

## 目录

目录	1
代码托管	2
1、哪些操作会被审计	2
2、Mac系统SSH协议clone卡死	2
流水线	2
为何不支持代码库打Tag自动触发?	2
为何不支持手动触发	2
任务执行过慢	2
主机部署任务执行后无效	2

# 代码托管

## 1、哪些操作会被审计

通过代码库设置菜单，可以查到“操作审计”，具体会被记录到操作审计中的操作为：

- 删除分支和push -f强制更新分支
- 删除Tag及Tag强制更新
- Clone代码(不含git pull、git fetch)，用以审计离职前大量Clone代码行为
- 修改代码库基本设置
- 修改分支提交规则
- 修改代码库成员

## 2、Mac系统SSH协议clone卡死

主要是部分版本（例如10.15.4）的Mac系统的openssh（例如OpenSSH\_8.1p1）有bug导致，需要将openSSH升级到最新版本OpenSSH\_8.2p1及以上。

未安装brew请先安装brew：

```
bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
brew install openssh
```

安装完毕后，打开一个新的终端，输入如下命令，确认一下openSSH版本：

```
ssh -V
```

# 流水线

## 为何不支持代码库打Tag自动触发？

A：原因是这样会把正常研发流程：代码提交、测试、发版(打Tag，归档构建产出至成品库)、上线的研发流程打破，把发版前后的流程割裂到两条流水线中，这是我们不提倡的。

我们提倡的研发实践如下：

例如某个团队是分支开发，分支发版（在开发分支上直接打Tag）的研发模式，则可以在开发分支上配置编译、测试、发版(打Tag，归档构建产出至成品库)、上线的全流程流水线。

例如某个团队是分支开发，代码合入发布分支(通常是Master分支)后，统一在发布分支打Tag的研发模式，则可以在开发分支配置编译、测试的流水线。在发布分支配置编译、测试、发版(打Tag，归档构建产出至成品库)、上线的全流程流水线，且由代码从开发分支合入发布分支时自动触发。

## 为何不支持手动触发

A：其实是所有流水线都支持手动触发、具体参考“流水线管理 -> 手动执行”章节。

系统暂时不支持仅手动触发，如果想实现类似效果，可以将第一个阶段配置为“手动触发”方式即可。这样定时触发或者代码提交自动触发生成一条空记录方便在需要的时候补充执行。如果只支持手动触发，则对历史的构建进行触发比较麻烦。具体效果参考下图灰色流水线构建记录。



## 任务执行过慢

通常是由于构建集群资源不足导致。请及时扩容构建集群或者取消任务重新尝试。

## 主机部署任务执行后无效

有些用户可能会很奇怪，明明自己的流水线部署插件执行了，但是应用却没有更新。

原因是流水线插件只是执行创建部署任务的工作，只要部署任务创建成功，流水线插件即显示为成功。

应用没有更新的可能导致因素为如下两种：

1. 该部署任务对应的部署模板被设置了需要审批才可执行。
2. 或者部署任务执行中导致了失败。

点击下图的“查看任务详情”即可跳转到实际的部署单界面，查看详情。

