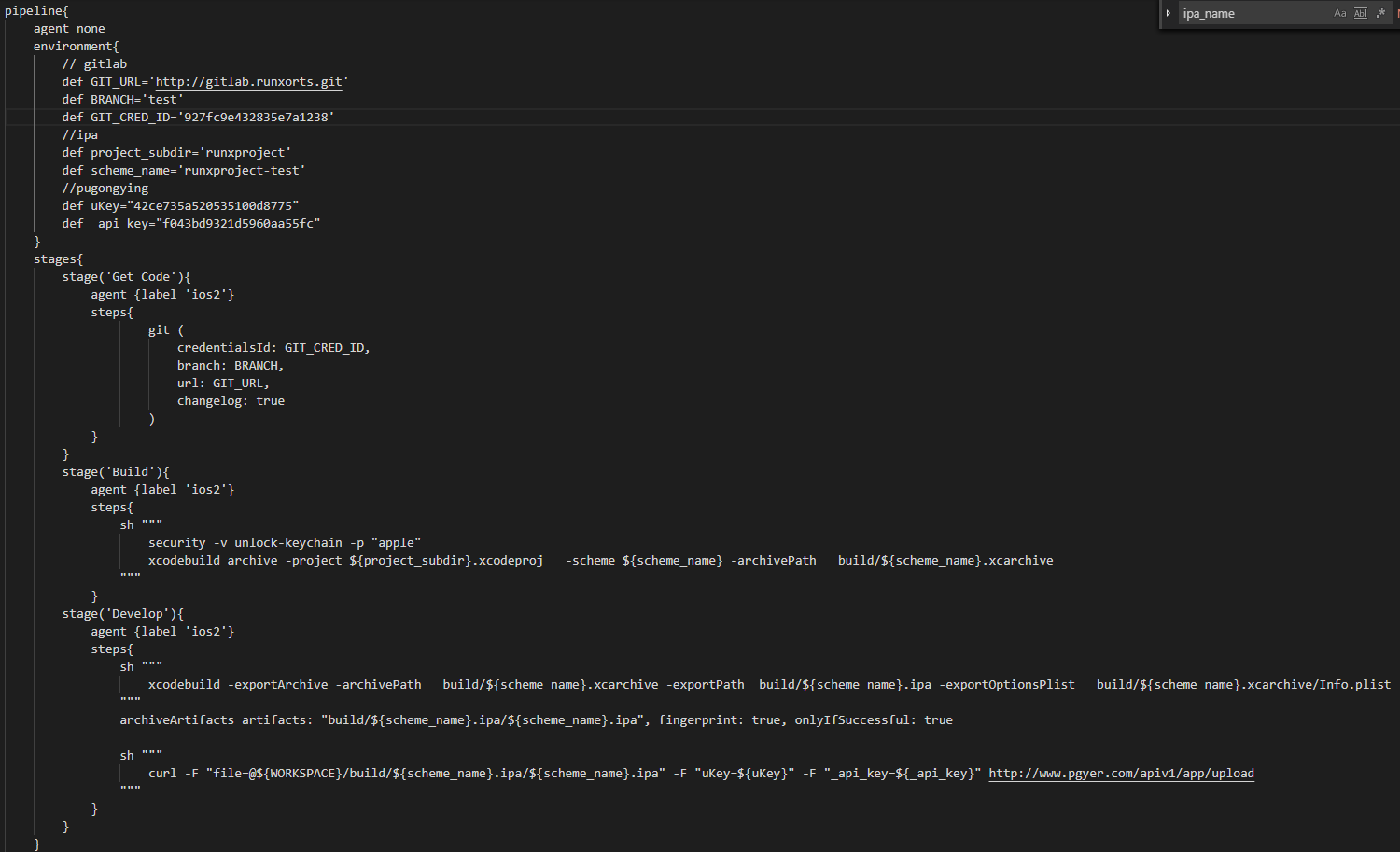
1. 节点名称
2. 节点描述
3. 同时最大构建数
4. 工作目录
5. 节点地址
6. 凭据
7. 主机密钥验证策略

创建jenkins任务，选择流水线



选择pipeline 添加构建命令

构建命令



关键命令详解：

security -v unlock-keychain -p "apple"（安全认证 –p后跟密码）

xcodebuild archive -project ${project\_subdir}.xcodeproj -scheme ${project\_subdir} -archivePath build/${project\_subdir}.xcarchive（构建）

xcodebuild –exportArchive –archivePath build/${project\_subdir}.xcarchive -exportPath build/${project\_subdir}.ipa -exportOptionsPlist build/${project\_subdir}.xcarchive/Info.plist

（打包）

curl -F "file=@${WORKSPACE}/build/${scheme\_name}.ipa/${scheme\_name}.ipa" -F "uKey=${uKey}" -F "\_api\_key=${\_api\_key}" <http://www.pgyer.com/apiv1/app/upload>

（将生成的ipa包传到第三方平台蒲公英以供下载）

其中：

**-project** 后面是.xcodeproj的路径

**-scheme** 后面是scheme的路径（通过配置不同的scheme文件区分不同环境）

**-archivePath** 后面是 打包好的 .xcarchive 的路径

**-exportPath** 后面是要导出的 .ipa 的路径

**-exportOptionsPlist** 后面是配置的 plist 文件路径

**archiveArtifacts artifacts** 存档

**注**：

1，打包过程中所需证书向ios开发索要

2，注册第三方平台蒲公英账号以供使用