

- 搭建开发环境
- [安装依赖](#)
 - [Android 开发环境](#)
 - [安装 Android Studio](#)
 - [安装Android SDK](#)
 - [配置 ANDROID_HOME 环境变量](#)
 - [把 platform-tools 目录添加到环境变量 Path 中](#)
 - [打包APK](#)
 - [第一步 下载项目](#)
 - [第二步 生成一个签名密钥](#)
 - [第三步](#)
 - [第四步](#)
 - [第五步](#)

搭建开发环境

安装依赖：

必须安装的依赖有：Node、React Native 命令行工具、Python2 以及 JDK 和 Android Studio。
虽然你可以使用任何编辑器来开发应用（编写 js 代码），但你仍然必须安装 Android Studio 来获得编译 Android 应用所需的工具和环境。

Node, Python2, JDK

注意Node 的版本必须大于等于 10，Python 的版本必须为 2.x（不支持 3.x），而 JDK 的版本必须是 1.8（目前不支持 1.9 及更高版本）。安装完 Node 后建议设置 npm 镜像以加速后面的过程（或使用科学上网工具）。

```
npm config set registry https://registry.npm.taobao.org --global
npm config set disturl https://npm.taobao.org/dist --global
```

Yarn、React Native 的命令行工具（react-native-cli）

Yarn是 Facebook 提供的替代 npm 的工具，可以加速 node 模块的下载。React Native 的命令行工具用于执行创建、初始化、更新项目、运行打包服务（packager）等任务。

```
npm install -g yarn react-native-cli
```

安装完 yarn 后同理也要设置镜像源：

```
yarn config set registry https://registry.npm.taobao.org --global
yarn config set disturl https://npm.taobao.org/dist --global
```

安装完 yarn 之后就可以用 yarn 代替 npm 了，例如用yarn代替npm install命令，用yarn add 某第三方库名代替npm install 某第三方库名。

Android 开发环境

请注意！！国内用户必须必须有稳定的翻墙工具，否则在下载、安装、配置过程中会不断遭遇链接超时或断开，无法进行开发工作。某些翻墙工具可能只提供浏览器的代理功能，或只针对特定网站代理等等，请自行研究配置或更换其他软件。总之如果报错中出现有网址，那么 99% 就是无法正常翻墙。

安装 Android Studio

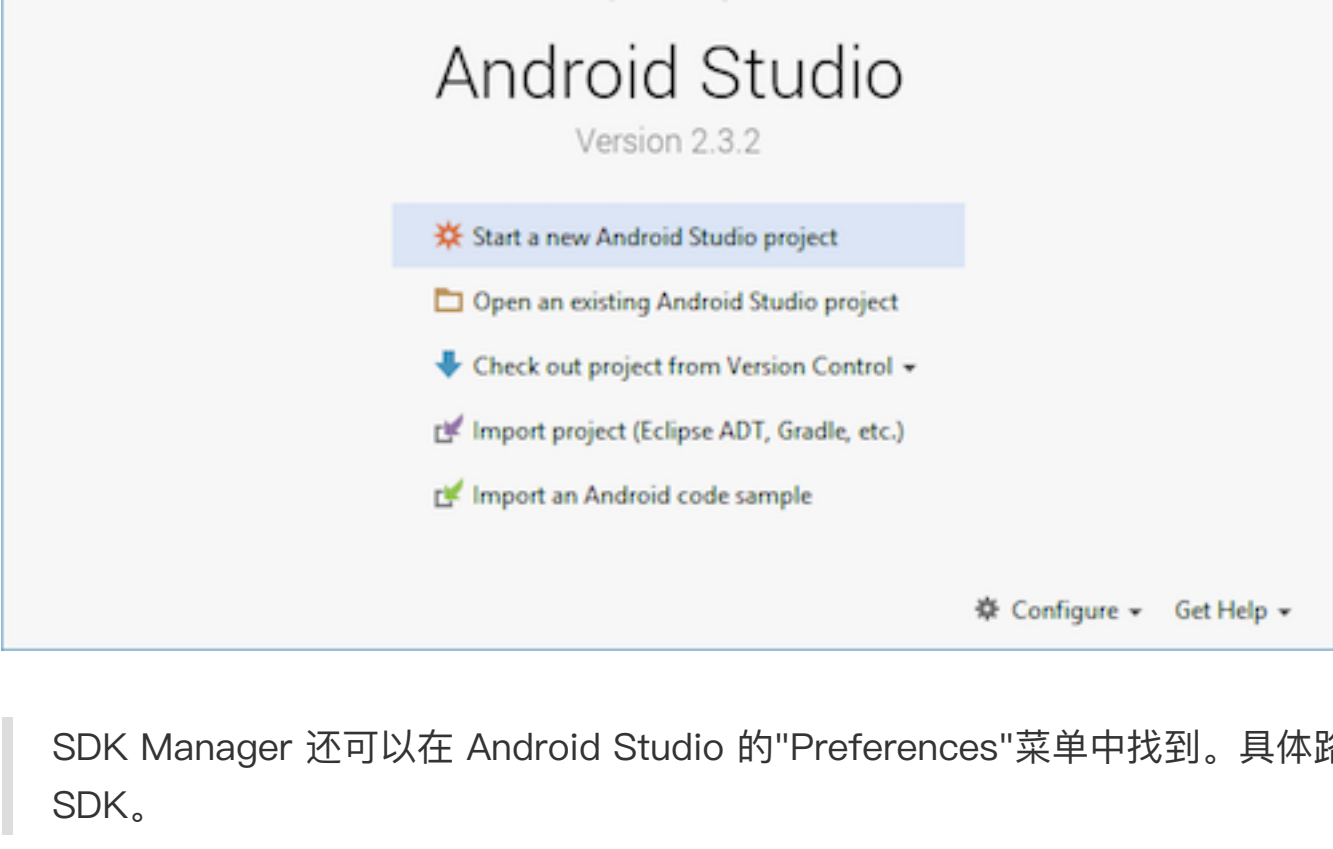
首先下载和安装 Android Studio，国内用户可能无法打开官方链接，请自行使用搜索引擎搜索可用的下载链接。安装界面中选择"Custom"选项，确保选中了以下几项：

- Android SDK
- Android SDK Platform
- Performance (Intel ® HAXM) (AMD 处理器看这里)
- Android Virtual Device

安装Android SDK

ndroid Studio 默认会安装最新版本的 Android SDK。目前编译 React Native 应用需要的是Android 9 (Pie)版本的 SDK（注意 SDK 版本不等于终端系统版本，RN 目前支持 android4.1 以上设备）。

你可以在 Android Studio 的 SDK Manager 中选择安装各版本的 SDK。你可以在 Android Studio 的欢迎界面中找到 SDK Manager。点击"Configure"，然后就能看到"SDK Manager"。



SDK Manager 还可以在 Android Studio 的"Preferences"菜单中找到。具体路径是Appearance & Behavior → System Settings → Android SDK。

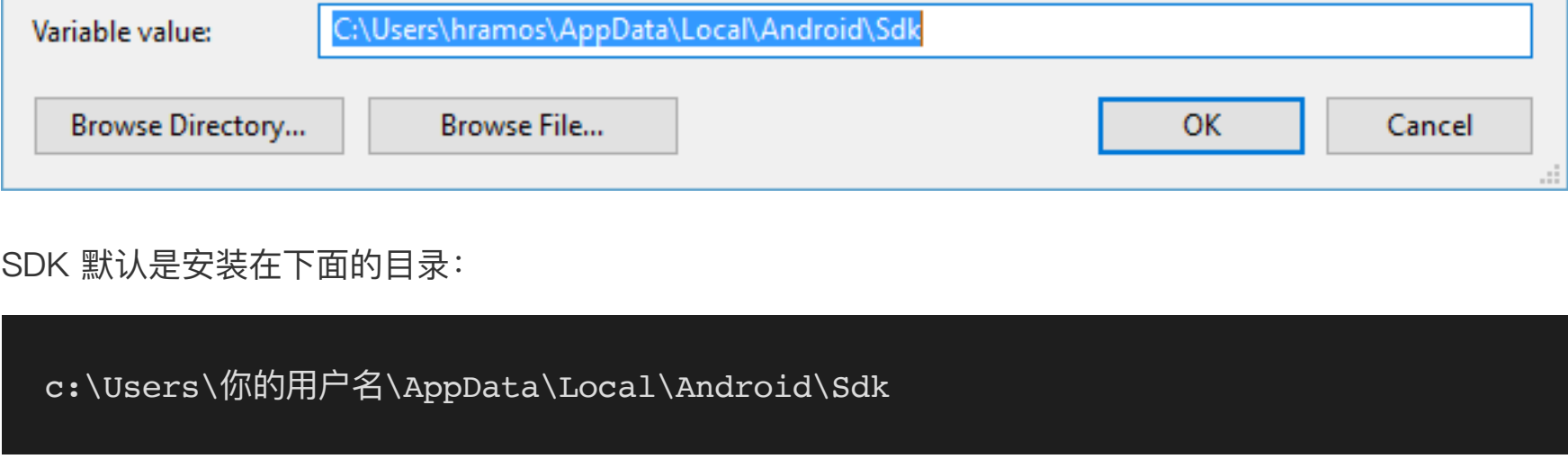
在 SDK Manager 中选择"SDK Platforms"选项卡，然后在右下角勾选"Show Package Details"。展开Android 9 (Pie)选项，确保勾选了下面这些组件（重申你必须使用稳定的翻墙工具，否则可能都看不到这个界面）：

- Android SDK Platform 28
- Intel x86 Atom_64 System Image（官方模拟器镜像文件，使用非官方模拟器不需要安装此组件）

然后点击"SDK Tools"选项卡，同样勾中右下角的"Show Package Details"。展开"Android SDK Build-Tools"选项，确保选中了 React Native 所必须的28.0.3版本。你可以同时安装多个其他版本。

最后点击"Apply"来下载和安装这些组件。

配置 ANDROID_HOME 环境变量



SDK 默认是安装在下面的目录：

```
c:\Users\你的用户名\AppData\Local\Android\Sdk
```

你可以在 Android Studio 的"Preferences"菜单中查看 SDK 的真实路径，具体是Appearance & Behavior → System Settings → Android SDK。

你需要关闭现有的命令提示窗口然后重新打开，这样新的环境变量才能生效。

把 platform-tools 目录添加到环境变量 Path 中

此目录的默认路径为：

```
c:\Users\你的用户名\AppData\Local\Android\Sdk\platform-tools
```

打包APK

第一步 下载项目

去github下载移动端项目：<https://github.com/jitsi/jitsi-meet>

第二步 生成一个签名密钥

命令行输入：

```
keytool -genkey -v -keystore my-release-key.keystore -alias my-key-alias -keyalg RSA -keysize 2048 -validity 10000
```

其中红线框部分：前者是即将生成的密钥库文件，后者是其别称

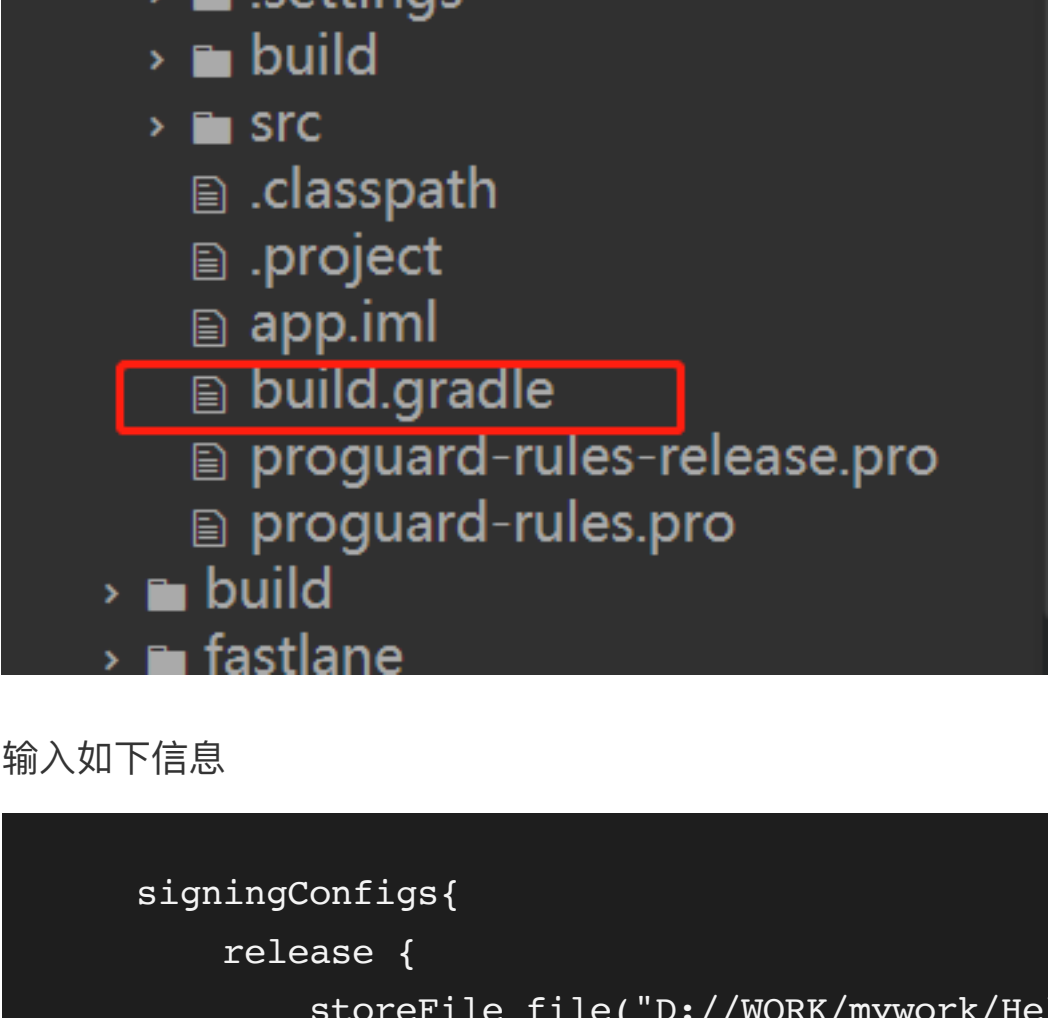
```
D:\WORK\mywork\Hello>keytool -genkey -v -keystore my-release-key.keystore -alias my-key-alias -keyalg
RSA -keysize 2048 -validity 10000
Picked up _JAVA_OPTIONS: -Xmx512M
输入密钥库口令：
再次输入新口令：
您的名字与姓氏是什么？
[Unknown]:
```

然后跟着提示一步一步走,最后它会生成一个叫做my-release-key.keystore的密钥库文件

	2019/9/17 0:55	ICO 图片文件	4 KB
	2019/9/17 0:55	JavaScript 文件	1 KB
	2019/9/17 0:55	Chrome HTML D...	8 KB
	2019/9/17 0:55	JavaScript 文件	1 KB
	2019/9/17 0:55	JavaScript 文件	8 KB
	2019/9/17 0:55	文件	13 KB
	2019/9/17 0:55	JavaScript 文件	1 KB
	2019/9/17 0:55	文件	3 KB
	2019/9/17 0:55	JavaScript 文件	1 KB
	2019/9/17 20:15	KEYSTORE 文件	3 KB
	2019/9/17 0:55	JSON File	5 KB
	2019/9/17 19:23	JSON File	782 KB
	2019/9/17 0:55	Chrome HTML D...	0 KB
	2019/9/17 0:55	MD 文件	6 KB
	2019/9/17 0:55	Chrome HTML D...	1 KB

第三步

打开下图所示位置的build.gradle文件



输入如下信息

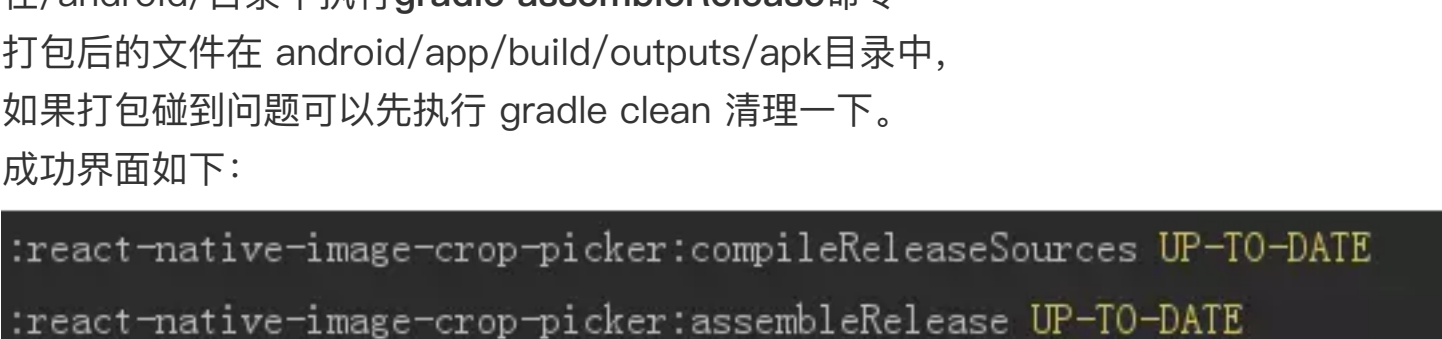
```
signingConfigs{
    release {
        storeFile file("D://WORK/mywork/Hello/my-release-key.keystore") //根据自己的文件位置写
        storePassword "123456"
        keyAlias "my-key-alias"
        keyPassword "123456"
    }
}
```

然后在buildTypes内添加一条语句如下

```
buildTypes {
    release {
        minifyEnabled enableProguardInReleaseBuilds
        proguardFiles getDefaultProguardFile("proguard-android.txt"), "proguard-rules.pro"
        signingConfig signingConfigs.release //添加这句话引用签名配置
    }
}
```

第四步

1. 命令行运行：cnpm install
2. 点击这个小图标，build



第五步

在/android/目录中执行gradle assembleRelease命令

打包后的文件在 android/app/build/outputs/apk目录中，

如果打包碰到问题可以先执行 gradle clean 清理一下。

成功界面如下：

```
:react-native-image-crop-picker:compileReleaseSources UP-TO-DATE
:react-native-image-crop-picker:assembleRelease UP-TO-DATE
:react-native-toast:compileReleaseSources UP-TO-DATE
:react-native-toast:assembleRelease UP-TO-DATE

BUILD SUCCESSFUL

Total time: 4 mins 0.232 secs
```