Android APK脱壳--腾讯乐固、360加固一键脱壳 - 简书

笔记本: gfx逆向
创建时间: 2020/1/3 20:47
URL: https://www.jianshu.com/p/138c9de2c987

Android APK脱壳--腾讯乐固、360加固一键脱壳

编码前线

9 2018.09.28 17:22:18 字数 478 阅读 59,944

概述

> 现在使用Proguard进行混淆的代码，也很容易被破解，所以就出现了加固工具，让反编译的难度更大。但是有了加固技术，就会有反加固技术，正所谓道高一尺魔高一丈。

经过加固后的apk，通过`dex2jar`反编译：

腾讯乐固：

legu.png

360加固：

360jiagu.png

从上面可以看出，经过加固后的apk，通过常规方法反编译无法获取到源码。

下载工具

脱壳工具FDex2

> 通过Hook ClassLoader的loadClass方法，反射调用getDex方法取得Dex(com.android.dex.Dex类对象)，在将里面的dex写出。

下载地址：

> 链接:https://pan.baidu.com/s/1smxtinr 密码:dk4v

VirtualXposed

> VirtualXposed：无需root手机即可使用xp框架。

下载地址：

> https://vxposed.com/

脱壳

Step1:

将`VirtualXposed`、`FDex2`和**需要脱壳的应用**都安装到手机上。

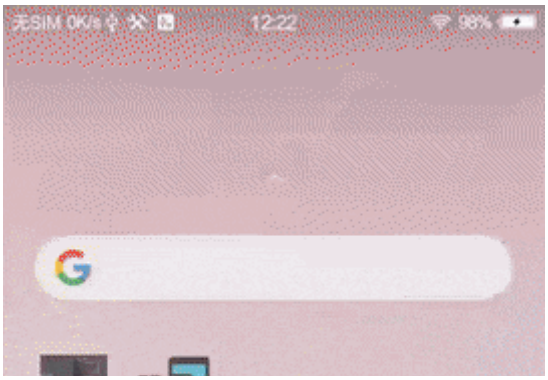
Step2:

启动`VirtualXposed`，并在`VirtualXposed`中安装`FDex2`：

vp-install-fdex2.gif

Step3:

在`VirtualXposed`中激活`FDex2`：



active-fdex2.gif

Step4:

在`VirtualXposed`中安装要脱壳的应用，方法和Step2一样。

Step5:

启动`VirtualXposed`中的`FDex2`，并配置要脱壳的应用。

fdex2-config.png

Step6:

在`VirtualXposed`中运行要脱壳的应用。

Step7:

脱壳后的dex文件：

shelling-dex.png

导出脱壳的dex文件：

root设备：

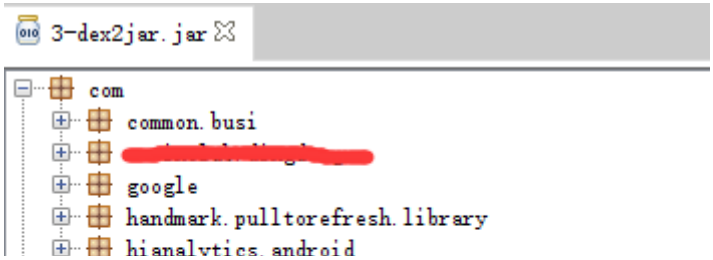
```
adb root
adb pull /data/user/0/iv.va.exposed/virtual/data/user/0/{packageName}   {电脑
```

未root设备：

在`VirtualXposed`中，设置-->高级设置-->文件管理，安装文件管理器，然后通过文件管理器进入到指定的目录，通过`分享`功能发到电脑上。

Step8：

通过`dex2jar`对 脱壳的dex进行反编译：



shelling-dex2jar.png

从上图就可以看到脱壳后的dex文件被成功的反编译。

FDex2核心代码MainHook

```
package com.ppma.xposed;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.lang.reflect.Method;

import de.robv.android.xposed.IXposedHookLoadPackage;
import de.robv.android.xposed.XC_MethodHook;
import de.robv.android.xposed.XSharedPreferences;
import de.robv.android.xposed.XposedBridge;
import de.robv.android.xposed.XposedHelpers;
import de.robv.android.xposed.callbacks.XC_LoadPackage;

public class MainHook implements IXposedHookLoadPackage {

    XSharedPreferences xsp;
    Class Dex;
    Method Dex_getBytes;
    Method getDex;
    String packagename;


    public void handleLoadPackage(XC_LoadPackage.LoadPackageParam lpparam) thr
        xsp = new XSharedPreferences("com.ppma.appinfo", "User");
        xsp.makeWorldReadable();
        xsp.reload();
        initRefect();
        packagename = xsp.getString("packagename", null);
        XposedBridge.log("设定包名: "+packagename);
        if ((!lpparam.packageName.equals(packagename))||packagename==null) {
            XposedBridge.log("当前程序包名与设定不一致或者包名为空");
            return;
        }
        XposedBridge.log("目标包名: "+lpparam.packageName);
        String str = "java.lang.ClassLoader";
```

```
        String str2 = "loadClass";

        XposedHelpers.findAndHookMethod(str, lpparam.classLoader, str2, String
            protected void afterHookedMethod(MethodHookParam param) throws Thr
                super.afterHookedMethod(param);
                Class cls = (Class) param.getResult();
                if (cls == null) {
                    //XposedBridge.log("cls == null");
                    return;
                }
                String name = cls.getName();
                XposedBridge.log("当前类名：" + name);
                byte[] bArr = (byte[]) Dex_getBytes.invoke(getDex.invoke(cls,
                if (bArr == null) {
                    XposedBridge.log("数据为空：返回");
                    return;
                }
                XposedBridge.log("开始写数据");
                String dex_path = "/data/data/" + packagename + "/" + packager
                XposedBridge.log(dex_path);
                File file = new File(dex_path);
                if (file.exists()) return;
                writeByte(bArr, file.getAbsolutePath());
            }
            } );
    }

    public void initRefect() {
        try {
            Dex = Class.forName("com.android.dex.Dex");
            Dex_getBytes = Dex.getDeclaredMethod("getBytes", new Class[0]);
            getDex = Class.forName("java.lang.Class").getDeclaredMethod("getDe
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        } catch (NoSuchMethodException e) {
            e.printStackTrace();
        }

    }

    public  void writeByte(byte[] bArr, String str) {
        try {
            OutputStream outputStream = new FileOutputStream(str);
            outputStream.write(bArr);
            outputStream.close();
        } catch (IOException e) {
            e.printStackTrace();
            XposedBridge.log("文件写出失败");
        }
    }
}
```

参考链接

1. 【手机端】腾讯乐固，360加固一键脱壳
2. 安卓xposed脱壳工具FDex2

编码前线

程序员都在关注的公众号

WeChat.jpg