

- 1、安装jenkins，下载jenkins.war 放到tomcat 或yum install jenkins 建议yum 装
- 2、安装 Android_SDK ,再安装build-tools 和api要和android 开发所用的版本一致
- 3、安装 Android_NDK(问一下开发，需要的话安装)
- 4、安装java
- 5、安装gradle(版本和android开发本地的版本必须一致)
- 6、设置 sdk、ndk、java、 gradle环境变量

```
1. export JAVA_HOME=/opt/java
2. export CLASSPATH=.:$JAVA_HOME/jre/lib/rt.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/t
3. export PATH=$PATH:$JAVA_HOME/bin
4. export ANDROID_NDK_HOME=/opt/android-ndk
5. export PATH=$PATH:$ANDROID_NDK_HOME
6. export ANDROID_HOME=/opt/android-sdk-linux
7. export PATH=$PATH:$ANDROID_HOME/tools:$ANDROID_HOME/platform-tools
8. export GRADLE_HOME=/opt/gradle
9. export PATH=$PATH:$GRADLE_HOME/bin
```

7、安装360加固linux版

给360jiagubao@360.cn 要最新版，或者 <https://yunpan.cn/cWkg4YV5hmpHlU>（提取码：cfb9）下载传到服务器解压缩

- 8、jenkins安装 [Android Lint Plugin](#) gradle Plugin git 钉钉 等插件，然后在global tool configuration里配置好



Global Tool Configuration

Maven Configuration

Default settings provider

Use default maven settings

Default global settings provider

Use default maven global settings

JDK

JDK 安装...

Git

Git installations



Git

Name

git1.8

Path to Git executable

/usr/bin/git

新建 android 项目

源码管理

☐ None

☒ Git

Repositories

Repository URL

Credentials

高级...

Add Repository

Branches to build

Branch Specifier (blank for 'any')

Add Branch

源码库浏览器

(自动)

保存

Apply

Add

配置gradle

构建

Invoke Gradle script

Invoke Gradle

Gradle Version

gradle

Use Gradle Wrapper

Switches

build --debug --info --stacktrace

Tasks

Root Build script

\${workspace}/app

Build File

Specify Gradle build file to run. Also, [some environment variables are available to the build script](#)

Force GRADLE_USER_HOME to use workspace

as Gradle properties

保存

Apply

增加一个构建 步骤 执行shell脚本，用来加固和多渠道打包

Execute shell

Command

python /opt/android_autobuild/jiagu.py

See [the list of available environment variables](#)

高级...

加固脚本，能运行就行了，别管为啥要for

```
1.  #!/usr/bin/python
2.  # coding:utf-8
3.  # 贝加尔 20150527  ly
4.
5.  import time
6.  import os
7.  import psutil
8.  import subprocess
9.  import ddmsg
10.
11.  robot_url = 'https://oapi.dingtalk.com/robot/send?access_token=xxxxab2762xxxxxxxd7'
12.  #这人是你的机器 人的url
13.  stamp = time.strftime('%Y%m%d%H%M%S', time.localtime())
14.  root_path = '/var/ftp/pub'
15.  #这是ftp或者nginx 文件服务的目录，ftp最好匿名
16.  build_path = '/var/lib/jenkins/workspace/AAwork_Android/app/build/outputs/apk'
17.  #这是打包后apk的目录
18.  ftpdir = os.path.join(root_path, stamp)
19.  os.mkdir(ftpdir)
```

```

20. apklist = os.listdir(build_path)
21.
22. for apk in apilist:
23.     srcapk = os.path.join(build_path, apk)
24.     cmd = ["java", "-jar", "/opt/360jiagubao_linux_64/jiagu/jiagu.jar", "-jiagu", s
25.     subprocess.call(cmd)
26.
27. msg = ddmmsg.DingMsg(robot_url)
28. msg.send()

```

这是发钉钉消息的脚本，半成品，能用就行了,大概意思就是这样

```

1.  #!/usr/bin/python
2.  # coding:utf-8
3.  # 贝加乐 20170527 ly
4.  import json
5.  import urllib2
6.
7.  robot_url = 'https://oapi.dingtalk.com/robot/send?access_token=c3f97911ffe2a7ab2762
8.  class DingMsg(object):
9.      """docstring for DingMsg"""
10.     def __init__(self, url):
11.         super(DingMsg, self).__init__()
12.         self.url= url
13.         self.header_key = 'Content-Type'
14.         self.header_val = 'application/json'
15.
16.     def send(self):
17.         jmsg = self.__sendLink()
18.         req = urllib2.Request(self.url, jmsg)
19.         req.add_header(self.header_key, self.header_val)
20.         rp = urllib2.urlopen(req)
21.
22.
23.     def __sednText():
24.         pass
25.
26.     def __sendLink(self):
27.         msg = {"msgtype":"link", "link":{"text":"请下载最近日期的安装包","title":"下
28.         , "messageUrl":"http://192.168.0.185"}}
29.         return json.dumps(msg).encode('utf-8')
30.     def __sendFeedCard():
31.         pass
32.     def __sendMarkdown():
33.         pass

```

这是钉钉消息



点击消息就可以跳转到内网的nginx文件服务器上下载文件，手机连上wifi就可以直接下载安装

<div><div><div></div><div></div><div></div></div><div>192.168.0.185</div></div>	
Index of /	
<hr/>	
../	
20170527153552/	27-May-2017 15:00
20170527153614/	27-May-2017 15:00
20170527153659/	27-May-2017 15:00
20170527153928/	27-May-2017 15:00
20170527154131/	27-May-2017 15:04
20170527154242/	27-May-2017 15:04
20170527155832/	27-May-2017 16:00
<hr/>	

Index of /20170527155832/

[illegible]

这是360加固的帮助文件

```

1. /opt/360jiagubao_linux_64/jiagu
2. ? -login <username> <password> 首次使用必须先登录
3. -importsign <keystore_path> <keystore_password> <alias> <alias_password>
4. 导入签名信息
5. -importmulpkg <mulpkg_filepath> 导入多渠道配置信息,txt格式
6. -showsign 查看已配置的签名信息
7. -showmulpkg 查看已配置的多渠道信息
8. -help 显示帮助信息
9. -jiagu <inputAPKpath> <outputPath> 加固应用,可选参数
10. [-autosign] 自动签名
11. [-automulpkg] 自动多渠道
12. [-pkgparam mulpkg_filepath] 使用mulpkg_filepath配置文件的渠道信息

```

先登录，再导入签名文件，再导入渠道配置文件，这是我的渠道

```
1. UMENG_CHANNEL 360应用平台 360
2. UMENG_CHANNEL 小米 xiaomi
3. UMENG_CHANNEL 联想 lenovo
4. UMENG_CHANNEL 华为 huawei
5. UMENG_CHANNEL 魅族 meizu
6. UMENG_CHANNEL 百度 baidu
7. UMENG_CHANNEL oppo oppo
8. UMENG_CHANNEL vivo vivo
9. UMENG_CHANNEL 应用宝 yyb
10. UMENG_CHANNEL 乐视 leshe
11. UMENG_CHANNEL 豌豆荚 wanjia
12. UMENG_CHANNEL 阿里应用分发 ali
```

签名文件导入后往放脚本的目录里复制一个，不然会报错

配置nginx 文件下载，或者ftp

安装 vsftp 匿名登录，即开即用，根本不用配置，防火墙打开就行了

nginx 配置

```
1.  server {
2.      listen      80;
3.      server_name  192.168.0.185
4.      charset utf-8;
5.      root        /var/ftp/pub;
6.
7.      location / {
8.          autoindex on;
9.          autoindex_exact_size off;
10.         autoindex_localtime on;
11.     }
12. }
```

钉钉配置机器人，首先你要是群主，你能把机器人加进去



然后复制机器人的url 放到脚本里的robot_url里
可以构建了

