1. CCFileUtils.cpp



encryptFile：apk分解生成的加密文件，顺序要对。

encryptJarFile：java代码生成的加密文件。

GAME\_JAR\_ID 默认可以不改。(encryptJarFile解密出来的jar包)

encryptSize：apk 字节大小

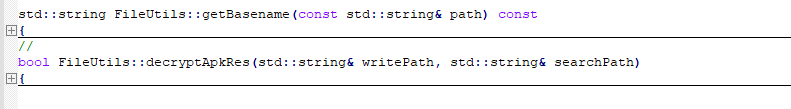
encryptkey：解密key

plugFileName:生成的解密的路径

plugZipName：java代码中对应的zip文件名

BUFFER\_SIZE：用默认即可。

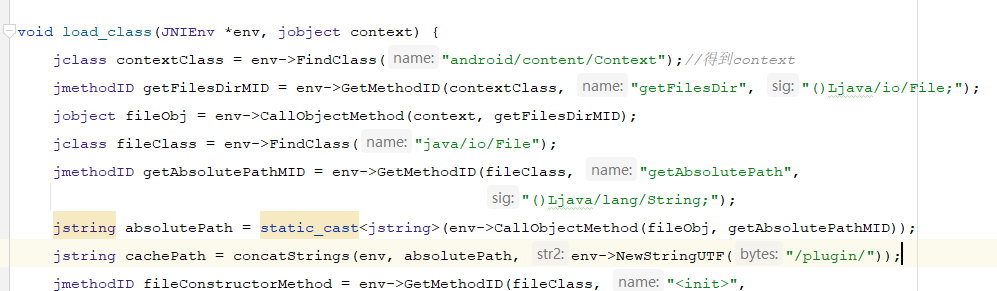
MAX\_FILENAME：用默认即可。

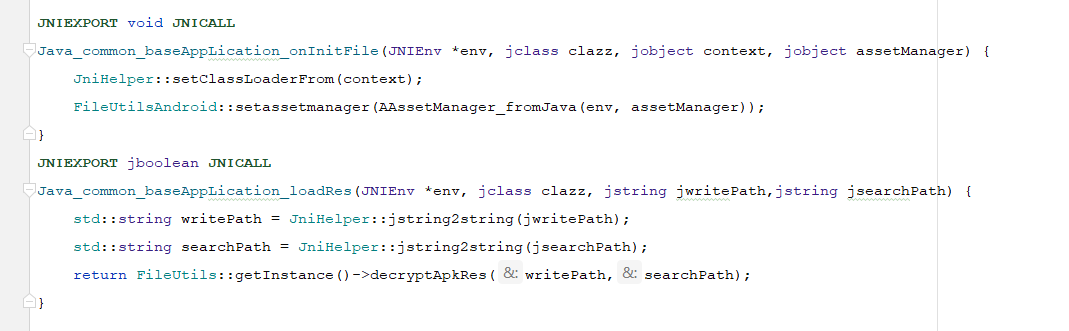
解密解压函数。

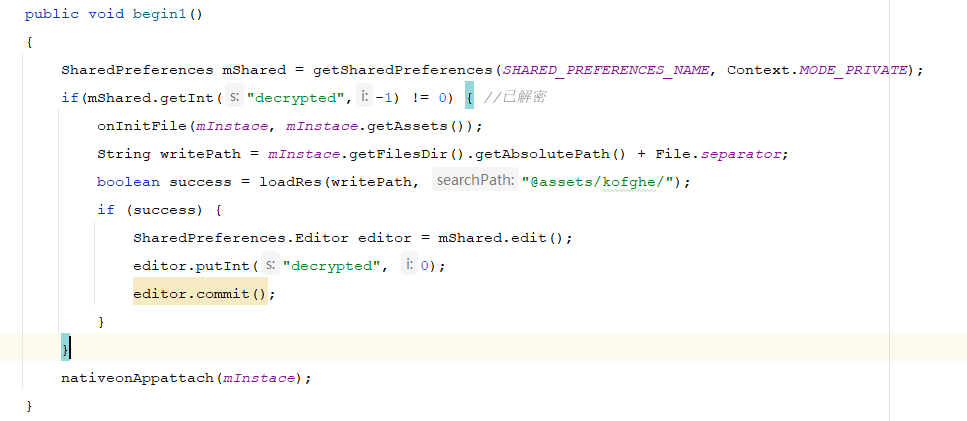
2．JniImp.cpp

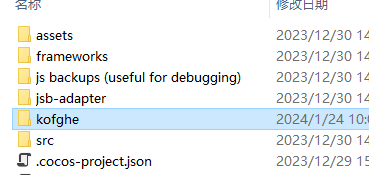


read\_file函数不需要调用了。

jar包的路径存放在plugin目录下，跟apk解密后的路径相同。所以此处也要更改为plugin

baseApplication中导出初始化函数跟解密函数。

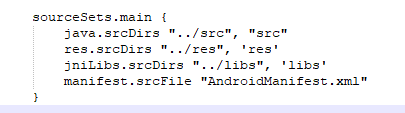
3.baseApplication.java 

在begin1中调用解密等函数。待解密资源保存到与assets目录。

如何待解密资源目录变更，则baseApplication.java 中 loadRes第二个参数也需要更改。如存放目录为assets/kofghe。则loadRes第二个参数为@assets/assets/kofghe/。

4.app/build.gradle

sourceSets.main去掉assets拷贝



android.applicationVariants.all 添加目录。如果目录在assets下，则不需要处理

