番外实战篇1-某电影视全家桶

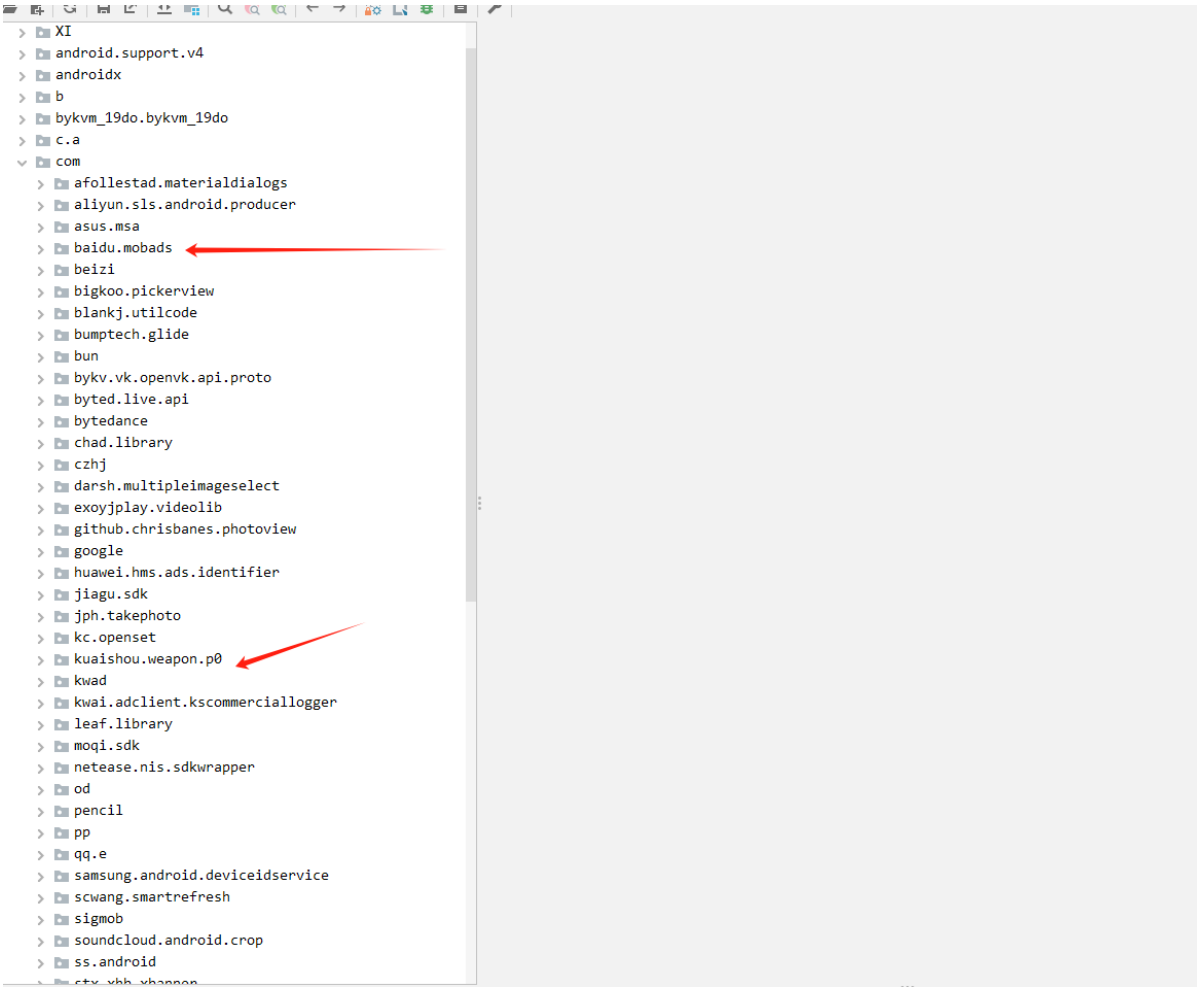
1、去除广告

关键字:initsdk、loadad、initad等

广告关键词	厂商	文档
com.qq.e.ads	腾讯优量汇广告	https://developers.adnet.qq.com/doc/android/union/union_splash
CSJAD、TTAdSdk、bytedance、pangolin	穿山甲广告	https://www.csjplatform.com/supportcenter/5395
ADMob、google.ads	谷歌广告	https://developers.google.com/admob/android/app-open?hl=zh-cn#extend
TorchAd	360广告	https://easydoc.soft.360.cn/doc?project=186589faed863b0a24f15f9bcbafd5c7&doc=2cbbbe19c5cb90f5e7a41c7037b0029a&config=title_menu_toc
kwad	快手广告	https://u.kuaishou.com/home/help/detail/1334/1370/1310
baidu.mobads	百度广告	http://bce.ssp.baidu.com/mssp/sdk/BaiduMobAds_MSSP_bd_SDK_android_v5.1.pdf
MimoSdk	米盟广告	https://t5.a.market.xiaomi.com/download/AdCenter/0d3a369516ee146e8a9d5c290985939da4624fe0a/AdCenter0d3a369516ee146e8a9d5c290985939da4624fe0a.html
sigmob.sdk	sigmob广告	https://doc.sigmob.com/#/Sigmob%E4%BD%BF%E7%94%A8%E6%8C%87%E5%8D%97/SDK%E9%9B%86%E6%88%90%E8%AF%B4%E6%98%8E/Android/SDK%E6%8E%A5%E5%85%A5%E9%85%8D%E7%BD%AE/
TradPlus	TradPlus聚合广告	https://service.cocos.com/document/zh/tradplusad.html

通过免广告关键字来实现部分广告的去除

第一步、查看代码



根据不同的广告可以进入对应的开发文档

https://developers.adnet.qq.com/doc/android/union/union_splash

速访问。管理书签...

快捷导航

首页
- 概述
- 背景
- 广告位创建流程
- 支持广告类型

SDK嵌入
- SDK修订历史
- 导入示例工程
- 嵌入广告SDK

接入代码
- 开屏广告
- 激励广告
- 插屏广告
- 横幅(Banner)广告
- 信息流/详情页插入/视频贴片...
- 开发者自渲染

实时竞价

流量聚合

其他内容
- 断点续安装提示
- 断点续安装或激活弹窗提示
- 应用下载弹窗确认
- 进入应用下载任务列表
- 个性化推荐广告
- 广告去重

问题排查
- debug工具

SplashADListener adListener)	posId（广告位 ID）、adListener（广告状态监听器）。
SplashAD(Activity activity, String posId, SplashADListener adListener, int fetchDelay)	构造方法，这个方法可以通过传入 fetchDelay 参数，设置开屏广告从请求到展示所花的最大时长（并不是指广告曝光时长），取值范围为[1500, 5000]ms。如果需要使用默认值，可以调用上一个构造方法，或者给 fetchDelay 设为0。
preLoad()	预加载广告，调用后开屏广告会尝试进行广告数据和图片/视频素材的预加载（若视频广告的视频素材未能提前预加载，在得到播放机会时将退化为图片广告进行播放），预加载的广告在得到展示机会时可以跳过广告素材下载的步骤而直接展示，该接口有频次控制，默认最小间隔2小时及以上调用才会生效，建议一天执行一次。
fetchAdOnly()	拉取广告，配合showAd使用，实现和fetchAndShowIn相同的功能。
showAd(ViewGroup container)	展示广告，配合fetchAdOnly使用，实现和fetchAndShowIn相同的功能。
fetchFullScreenAdOnly()	拉取全屏广告，配合showFullScreenAd使用，实现和fetchFullScreenAndShowIn相同的功能。
showFullScreenAd(ViewGroup container)	配合fetchFullScreenAdOnly使用，实现和fetchFullScreenAndShowIn相同的功能。
isValid()	广告是否有效，无效广告将无法展示
setDeveloperLogo(int logoRes)	设置开发者 logo，仅在全屏广告时有效，参数要求是 Resource ID。
setDeveloperLogo(byte[] logoData)	设置开发者 logo，仅在全屏广告时有效，参数要求是能转为 Bitmap 的字节数组。
getECPMLevel()	获取本条广告实时的eCMP价格层级标签，每个层级标签对应线下预定的eCPM价格范围，成功返回一个包含数字的字符串，比如“1”，无权限调用该接口或后台异常会返回空字符串
getECPM()	获取本条广告实时的eCPM价格，单位是分
getExtraInfo()	（4.410.1280新增）获取本条广告一些额外信息

SplashADListener

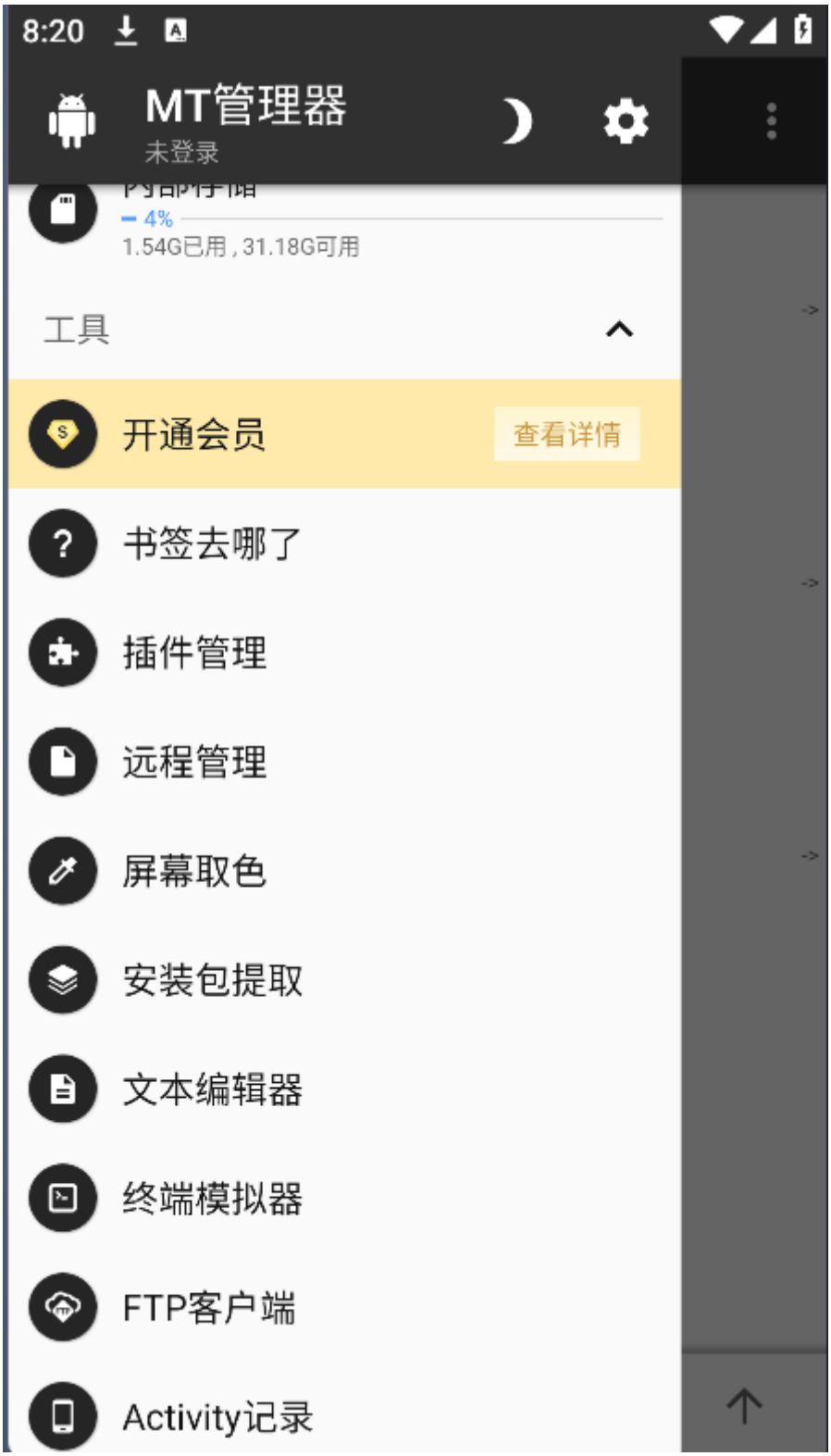
- **com.qq.e.ads.splash.SplashADListener:**

  开屏广告回调接口

方法名	方法介绍
onNoAD(AdError error)	广告加载或展示过程中出错，AdError中包含了错误码和错误描述，具体错误码内容可参考错误码部分
onADDismissed()	广告关闭时调用，可能是用户关闭或者展示时间到。此时一般需要跳过开屏的 Activity，进入应用内容页面
onADPresent()	广告成功展示时调用，成功展示不等于有效展示（比如广告容器高度不够）
onADClicked()	广告被点击时调用，不代表满足计费条件（如点击时网络异常）
onADExposure()	广告曝光时调用

如果是多个平台的聚合广告，当加载广告会进入启动类初始化，可以搜索关键字:initsdk、loadad、initad等

打开mt管理器的activity记录



打开待测试的app，然后再返回mt管理器

紫电视频

com.pencil.yereen.MainActivity > b.s.g.b

根据对应的类找到代码

```
saiclassify
saidownload
saifeedback
saihomepage
sailogin
sailookcategory
saimine
saiphotoview
saipush
sairank
saisplash
  SaiSplashActivity
    a
    b
    f12382g Handler
    g(SaiPrivateEvent) void
    i(SaiAppInitEvent) void
    initContentView(Bundle) int
    initData() void
    initParam() void
    initVariableId() int
    k() void
    m() void
    mo416initViewModel() SaiSplashViewModel
    n(SaiSplashActivity$b) void
    onDestroy() void
  SaiSplashAdActivity
  SaiSplashViewModel
saiuserinfo
saivideodetail
saivideosearch
saiwakeup
saiwebsite
toolbar
MainActivity
MainViewModel
DataBinderMapperImpl
R$anim
R$animator
R$array
R$attr
R$bool
R$color
R$dimen
R$drawable
R$id
R$integer
R$interpolator
R$layout
R$menu
R$mipmap
R$plurals
R$raw
R$string
R$style
R$styleable
R$xml
SaiScaleTransitionPagerTitleView
pp
qq.e
```

```
SaiSplashActivity
/* Loaded from: classes2.dex */
public static /* synthetic */ class a {
    public static final /* synthetic */ int[] a;

    static {
1       int[] iArr = new int[b.values().length];
        a = iArr;
        try {
            iArr[b.NONET.ordinal()] = 1;
        } catch (NoSuchFieldError unused) {
        }
        try {
            a[b.DELAY.ordinal()] = 2;
        } catch (NoSuchFieldError unused2) {
        }
        try {
            a[b.FIRSTLAUNCH.ordinal()] = 3;
        } catch (NoSuchFieldError unused3) {
        }
    }
}

/* Loaded from: classes2.dex */
public enum b {
    NONET,
    DELAY,
    FIRSTLAUNCH
}

/* JADX INFO: Access modifiers changed from: private */
/* renamed from: f */
public /* synthetic */ void g(SaiPrivateEvent saiPrivateEvent) throws Exception {
1       initData();
}

/* JADX INFO: Access modifiers changed from: private */
/* renamed from: h */
public /* synthetic */ void i(SaiAppInitEvent saiAppInitEvent) throws Exception {
1       dismissDialog();
2       n(b.FIRSTLAUNCH);
}

/* JADX INFO: Access modifiers changed from: private */
/* renamed from: j */
public /* synthetic */ void k() {
1       startActivity(MainActivity.class);
2       finish();
}

/* JADX INFO: Access modifiers changed from: private */
/* renamed from: l */
public /* synthetic */ void m() {
1       startActivity(MainActivity.class);
2       finish();
}

@Override // com.pencil.base.BaseAt
public int initContentView(Bundle bundle) {
    return R.layout.sai_acitivity_splash;
}

@Override // com.pencil.base.BaseAt
@SuppressLint({"MissingPermission"})
public void initData() {
1       super.initData();
2       if (!v.c().b(SaiSPKey.AGREE_PRIVATE, false)) {
6           ((SaiSplashViewModel) this.viewModel).b(b.s.c.b.a().c(SaiPrivateEvent.class).observeOn(AndroidSchedulers.mainThread()).subscribe(new Consumer() { // from class: b.l.
                @Override // io.reactivex.functions.Consumer
                public final void accept(Object obj) {
                    SaiSplashActivity.this.g((SaiPrivateEvent) obj);
                }
            }));
7           a0.a.h(this);
            return;
        }
        try {
8           s.a.d();
```

分析代码

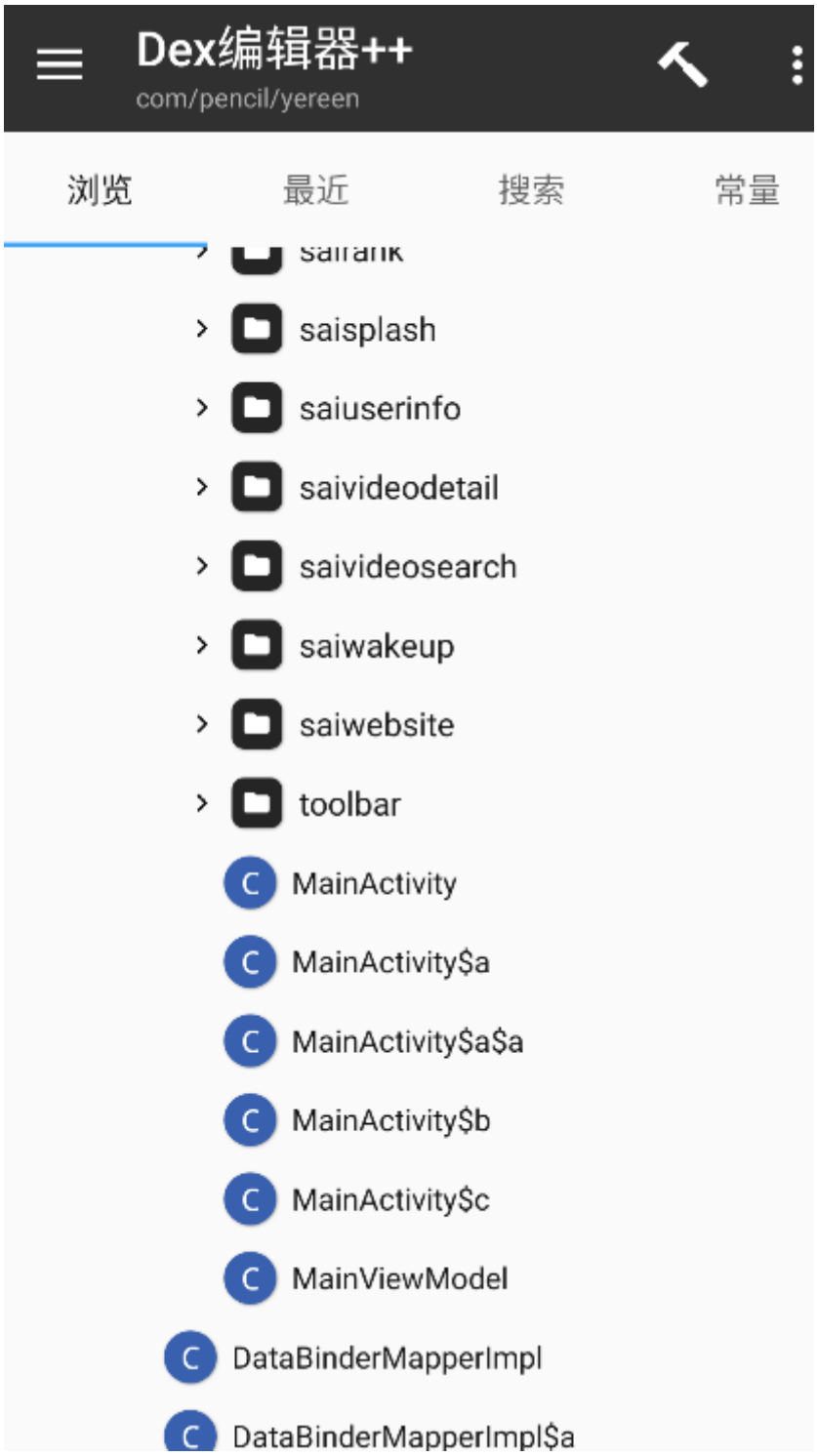
```
public void c(SaiAdResp.AdBean adBean, int i2, int i3) {
1       HashMap hashMap = new HashMap();
2       hashMap.put("is_click", Integer.valueOf(i2));
3       hashMap.put("tag_id", adBean.getTag_id());
4       hashMap.put("count", 1);
        if (i3 != -1) {
5           hashMap.put(RewardItem.KEY_ERROR_CODE, Integer.valueOf(i3));
        }
        if (i3 == 411) {
6           b.c.a.b.v.c().s(SaiSPKey.phonePermisson, true);
7           e0.b("==============>>>> 触发授权弹窗");
        }
        e0.b("==============>>>> apiBack " + adBean.getTag_id() + "-->is_click " + i2 + "-->> errorCode" + i3);
10      b.l.f.f.u().a(hashMap).subscribe((Subscriber<? super SaiBaseBean>) new a());
    }

    public void d() {
1       Constant constant = Constant.INSTANCE;
        n(constant.GroMore);
2       if (n(constant.GDT)) {
3           u.a.a();
        }
4       if (n(constant.CSJ)) {
5           t.a.d();
        }
6       if (n(constant.TaoDou)) {
7           w.a.a();
        }
8       if (n(constant.OsetSdk)) {
9           v.a.a();
        }
    }

    public void k(final Activity activity, final ViewGroup viewGroup, final String str, boolean z) {
        SaiAdResp.AdBean e2;
1       if (this.f4034b < 2 && (e2 = c0.a.e(str, "", false)) != null) {
            this.f4034b++;
4           String valueOf = String.valueOf(e2.getSdk_id());
5           Constant constant = Constant.INSTANCE;
            if (valueOf.equals(constant.CSJ)) {
6               t.a.f(activity, e2, str, viewGroup, new b() { // from class: b.l.g.a
                    @Override // b.l.g.s.b
                    public final void a(Boolean bool, View view) {
                        s.this.f(viewGroup, activity, str, bool, view);
                    }
                });
7           } else if (valueOf.equals(constant.GDT)) {
8               u.a.c(activity, e2, str, new b() { // from class: b.l.g.b
                    @Override // b.l.g.s.b
                    public final void a(Boolean bool, View view) {
```

可以通过注释代码或者置空来防止广告

mt导出app，并dex编辑器打开dex文件

文件夹: 9 文件: 8
..
about_files
81-01-01 01:01
assets
81-01-01 01:01
acct
25-03-23 08:09
bin
25-02-24 19:56
cache
25-03-23 08:09
MultiDex
classes.dex
classes2.dex
classes3.dex
全选
取消
确定
81-01-01 01:01
AndroidManifest.xml
81-01-01 01:01 60.87K
classes.dex
81-01-01 01:01 8.32M
classes2.dex
24-11-14 17:32
mnt
25-03-23 08:09
odm
24-11-14 17:32
oem

找到对应目录代码



先在jadx全局找到d()方法，再去mt管理器比较方便

```
.method public d()V
    .registers 3

    .line 1
    sget-object v0, Lcom/pencil/saibeans/Constant;->INSTANCE:Lcom/pencil/saibeans/Constant;

    iget-object v1, v0, Lcom/pencil/saibeans/Constant;->GroMore:Ljava/lang/String;

    invoke-virtual {p0, v1}, Lb/l/g/s;->n(Ljava/lang/String;)Z

    .line 2
    iget-object v1, v0, Lcom/pencil/saibeans/Constant;->GDT:Ljava/lang/String;

    invoke-virtual {p0, v1}, Lb/l/g/s;->n(Ljava/lang/String;)Z

    move-result v1

    if-eqz v1, :cond_14

    .line 3
    sget-object v1, Lb/l/g/u;->a:Lb/l/g/u;

    invoke-virtual {v1}, Lb/l/g/u;->a()V

    .line 4
    :cond_14
    iget-object v1, v0, Lcom/pencil/saibeans/Constant;->CSJ:Ljava/lang/String;

    invoke-virtual {p0, v1}, Lb/l/g/s;->n(Ljava/lang/String;)Z
```

可以经该方法代码注释掉或者增加

修改前

```
(Lb/l/g/s;)V

    .line 10
    invoke-virtual {p1, p2}, Lrx/Observable;->subscribe(Lrx/Subscriber;)Lrx/Subscription;

    return-void
.end method

.method public d()V
    .registers 3

    .line 1
    sget-object v0, Lcom/pencil/saibeans/Constant;->INSTANCE:Lcom/pencil/saibeans/Constant;

    iget-object v1, v0, Lcom/pencil/saibeans/Constant;->GroMore:Ljava/lang/String;

    invoke-virtual {p0, v1}, Lb/l/g/s;->n(Ljava/lang/String;)Z
```

修改后

```
.method public d()V
    .registers 3
return-void
    .line 1
```

return-void常见于void方法的结尾，用来结束方法的执行

然后保存退出（常规方法）

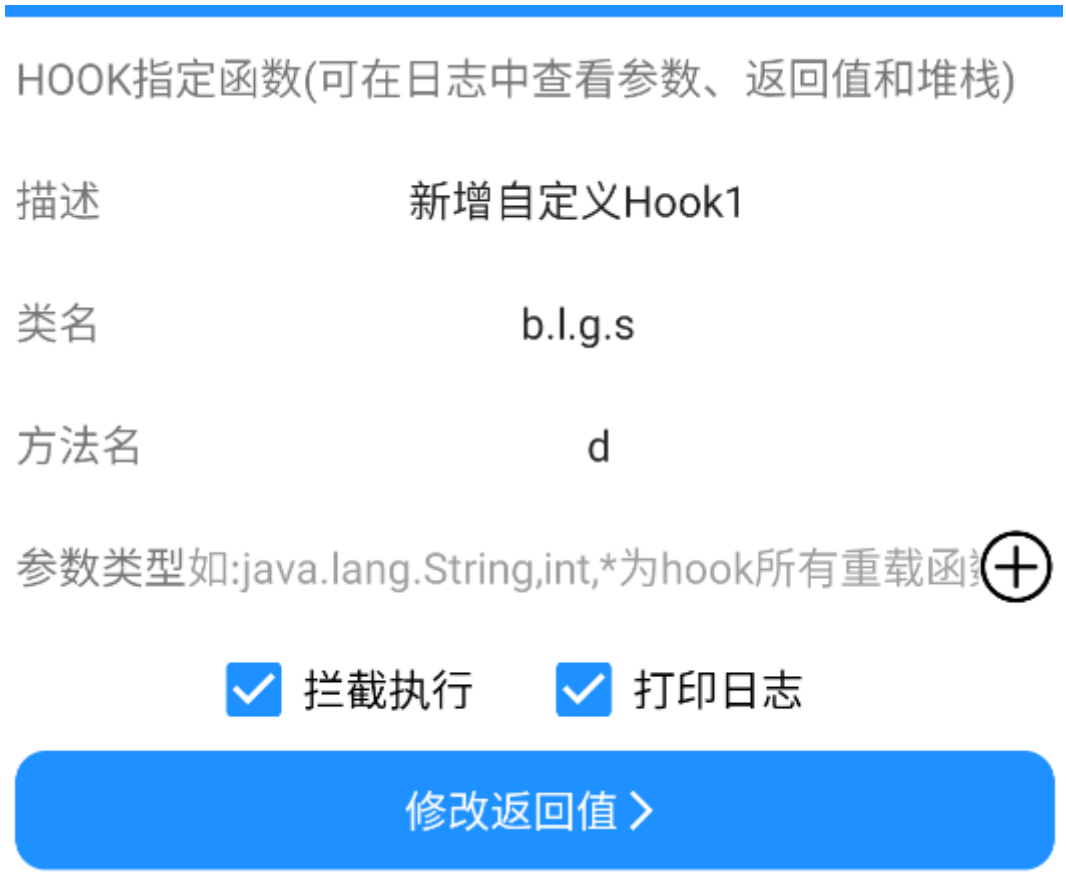
又因为该app有签名校验，所以我们可以在算法助手出进行hook

第一步复制一下方法签名

```
Lb/l/g/s;->d()V
```

算法助手--自定义hook

紫电视频
com.violetele.zdvod
SharedPreferences配置操作监听
读入配置文件
媒体功能
调用相机替换为选择照片
其他
Log捕获
Application监听
拦截应用退出
防止应用闪退
解除截屏录屏限制
读取应用签名监听
添加自定义Hook



保存

算法助手启动app，此时无广告了

斗罗大陆
推荐
内地
美剧
韩剧
华语
欧美
一身正气、心系苍生
正在热播
做自己的光
更新至26集
欢颜
18集全
安乐传
更新至27集
上次观看 做自己的光 1 00:53
点击 继续观看
首页
排行
分类
我的

其他

点击登录
VIP免广告特权：
邀请好友
推广记录
已累计邀请
已累计获得免广告

搜索免广告

```
MainActivity   s   SaiMineViewModel
            i3 += 10;
        }
        if (i2 >= 10) {
            i3 += 30;
        }
        if (i2 >= 50) {
            return "永久";
        }
        return "" + i3;
    }

    public void O(SaiUserInfo saiUserInfo, boolean z) {
1       g0.i(BaseApp.getInstance(), saiUserInfo);
        if (z) {
2           v.c().q(SaiSPKey.appToken, saiUserInfo.getToken());
        }
3       this.f12331e.set(saiUserInfo);
4       this.f12334h.setValue(saiUserInfo.getPic());
5       if (saiUserInfo.getLogin_type() == 1) {
6           this.f12333g.set("点击登录");
7           this.f12332f.set(Boolean.FALSE);
        } else {
8           this.f12333g.set(saiUserInfo.getNickname());
9           this.f12332f.set(Boolean.TRUE);
        }
10      ObservableField<String> observableField = this.f12335i;
        observableField.set("ID:" + saiUserInfo.getUser_id());
11      ObservableField<SpannableString> observableField2 = this.f12336j;
        observableField2.set(SaiAppUtils.d(saiUserInfo.getInvited_count() + "人"));
12      if (saiUserInfo.getInvited_count() > 0) {
13          ObservableField<SpannableString> observableField3 = this.k;
            observableField3.set(SaiAppUtils.d(N(saiUserInfo.getInvited_count()) + "天 "));
        } else {
14          this.k.set(SaiAppUtils.d("0天"));
        }
15      if (saiUserInfo.getFree_time() * 1000 > System.currentTimeMillis()) {
16          this.n.set(0);
17          this.m.set(b.s.f.d.a(Long.valueOf(saiUserInfo.getFree_time() * 1000)));
18          this.l.set("剩余免广告: ");
19          BaseApp.getInstance().setFreeTime(saiUserInfo.getFree_time() * 1000);
            return;
        }
20      this.l.set("邀请好友获得终身免广告特权");
21      this.n.set(8);
22      BaseApp.getInstance().setFreeTime(0L);
    }

    public void o(Dialog dialog, String str) {
2       b.l.f.f.u().G(str).subscribe((Subscriber<? super SaiBaseBean>) new c(dialog, str));
    }
}
```

进入getFreetime

```
}

public long getFree_time() {
    return this.free_time;
}

public int getGender() {
    return this.gender;
}
```

给其赋值，需要将设置的时间戳减去1970年1月1日的时间戳然后x365x1000

```
.method public getFree_time()J
    .registers 3

    .line 1
    iget-wide v0, p0, Lcom/pencil/saibeans/SaiUserInfo;->free_time:J

    const-wide v0, 0x32d57bf5e8L

    return-wide v0
.end method
```

可以通过算法助手 直接hook

2、去除签名校验

[吾爱破解安卓逆向入门教程《安卓逆向这档事》六、校验的N次方-签名校验对抗、PM代{过}{滤}理、IO重定向]

1.核心破解，免签名安装
2.一键去签名工具
[实战破解]白描-动态代{过}{滤}理Hook签名校验
3.手动分析签名校验

第一步、先对app进行重新签名，安装后无法启动

第二步、使用算法助手

紫电视频
com.violetele.zdvod

执行sql语句(execSQL)

SharedPreferences配置操作监听

读入配置文件

媒体功能

调用相机替换为选择照片

其他

Log捕获

Application监听

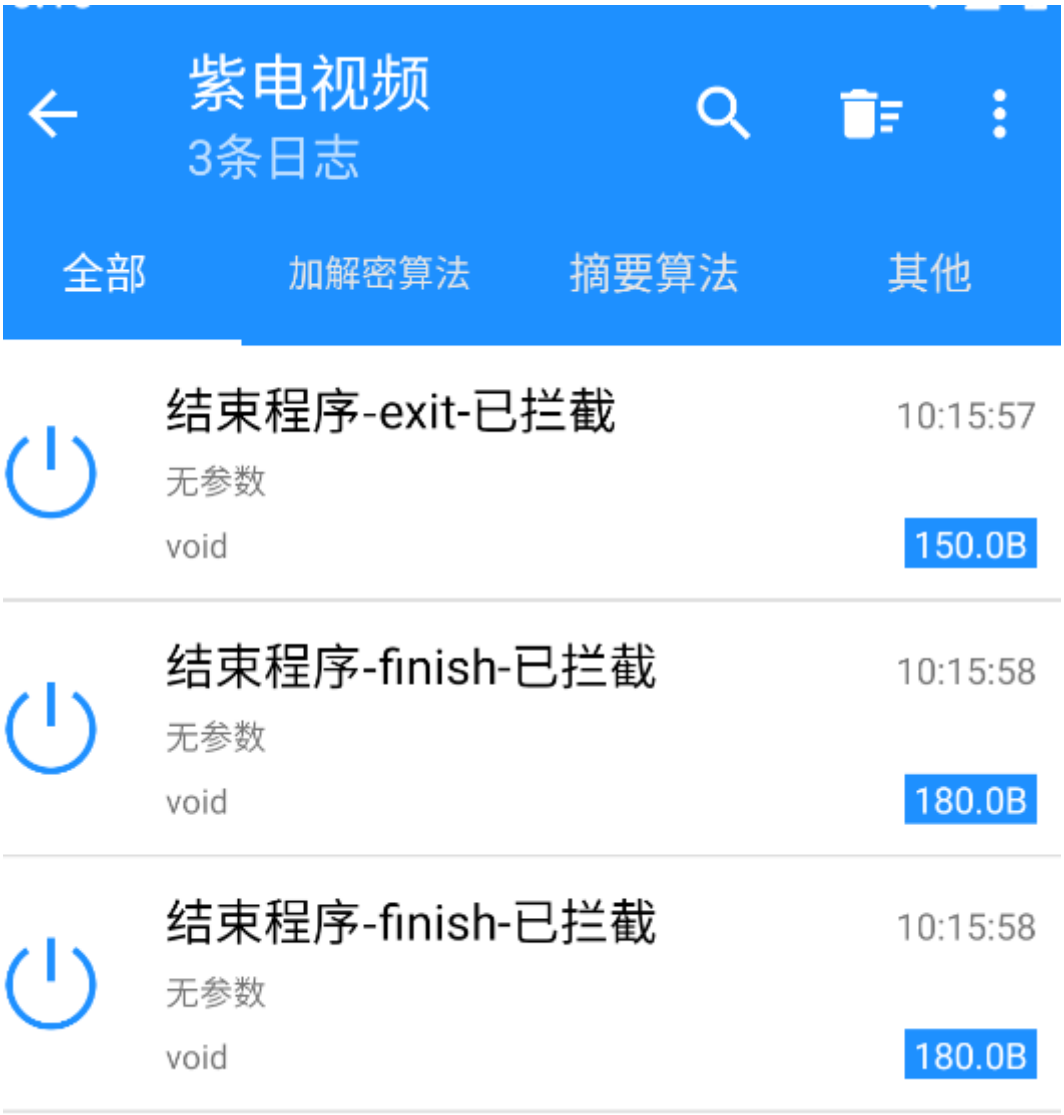
拦截应用退出

防止应用闪退

解除截屏录屏限制

读取应用签名监听

同样无法启动，算法查看日志


紫电视频
3条日志
全部
加解密算法
摘要算法
其他
结束程序-exit-已拦截
10:15:57
无参数
void
150.0B
结束程序-finish-已拦截
10:15:58
无参数
void
180.0B
结束程序-finish-已拦截
10:15:58
无参数
void
180.0B

查看日志

类名：java.lang.Runtime

日志名称：结束程序-exit-已拦截

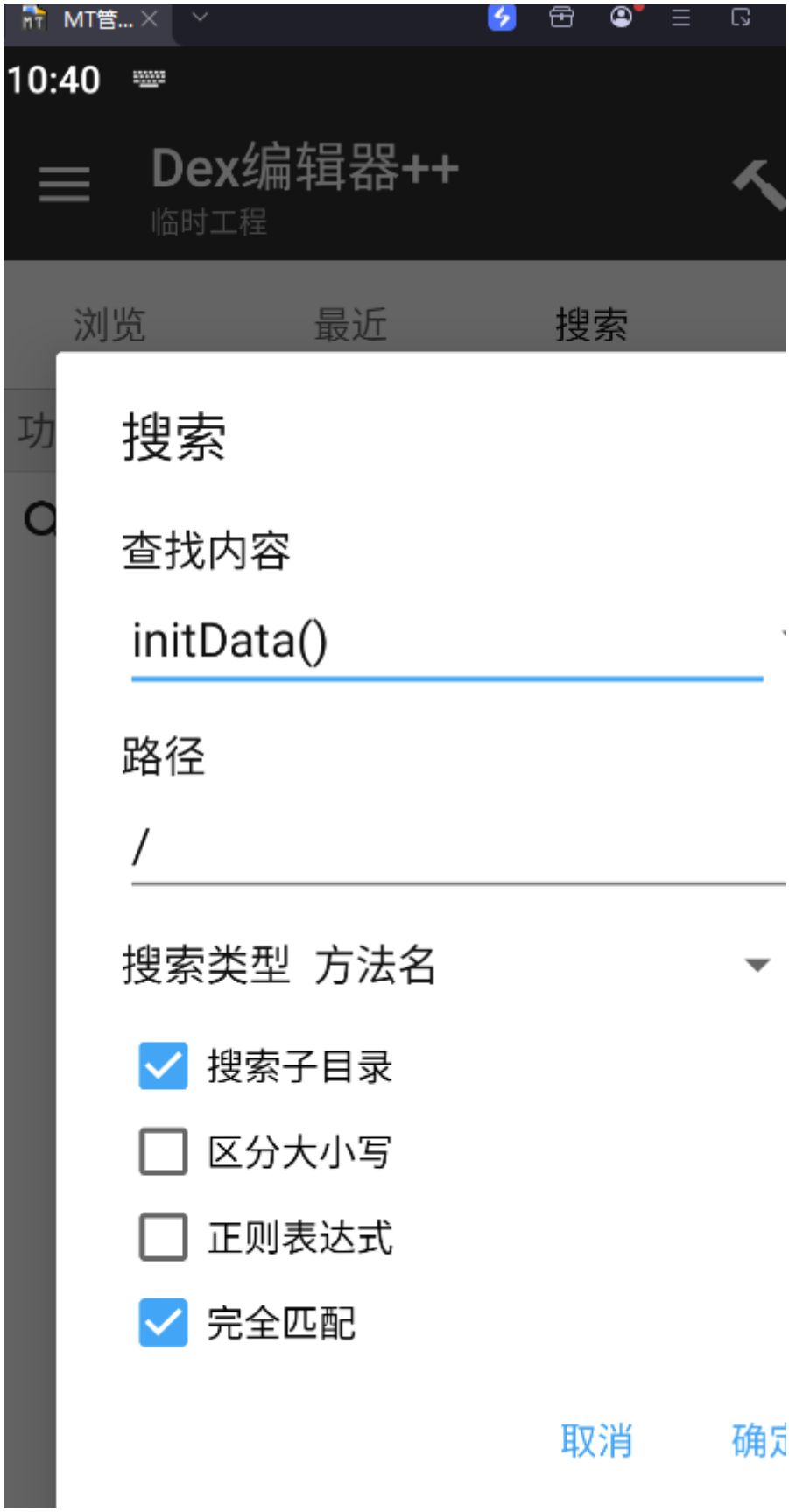
返回结果类型:void
返回结果值：void

调用堆栈：
at de.robv.android.xposed.XposedBridge$AdditionalHookInfo.callback(Unknown Source:79)
at org.lsposed.lspd.nativebridge.HookBridge.invokeOriginalMethod(Native Method)
at de.robv.android.xposed.XposedBridge$AdditionalHookInfo.callback(Unknown Source:110)
at com.pencil.yereen.saisplash.SaiSplashActivity.initData(SaiSplashActivity.java:36)
at com.pencil.base.BaseAt.onCreate(BaseAt.java:9)

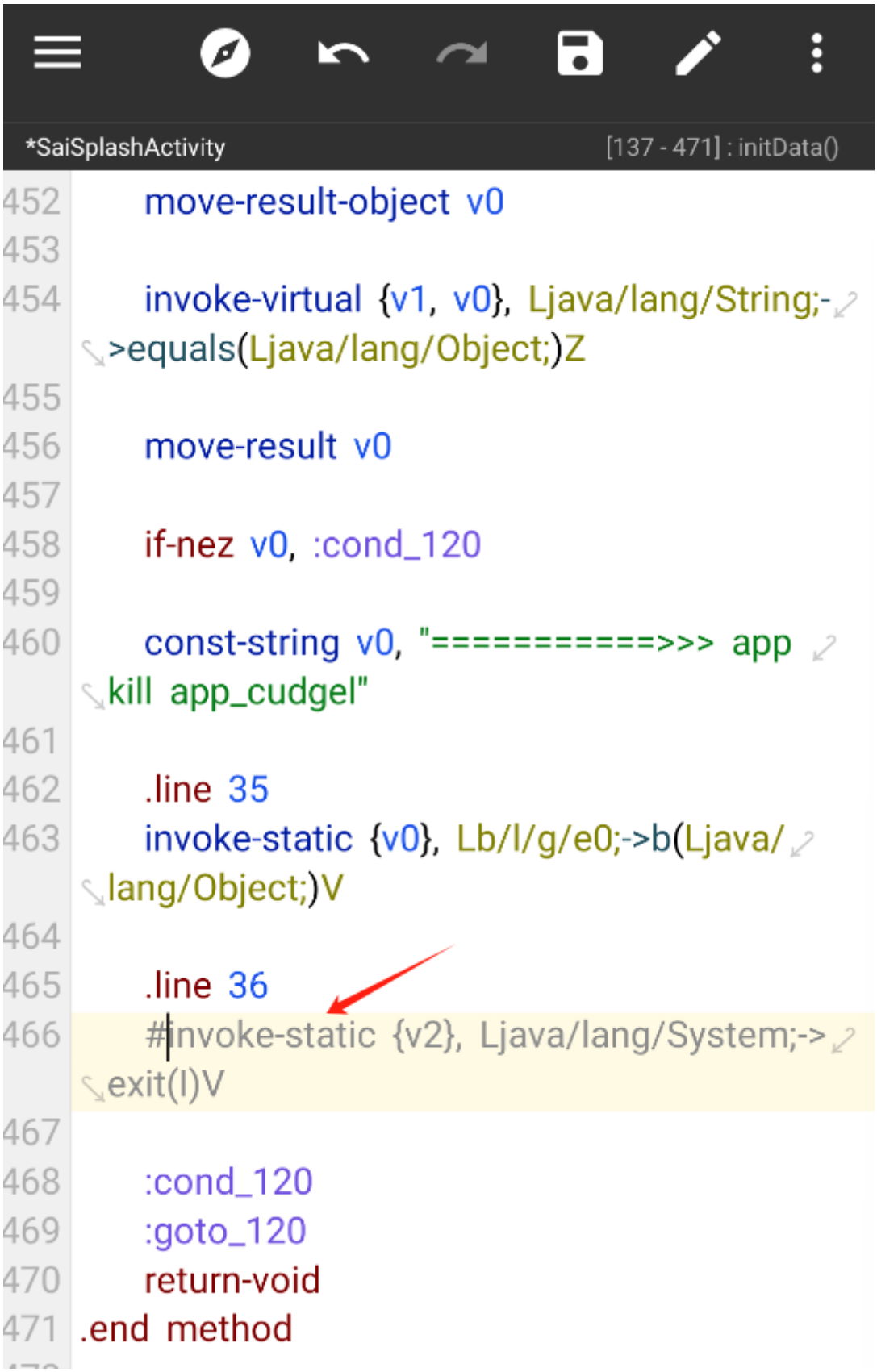
跟进)
com.pencil.yereen.saisplash.SaiSplashActivity.initData(SaiSplashActivity.java:36)

```
@Override // com.pencil.base.BaseAt
public void initData() {
    super.initData();
    h();
    u();
    ((b.l.b.c) this.f12203b).f3569d.postDelayed(new Runnable() { // from class: b.l.i.f
        @Override // java.lang.Runnable
        public final void run() {
            MainActivity.this.j();
        }
    }, 6000L);
    boolean z = false;
    if (BaseApp.getInstance().getSysInitBean() != null && !(z = d.a(false, BaseApp.getInstance().getSysInitBean().getUpgrade_info())) && v.c().f(SaiSPKey.INSTANCE.getLaunchCount())
        a0.a.f(this);
    }
    if (BaseApp.restartApp) {
        b.s.c.b.a().b(new SaiAppRestart());
    }
    ((MainViewModel) this.viewModel).t();
    try {
        v();
        c0 c0Var = c0.a;
        c0Var.s(this, z);
        c0Var.c();
    } catch (Exception unused) {
        e0.b("==========>>> ${e.message}");
    }
    String obj = b.c.a.b.d.f().toString();
    c0 c0Var2 = c0.a;
    if (!b.c.a.b.i.c(obj, c0Var2.f3986c).equals(c0Var2.j(R.string.app_cudgel))) {
        e0.b("===========>>> app kill app_cudgel");
        finish();
    }
    w();
}

@Override // com.pencil.base.BaseAt
public int initVariableId() {
```

我们可以注释导致这个kill进程的代码


MT管...
10:40
Dex编辑器++
临时工程
浏览
最近
搜索
搜索
查找内容
initData()
路径
/
搜索类型 方法名
搜索子目录
区分大小写
正则表达式
完全匹配
取消
确定

10:41
Dex编辑器++
临时工程
浏览
最近
搜索
常量
功能
发起新搜索
在当前结果中搜索
在当前结果中替换
清除搜索
搜索结果(3) - app kill
b.l.g
c0
const-string v0, "============>>> app kill app_cudgel"
com.pencil.yereen
MainActivity
const-string v0, "============>>> app kill app_cudgel"
com.pencil.yereen.saisplash
SaiSplashActivity
const-string v0, "============>>> app kill app_cudgel"



保存

然后查看其他的算法日志app

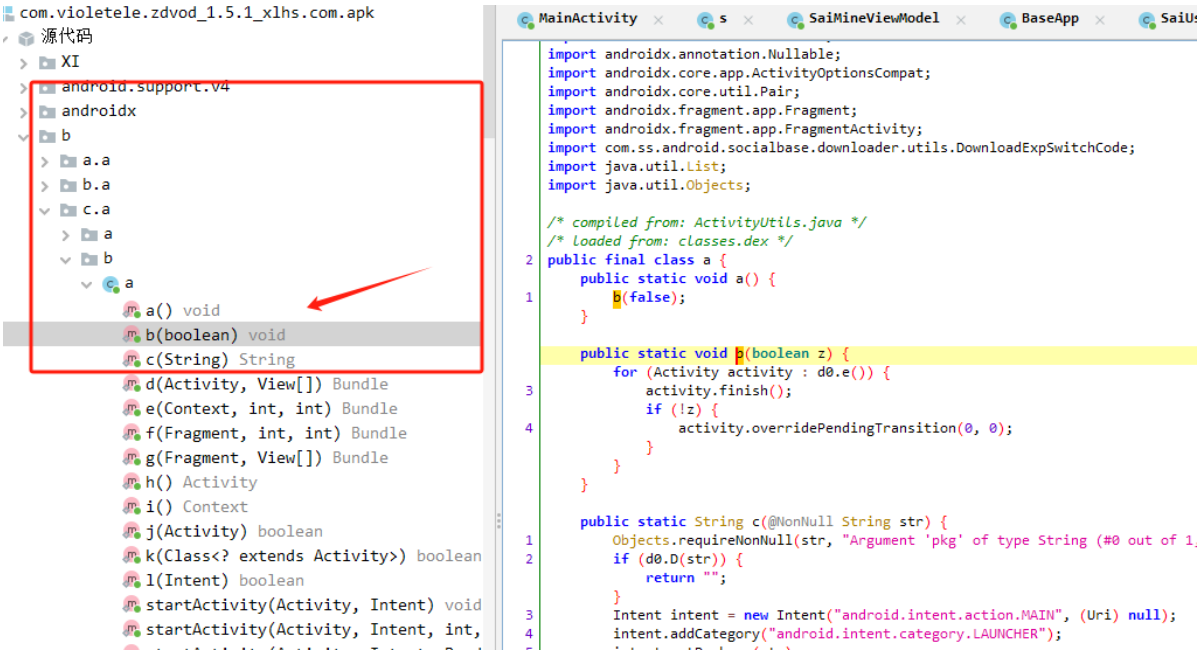
类名：com.pencil.yereen.saisplash.SaiSplashActivity

日志名称：结束程序-finish-已拦截

返回结果类型:void
返回结果值：void

调用堆栈：
```
    at de.robv.android.xposed.XposedBridge$
AdditionalHookInfo.callback(Unknown Source:
79)
    at b.c.a.b.a.b(ActivityUtils.java:3)
    at b.c.a.b.a.a(ActivityUtils.java:1)
    at b.l.g.c0.v(SaiFunUtils.java:4)
    at com.vmbind.base.BaseApplication$a
.onActivityDestroyed(BaseApplication.java:
2)
    at
com.trello.rxlifecycle2.components.support.RxAppC
ompatActivity.onDestroy(RxAppCompatActivity.java:
2)
    at
com.pencil.base.BaseAt.onDestroy(BaseAt.java:1)
    at com.pencil.yereen.MainAct
ivity.onDestroy(MainActivity.java:
1)
```

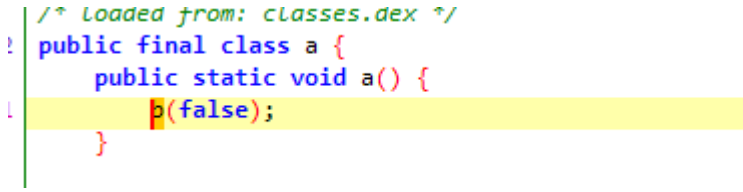
根据日志找到b.c.a.b.a.b



查看b方法

```
public static void b(boolean z) {
        for (Activity activity : d0.e()) {
            activity.finish();
            if (!z) {
                activity.overridePendingTransition(0, 0);
            }
        }
    }
```

跟进b方法，按键x 查找用例

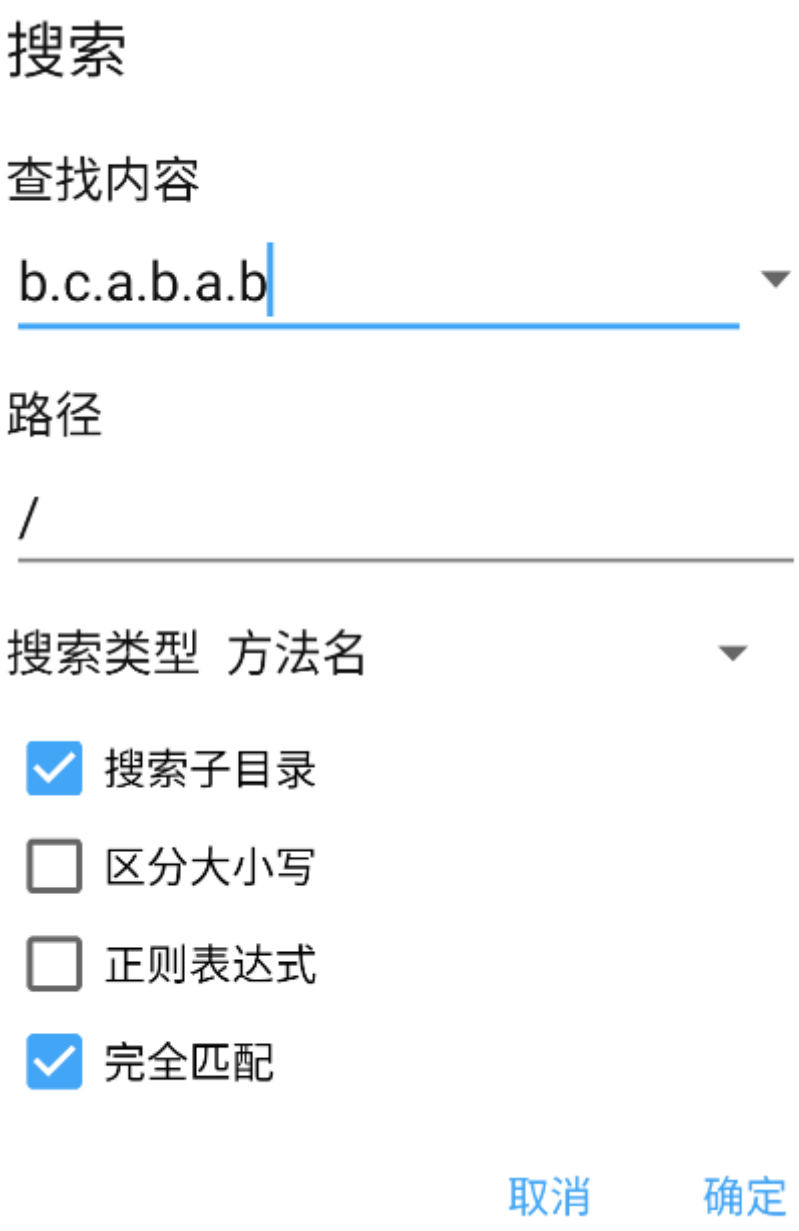


a方法将b设置为false

跟进a，查看到周围的代码如下

```java
public void t(HashMap<String, SaiAdResp.InfoBean> hashMap) {
    for (String str : hashMap.keySet()) {
        List<SaiAdResp.AdBean> sdk_list = hashMap.get(str).getSdk_list();
        for (int i2 = 0; i2 < sdk_list.size(); i2++) {
            if (String.valueOf(sdk_list.get(i2).getSdk_id()).equals(Constant.INSTANCE.ADAPI)) {
                b.c.a.b.v.d(SaiSPKey.adConfig).m(str + "-" + sdk_list.get(i2).getTag_id() + "-num", sdk_list.get(i2).getNum());
            } else {
                b.c.a.b.v.d(SaiSPKey.adConfig).m(str + "-" + sdk_list.get(i2).getSdk_id() + "-num", sdk_list.get(i2).getNum());
            }
        }
        b.c.a.b.v.d(SaiSPKey.adConfig).m(str + "-index", 0);
    }
}

public void u(View view, boolean z, String str) {
    if (z) {
        b.c.a.b.y.j(view).c(0).d(str).h();
    } else {
        b.c.a.b.y.j(view).c(0).d(str).f();
    }
}

public void v() {
    String obj = b.c.a.b.d.f().toString();
    c0 c0Var = a;
    if (!b.c.a.b.i.c(obj, c0Var.f3986c).equals(c0Var.j(R.string.app_cudgel))) {
        e0.b("===========>>> app kill app_cudgel");
        b.c.a.b.a.a();
    }
}
```

修改代码



此处直接断掉

return-void常见于void方法的结尾，用来结束方法的执行

```
10      .registers 1
11
12      const/4 v0, 0x0
13
14      .line 1
15      invoke-static {v0}, Lb/c/a/b/a;->b(Z)V
16
17      return-void
18  .end method
19
20  .method public static b(Z)V
21      .registers 4
22  return-void
23      .line 1
24      invoke-static {}, Lb/c/a/b/d0;->e()Ljava/
    util/List;
25
26      move-result-object v0
```

保存

继续查找其他的app kill

☐ 类名 ☐ 方法名 ☑ 字段名 ☑ 代码 ☐ 资源 ☐ 注释 ☑ 忽略大小写 ☐ 正则表达式 ☐ 只在当前页搜索

节点	
b.l.g.c0.v() void	e0.b("===========>>> app kill app_cudgel");
com.pencil.yereen.MainActivity.initData() void	e0.b("===========>>> app kill app_cudgel");
com.pencil.yereen.saisplash.SaiSplashActivity.initDa	e0.b("===========>>> app kill app_cudgel");

浏览
最近
搜索
常量
功能
发起新搜索
在当前结果中搜索
在当前结果中替换
清除搜索
搜索结果(3) - app kill
b.l.g
c0
const-string v0, "============>>> app kill app_cudgel"
com.pencil.yereen
MainActivity
const-string v0, "============>>> app kill app_cudgel"
com.pencil.yereen.saisplash
SaiSplashActivity
const-string v0, "============>>> app kill app_cudgel"

继续修改最后一处

```
MainActivity                                   [359 - 532] : initData()
    j(I)Ljava/lang/String;
510
511     move-result-object v1
512
513     invoke-virtual {v0, v1}, Ljava/lang/String;->equals(Ljava/lang/Object;)Z
514
515     move-result v0
516
517     if-nez v0, :cond_99
518
519     const-string v0, "============>>> app kill app_cudgel"
520
521     .line 17
522     invoke-static {v0}, Lb/l/g/e0;->b(Ljava/lang/Object;)V
523
524     .line 18
525     invoke-virtual {p0}, Landroid/app/Activity;->finish()V
526
527     .line 19
528     :cond_99
```

此处仍有一个finish导致程序崩溃，注释掉这个代码

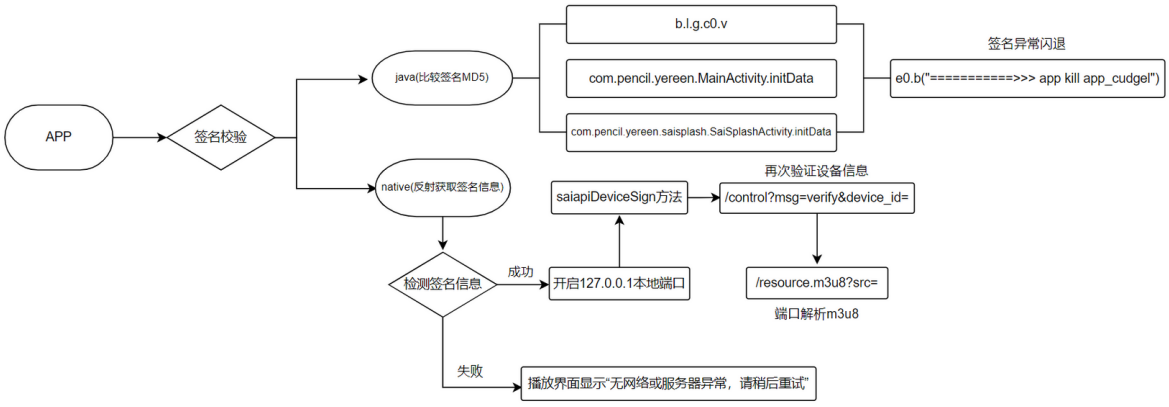
MainActivity [359 - 532] : initData()

```
j(I)Ljava/lang/String;

    move-result-object v1

    invoke-virtual {v0, v1}, Ljava/lang/String;->equals(Ljava/lang/Object;)Z

    move-result v0

    if-nez v0, :cond_99

    const-string v0, "============>>> app kill app_cudgel"

    .line 17
    invoke-static {v0}, Lb/l/g/e0;->b(Ljava/lang/Object;)V

    .line 18
    #invoke-virtual {p0}, Landroid/app/Activity;->finish()V

    .line 19
    :cond_99
    invoke-virtual {p0}, Lcom/pencil/yereen/
```

保存成功

保存 签名 安装，可以查看到app

VIP免广告特权：
邀请好友
推广记录
已累计邀请
已累计获得免广告
复制
填写邀请码
观看历史
我的收藏
我的下载
意见反馈
设置


APP
签名校验
java(比较签名MD5)
native(反射获取签名信息)
b.l.g.c0.v
com.pencil.yereen.MainActivity.initData
com.pencil.yereen.saisplash.SaiSplashActivity.initData
签名异常闪退
e0.b("==========>>> app kill app_cudgel")
检测签名信息
成功
开启127.0.0.1本地端口
saiapiDeviceSign方法
再次验证设备信息
/control?msg=verify&device_id=
/resource.m3u8?src=
端口解析m3u8
失败
播放界面显示"无网络或服务器异常，请稍后重试"

3、去除更新弹窗

```
public static boolean a(boolean z, SaiUpgradeInfo saiUpgradeInfo) {
        if (!a.b.b() && saiUpgradeInfo != null) {
            try {
                //获取版本号进行对比
                if (saiUpgradeInfo.getVersion_code() >= b.c.a.b.d.i()) {
                    boolean b2 = v.c().b(SaiSPKey.INSTANCE.getUpgrade_dialog(),
false);
                    if (z || !b2) {
                        if (saiUpgradeInfo.getIs_same() == 1) {
                            v c = v.c();
                            if (!c.b("appUpgrade_" +
saiUpgradeInfo.getVersion_code(), false)) {
                                new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                                v c2 = v.c();
                                c2.s("appUpgrade_" +
saiUpgradeInfo.getVersion_code(), true);
                                return true;
                            }
                        } else if (saiUpgradeInfo.getVersion_code() >
b.c.a.b.d.i()) {
                            new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                            return true;
                        } else if (z) {
                            ToastUtils.v("已经是最新版本");
                        }
                        e0.b("========>>> 包升级更新：${it.url}");
                    }
                } else if (z) {
                    ToastUtils.v("已经是最新版本");
                }
            } catch (Exception unused) {
            }
        }
        return false;
    }
```

第一步、打开算法助手的弹窗定位

第二步、算法助手启动app

触发后查看算法助手日志

类名：b.l.i.w.d

日志名称：Dialog弹窗

弹窗内容:考虑一下

立即更新

建议在wifi下更新哦~

返回结果类型:void
返回结果值：void

调用堆栈：
at de.robv.android.xposed.XC_MethodHook.callAfterHookedMethod(XC_MethodHook.java:68)
at EdHooker_.hook(Unknown Source:127)
at b.l.i.w.d.a(SaiUpgradeDg.java:9)
at com.pencil.yereen.MainActivity.initData(MainActivity.java:6)
at com.pencil.base.BaseAt.onCreate(BaseAt.java:9)
at com.pencil.yereen.MainActivity.onCreate(MainActivity.java:1)

根据日志提示来到代码

MainActivity | s | SaiMineViewModel | BaseApp | SaiUserInfo | SaiSplashActivity | a | c | a | c0 | d

```java
public class d extends Dialog implements View.OnClickListener {
    public Context a;

    /* renamed from: b  reason: collision with root package name */
    public SaiUpgradeInfo f4474b;

    /* renamed from: c  reason: collision with root package name */
    public Disposable f4475c;

    public d(@NonNull Context context, SaiUpgradeInfo saiUpgradeInfo) {
        super(context);
        this.a = context;
        this.f4474b = saiUpgradeInfo;
    }

    public static boolean a(boolean z, SaiUpgradeInfo saiUpgradeInfo) {
        if (!b.s.f.p.a.f4811b.b() && saiUpgradeInfo != null) {
            try {
                if (saiUpgradeInfo.getVersion_code() >= b.c.a.b.d.i()) {
                    boolean b2 = v.c().b(SaiSPKey.INSTANCE.getUpgrade_dialog(), false);
                    if (z || !b2) {
                        if (saiUpgradeInfo.getIs_same() == 1) {
                            v c2 = v.c();
                            if (!c2.b("appUpgrade_" + saiUpgradeInfo.getVersion_code(), false)) {
                                new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                                v c3 = v.c();
                                c3.s("appUpgrade_" + saiUpgradeInfo.getVersion_code(), true);
                                return true;
                            }
                        } else if (saiUpgradeInfo.getVersion_code() > b.c.a.b.d.i()) {
                            new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                            return true;
                        } else if (z) {
                            ToastUtils.v("已经是最新版本");
                        }
                        e0.b("========>>> 包升级更新: ${it.url}");
                    }
                } else if (z) {
                    ToastUtils.v("已经是最新版本");
                }
            } catch (Exception unused) {
            }
        }
        return false;
    }

    public void b() {
        b.s.c.c.b(this.f4475c);
    }
```

分析代码 跟进a方法

查找用例： b.l.i.w.d.a(boolean, SaiUpgradeInfo) boolean

节点	
b.l.i.r.q.onChanged(Object) void	b.l.i.w.d.a(true, BaseApp.getInstance().getSysInitBea
com.pencil.yereen.MainActivity.initData() void	if (BaseApp.getInstance().getSysInitBean() != null &&
com.pencil.yereen.saivideodetail.SaiDetailActivity.E	b.l.i.w.d.a(false, this.f12409j.getUpgrade_info());

第一种方法：注释掉app启动时检测的此代码，去除检测

MainActivity × s × SaiMineViewModel × BaseApp × SaiUserInfo × SaiSplashActivity × a × c × a × c0 × d × a

```
    @Override // com.pencil.base.BaseAt
    public int initContentView(Bundle bundle) {
        return R.layout.activity_main;
    }

    @Override // com.pencil.base.BaseAt
    public void initData() {
1       super.initData();
2       h();
3       u();
4       ((b.l.b.c) this.f12203b).f3569d.postDelayed(new Runnable() { // from class: b.l.i.f
            @Override // java.lang.Runnable
            public final void run() {
                MainActivity.this.j();
            }
        }, 6000L);
        boolean z = false;
5       if (BaseApp.getInstance().getSysInitBean() != null && !(z = d.a(false, BaseApp.getInstance().getSysInitBean().getUpgrade_info())) && v.c().f(SaiSPKey.INSTANCE.getLaunchCount(
8           a0.a.f(this);
        }
9       if (BaseApp.restartApp) {
10          b.s.c.b.a().b(new SaiAppRestart());
        }
11      ((MainViewModel) this.viewModel).t();
        try {
12          v();
13          c0 c0Var = c0.a;
            c0Var.s(this, z);
14          c0Var.c();
        } catch (Exception unused) {
15          e0.b("==========>>> ${e.message}");
        }
16      String obj = b.c.a.b.d.f().toString();
        c0 c0Var2 = c0.a;
        if (!b.c.a.b.i.c(obj, c0Var2.f3986c).equals(c0Var2.j(R.string.app_cudgel))) {
17          e0.b("===========>>> app kill app_cudgel");
18          finish();
        }
19      w();
    }

    @Override // com.pencil.base.BaseAt
    public int initVariableId() {
        return 10;
```

第二种方法，删除检测

```
    public static boolean a(boolean z, SaiUpgradeInfo saiUpgradeInfo) {
        if (!b.s.f.p.a.f4811b.b() && saiUpgradeInfo != null) {
            try {
                if (saiUpgradeInfo.getVersion_code() >= b.c.a.b.d.i()) {
                    boolean b2 = v.c().b(SaiSPKey.INSTANCE.getUpgrade_dialog(), false);
                    if (z || !b2) {
                        if (saiUpgradeInfo.getIs_same() == 1) {
                            v c2 = v.c();
                            if (!c2.b("appUpgrade_" + saiUpgradeInfo.getVersion_code(), false)) {
                                new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                                v c3 = v.c();
                                c3.s("appUpgrade_" + saiUpgradeInfo.getVersion_code(), true);
                                return true;
                            }
                        } else if (saiUpgradeInfo.getVersion_code() > b.c.a.b.d.i()) {
                            new d(b.c.a.b.a.h(), saiUpgradeInfo).show();
                            return true;
                        } else if (z) {
                            ToastUtils.v("已经是最新版本");
                        }
                        e0.b("========>>> 包升级更新: ${it.url}");
                    }
                } else if (z) {
                    ToastUtils.v("已经是最新版本");
                }
            } catch (Exception unused) {
            }
        }
        return false;
    }
```

d [39 - 220] : a()

```
>show()V

    return v3

    :cond_8e
    if-eqz p0, :cond_93

    .line 10
    invoke-static {v4}, Lcom/blankj/utilcode/util/ToastUtils;->v(Ljava/lang/CharSequence;)V

    :cond_93
    const-string p0, "========>>> 包升级更新：${it.url}"

    .line 11
    invoke-static {p0}, Lb/l/g/e0;->b(Ljava/lang/Object;)V

    goto :goto_9e

    :cond_99
    if-eqz p0, :cond_9e
```

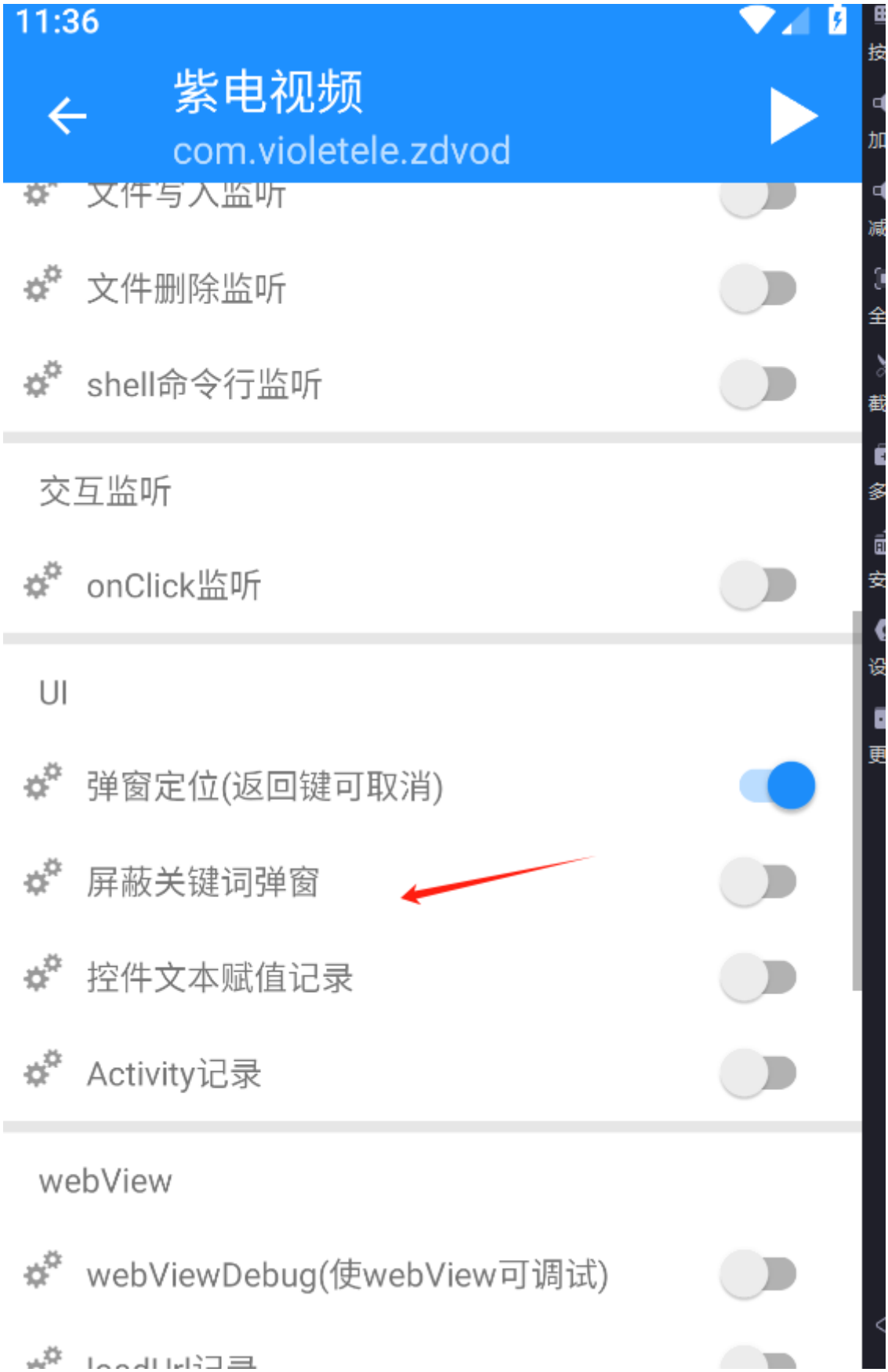
*d (1/5) [39 - 220] : a

```
.method public static a(ZLcom/pencil/saibeans/SaiUpgradeInfo;)Z
    .registers 7

    const-string v0, "appUpgrade_"

    .line 1
    sget-object v1, Lb/s/f/p/a;->b:Lb/s/f/p/a

    invoke-virtual {v1}, Lb/s/f/p/a;->b()Z

    move-result v1

    const/4 v2, 0x0

    #if-eqz v1, :cond_c

    return v2

    :cond_c
```

Tab / + - * = <

保存 签名 安装

或者使用 算法助手的拦截



4、布局优化

1.修改xml中的属性值

元素名称	属性名称	描述
LinearLayout	android:gravity	设置布局内部元素的位置，这里设置为中心和左对齐

元素名称	属性名称	描述
LinearLayout	android:orientation	设置 `LinearLayout` 的方向，这里设置为水平
LinearLayout	android:tag	为视图设置标签，这里设置为 "binding_13"
LinearLayout	android:background	设置背景图片，这里引用了名为 "sai_sp_stroke_divider" 的资源
LinearLayout	android:paddingLeft	设置左填充距离，这里引用了名为 "dp_15" 的资源
LinearLayout	android:paddingTop	设置顶部填充距离，这里设置为 10dp
LinearLayout	android:paddingBottom	设置底部填充距离，这里设置为 10dp
LinearLayout	android:visibility	设置视图可见性，这里设置为 "gone"
LinearLayout	android:layout_width	设置布局宽度，这里设置为 0dp
LinearLayout	android:layout_height	设置布局高度，这里设置为包裹内容
LinearLayout	android:layout_marginRight	设置右边距，这里引用了名为 "dp_8" 的资源
LinearLayout	android:layout_weight	设置布局权重，这里设置为 1.0
LinearLayout	android:paddingVertical	设置垂直方向的填充距离，这里设置为 10dp
ImageView	android:layout_width	设置图片视图宽度，这里引用了名为 "dp_32" 的资源
ImageView	android:layout_height	设置图片视图高度，这里引用了名为 "dp_32" 的资源
ImageView	android:src	设置图片资源，这里引用了名为 "icon_mine_invitecode" 的资源
TextView	android:textSize	设置文本字体大小，这里引用了名为 "sp_14" 的资源
TextView	android:textColor	设置文本颜色，这里引用了名为 "common_h0" 的资源
TextView	android:layout_width	设置文本视图宽度，这里设置为包裹内容
TextView	android:layout_height	设置文本视图高度，这里设置为包裹内容
TextView	android:layout_marginLeft	设置左边距，这里引用了名为 "dp_10" 的资源
TextView	android:text	设置文本内容，这里设置为 "填写邀请码"

```
android:visibility="gone"
android:visibility="gone" <!-- 初始隐藏，不影响其他控件布局 -->
```

2.上帝模式优化布局

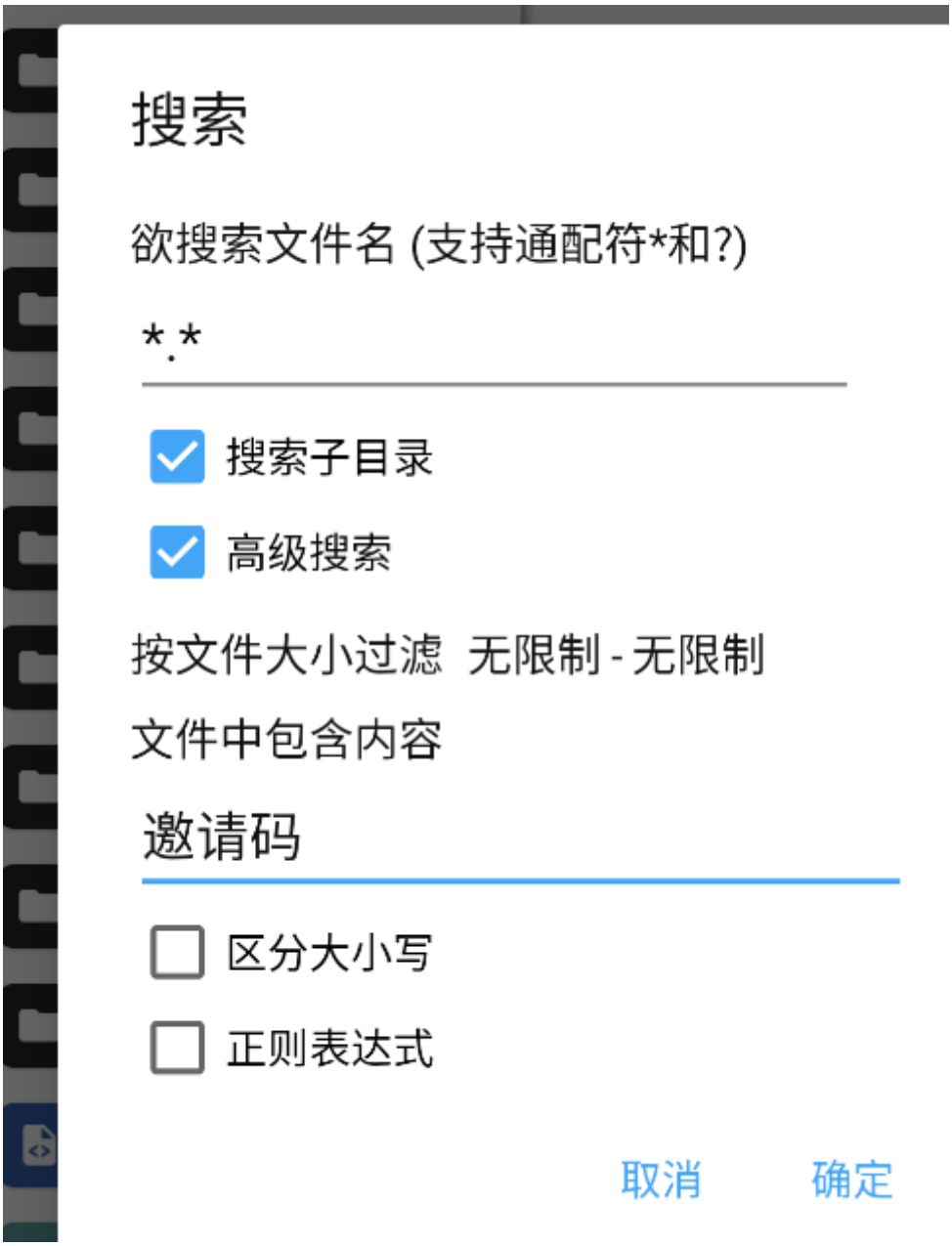
[白嫖大法]某影视邀请协议分析

（1）、手动改装

第一步 使用开发助手--布局查看

然后打开待测试的app，获取id 然后定位修改

同样可以通过mt搜索关键字修改


搜索
欲搜索文件名 (支持通配符*和?)
.
搜索子目录
高级搜索
按文件大小过滤 无限制 - 无限制
文件中包含内容
邀请码
区分大小写
正则表达式
取消
确定

GF1.xml
81-01-01 01:01 11.94K
GF2.xml
81-01-01 01:01 564B
gG.9.png
81-01-01 01:01 214B
Gg.png
81-01-01 01:01 58.78K
acct
25-03-23 09:07
bin
25-02-24 19:56
cache
25-03-23 09:07
config
25-03-23 09:07
打开方式...
反编译
字符常量池
gl.png
lost+found

```
"@7F060055"
                    android:layout_width="-2"
                    android:layout_height="-2"
                    
android:layout_marginTop="@7F0700CA"
                    android:text="VIP免广告特
权："
                    
android:layout_marginStart="@7F0700B1" />
                <TextView
                    android:textSize=
"@7F0702CE"
                    android:textStyle=
"0x00000001"
                    android:textColor=
"@7F060177"
                    android:gravity=
"0x00000001"
                    android:tag="binding_5"
                    android:background=
"@7F080112"
                    android:paddingTop=
"@7F0700B4"
                    android:layout_width=
```

修改

android:visibility="gone"

```
</RelativeLayout>
<RelativeLayout
    android:orientation="vertical"
    android:background="@drawable/icon_mine_vip"
    android:layout_width="match_parent"
    android:visibility="gone"
    android:layout_height="@dimen/dp_190"
    android:layout_marginTop="@dimen/dp_15"
    android:layout_marginStart="@dimen/dp_15"
    android:layout_marginEnd="@dimen/dp_15">
    <TextView
        android:textSize="@dimen/sp_17"
        android:textStyle="bold"
        android:textColor="@color/common_h0"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="@dimen/dp_22"
        android:text="VIP免广告特权："
        android:layout_marginStart="@dimen/dp_15" />
    <TextView
        android:textSize="@dimen/sp_13"
        android:textStyle="bold"
```

保存 安装

同理修改邀请码（我这没会员）

其他的地方可以发给ai调整或者自己手动分析调整

（2）、使用上帝模式

需要ls激活

上帝模式

通用

上帝模式
伙计们可别玩脱了

启用通知栏快捷设置
Android7.0+支持磁贴

支持

加入组织
这里有一群老伙计们

支持我的老伙计
太感谢你了我的伙计

应用规则

可以直接修改控件

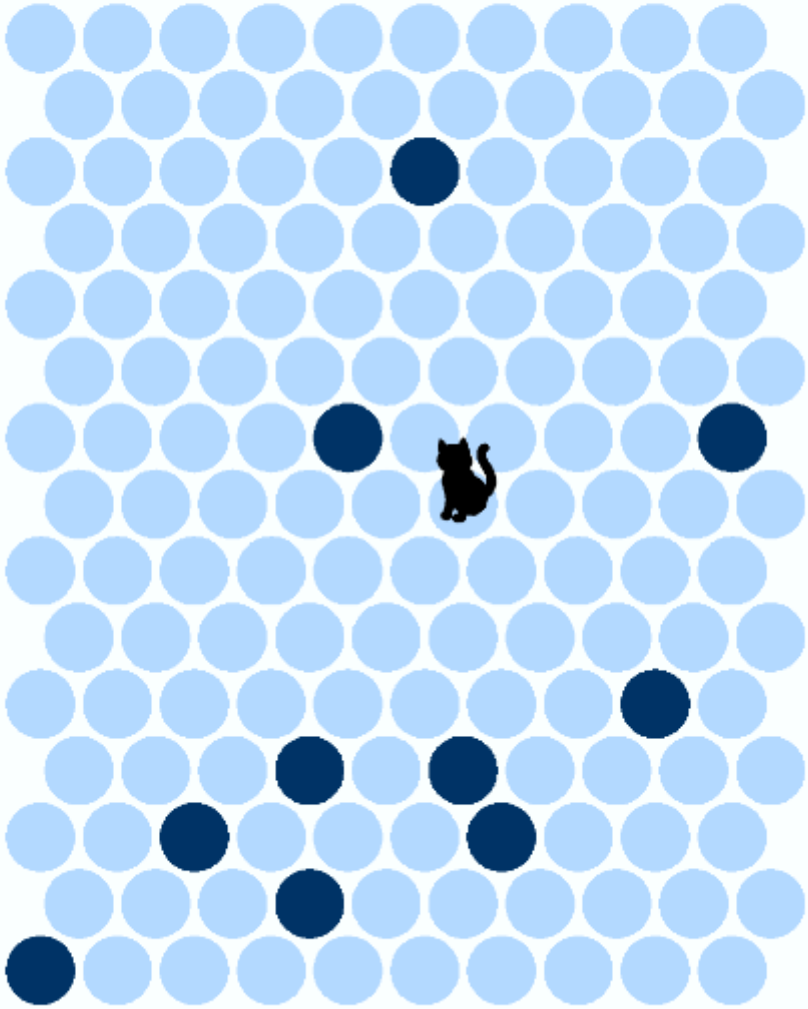
番外实战篇2-【2024春节】解题领红包活动，启动!

1、初级题1

【2024春节】解题领红包之Android初级题

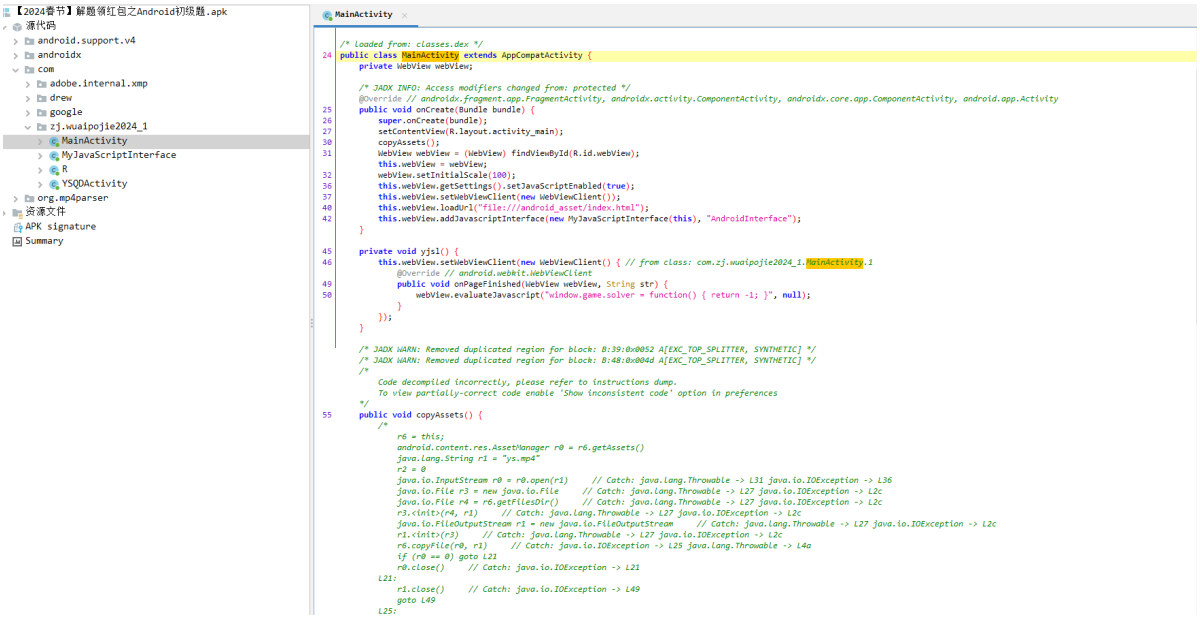
你的Flag被小猫叼走了，快抓住他！

围住小猫



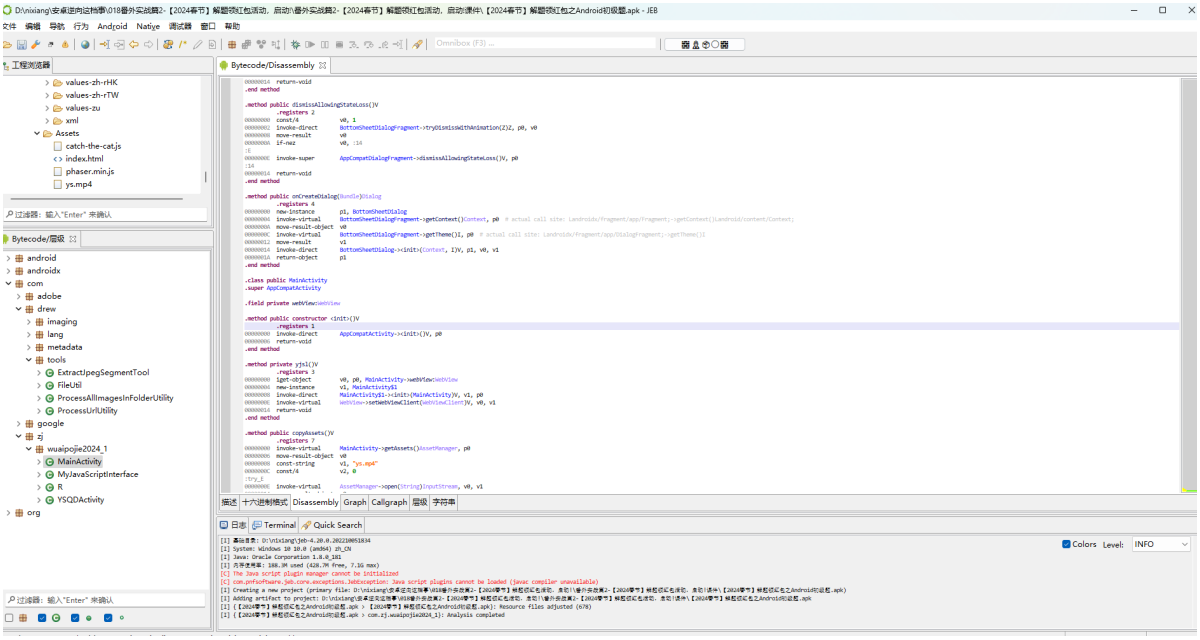
重置

吾爱破解 正己



（1）、方法1

使用jadx或者jeb查看代码



jeb需要右键反编译（解析）下

这段代码定义了一个名为 `MainActivity` 的 Android 活动，其主要功能是将 `assets` 目录下的 `ys.mp4` 文件复制到应用的文件目录，然后使用 `WebView` 加载 `assets` 目录中的 `index.html` 文件，同时注入 JavaScript 接口，并且在页面加载完成后注入一段 JavaScript 代码。

```
        }

        VideoView videoView0 = (VideoView)this.findViewById(id.videoView);
        videoView0.setVideoURI(Uri.parse(s));
        videoView0.setMediaController(new MediaController(this));
        videoView0.setOnPreparedListener(new MediaPlayer.OnPreparedListener() {
            @Override  // android.media.MediaPlayer$OnPreparedListener
            public void onPrepared(MediaPlayer mediaPlayer0) {
                mediaPlayer0.setVideoScalingMode(1);
            }
        });
        videoView0.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
            @Override  // android.media.MediaPlayer$OnCompletionListener
            public void onCompletion(MediaPlayer mediaPlayer0) {
                String s = YSQDActivity.extractDataFromFile(YSQDActivity.this.filePath);
                YSQDActivity.this.tv.setText(
            }
        });
        videoView0.start();
    }
}
```

```
com.zj.wuaipojie2024_1.YSQDActivity.extractDataFromFile(java.lang.String) : j
Descriptor: Lcom/zj/wuaipojie2024_1/YSQDActivity;->extractDataFromFile(Ljava/

[AST Objects Stack]
Class: class:Lcom/zj/wuaipojie2024_1/YSQDActivity;
  Method: method:com.zj.wuaipojie2024_1.YSQDActivity.playVideo
    Block: { /* 8 statements */ }
      Call: setOnCompletionListener
        New: new com.zj.wuaipojie2024_1.YSQDActivity$2
          Class: class:Lcom/zj/wuaipojie2024_1/YSQDActivity$2;
            Method: method:com.zj.wuaipojie2024_1.YSQDActivity$2.onCompletion
```

分析代码

跟进到myJavaScriptInterface0

```java
package com.zj.wuaipojie2024_1;

import android.content.Context;
import android.content.Intent;
import android.webkit.JavascriptInterface;

public class MyJavaScriptInterface {
    private Context mContext;

    public MyJavaScriptInterface(Context context0) {
        this.mContext = context0;
    }

    @JavascriptInterface
    public void onSolverReturnValue(int v) {
        if(v == -1) {
            Intent intent0 = new Intent(this.mContext, YSQDActivity.class);
            this.mContext.startActivity(intent0);
        }
    }
}
```

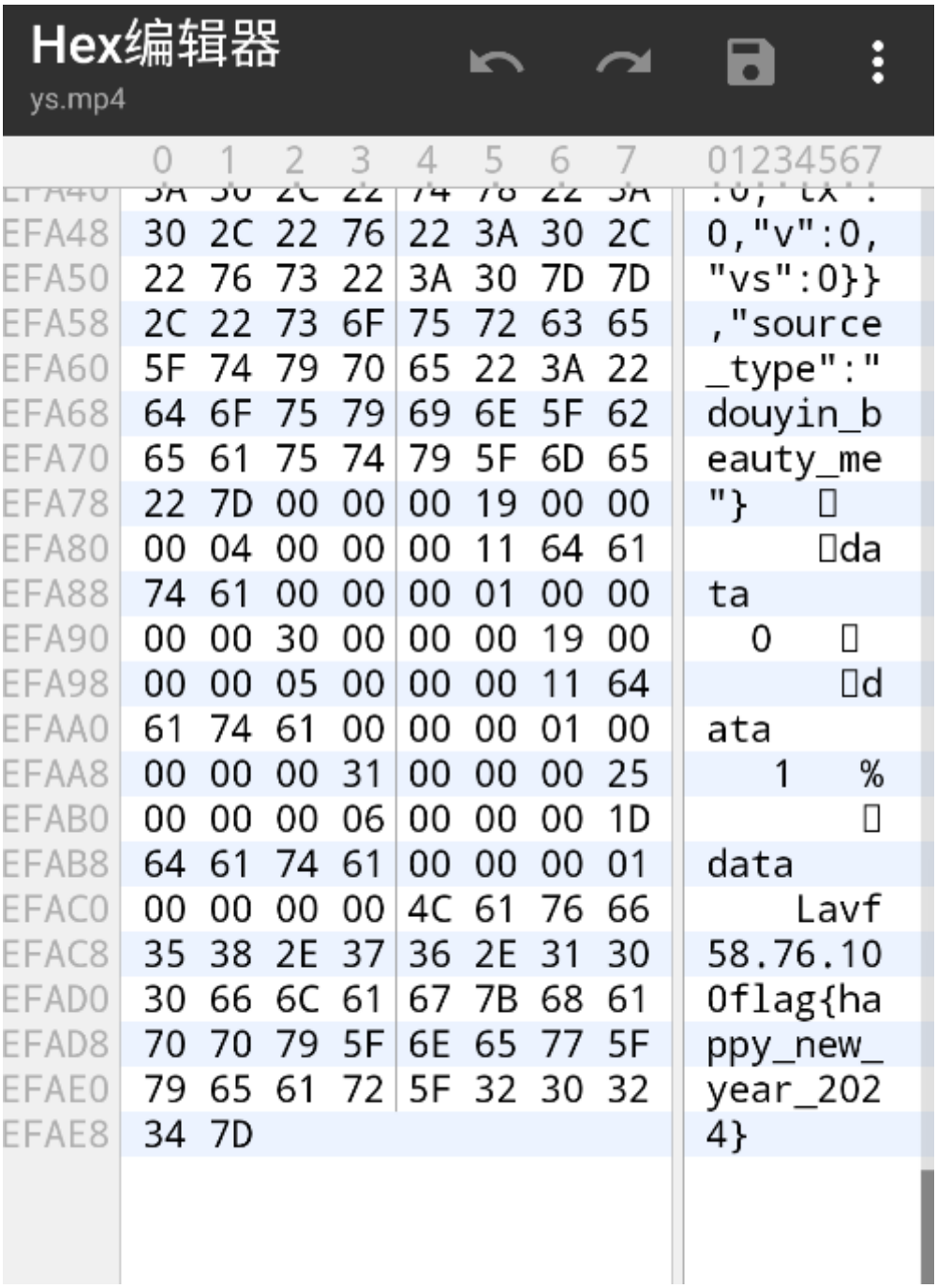
继续跟进YSQDActivity.

```java
public static String extractDataFromFile(String str) {
    // 定义局部变量str2
    String str2;
    // 定义局部变量indexOf
    int indexOf;
    try {
        // 创建一个用于读取文件的RandomAccessFile实例，以只读模式打开
        RandomAccessFile randomAccessFile = new RandomAccessFile(str, "r");
        // 获取文件的长度
        long length = randomAccessFile.length();
        // 从文件末尾开始向前搜索，搜索范围限制为文件末尾的30个字节内
        for (long max = Math.max(length - 30, 0L); max < length; max++) {
            // 定位文件指针到max位置
            randomAccessFile.seek(max);
            // 定义一个长度为30的字节数组用于读取数据
            byte[] bArr = new byte[30];
            // 从当前文件指针位置读取数据到bArr中
            randomAccessFile.read(bArr);
            // 将读取的字节数组转换成字符串，并检查是否包含"flag{"字符串
            if (new String(bArr, StandardCharsets.UTF_8).indexOf("flag{") != -1)
{
                // 如果找到"flag{"，则从这个位置开始，截取到"}"为止的字符串，并拼接上"}"
                String str3 = str2.substring(indexOf).split("\\}")[0] + "}";
                // 关闭文件并返回找到的字符串
                randomAccessFile.close();
                return str3;
            }
        }
        // 如果整个文件中都没有找到符合条件的字符串，则关闭文件并返回null
        randomAccessFile.close();
        return null;
    } catch (Exception e) {
        // 处理异常，打印异常堆栈信息
```

```
        e.printStackTrace();
        return null;
    }
}
```

根据代码我们可以看到在mp4视频文件的末尾

使用16进制打开视频文件



（2）、方法2

hook主动调用

手机启动frida

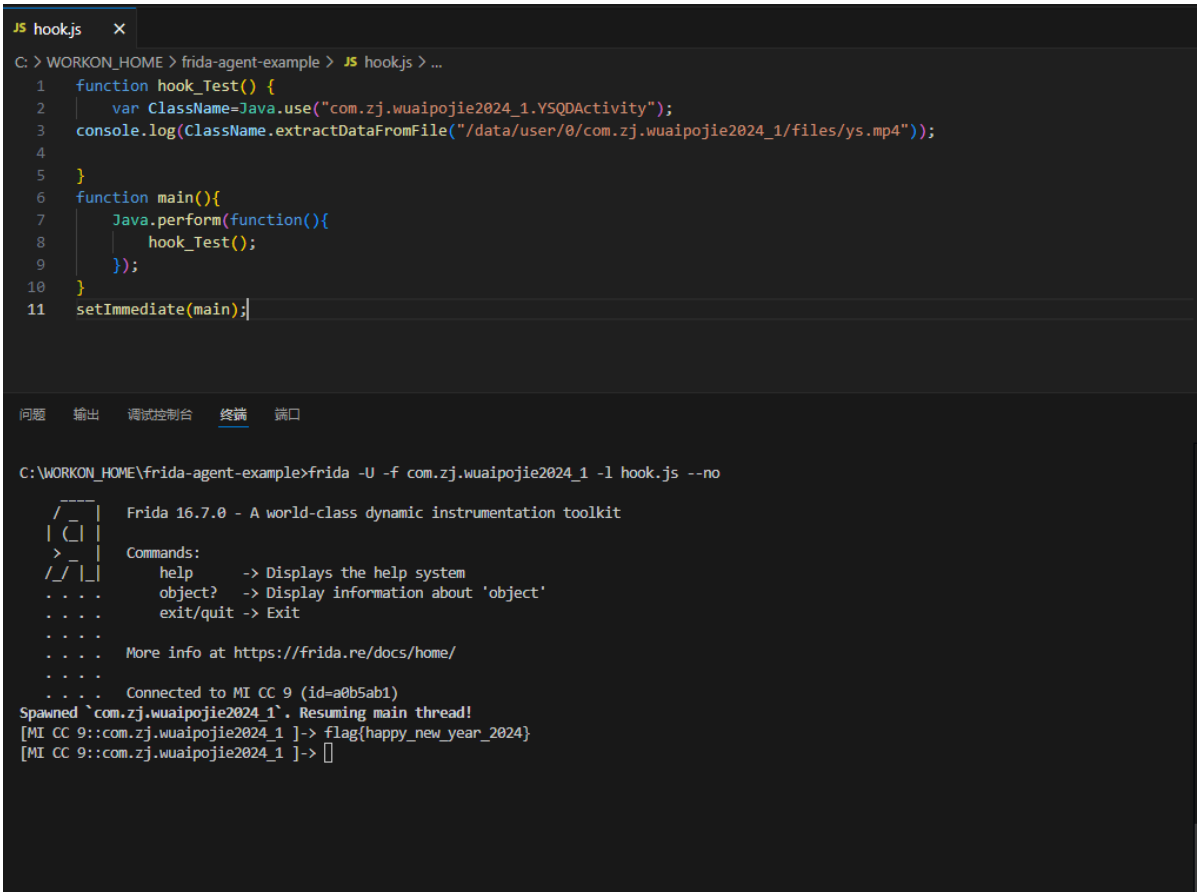
```
Microsoft Windows [版本 10.0.22621.4317]
(c) Microsoft Corporation。保留所有权利。

C:\Users\Administrator>adb shell
adb.exe: more than one device/emulator

C:\Users\Administrator>adb shell
gracelte:/ $ su
gracelte:/ # cd /data/local/tmp
gracelte:/data/local/tmp # chmod 777 frida
chmod: frida: No such file or directory
1|gracelte:/data/local/tmp # ls
gracelte:/data/local/tmp # ./frida-server-16.2.1-android-arm64
```

vs修改脚本

```
function hook_Test() {
    var ClassName=Java.use("com.zj.wuaipojie2024_1.YSQDActivity");
console.log(ClassName.extractDataFromFile("/data/user/0/com.zj.wuaipojie2024_1/files/ys.mp4"));

}
function main(){
    Java.perform(function(){
        hook_Test();
    });
}
setImmediate(main);
```

执行

```
frida -U -f com.zj.wuaipojie2024_1 -l hook.js --no
```



（3）、方法3

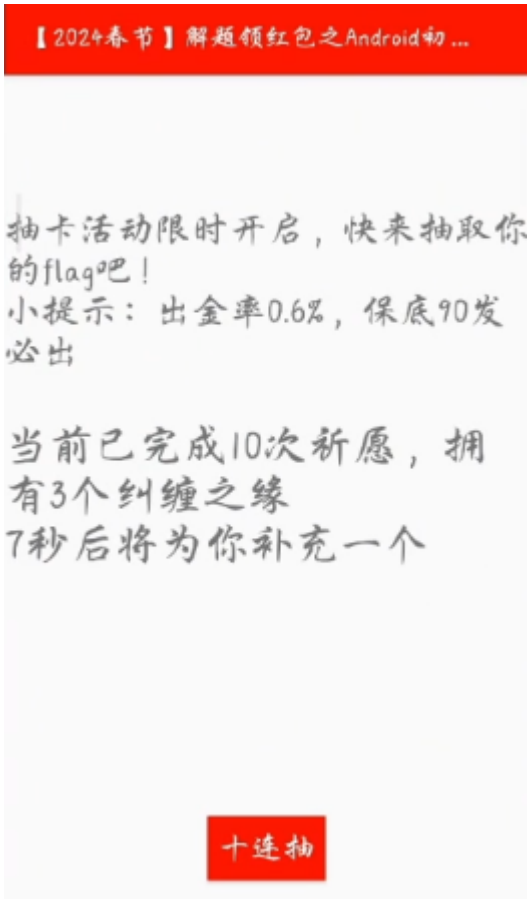
```
pip install objection

objection -g com.zj.wuaipojie2024_1 explore

android intent launch_activity com.zj.wuaipojie2024_1.YSQDActivity
```

```
C:\WORKON_HOME\frida-agent-example>objection -g com.zj.wuaipojie2024_1 explore
Using USB device `MI CC 9`
Agent injected and responds ok!

     _   _         _   _
 ___| |_|_|___ ___| |_|_|___ ___
| . | . | | -_|  _|  _| | . |   |
|___|___| |___|___|_| |_|___|_|_|
      |___|(object)inject(ion) v1.11.0

     Runtime Mobile Exploration
        by: @leonjza from @sensepost

[tab] for command suggestions
com.zj.wuaipojie2024_1 on (Xiaomi: 9) [usb] # android intent launch_activity com.zj.wuaipojie2024_1.YSQDActivity
```

flag{happy_new_year_2024}

2、初级题2

测试app需要重新签名安装，不然可能会报错



查看代码

```java
FlagActivity
package com.kbtx.redpack_simple;

import a.b.c.h;
import android.content.pm.PackageManager;
import android.content.pm.Signature;
import android.os.Bundle;
import android.widget.TextView;
import b.a.a.a.a;
import java.nio.ByteBuffer;

/* loaded from: classes.dex */
public class FlagActivity extends h {
    public static byte[] o = {86, -18, 98, 103, 75, -73, 51, -104, 104, 94, 73, 81, 125, 118, 112, 100, -29, 63, -33, -110, 108, 115, 51, 59, 55, 52, 77};

    @Override // a.b.c.h, a.i.a.d, androidx.activity.ComponentActivity, a.f.b.g, android.app.Activity
    public void onCreate(Bundle bundle) {
        byte[] bArr;
        Signature[] signatureArr;
        super.onCreate(bundle);
        setContentView(R.layout.activity_flag);
        byte[] bArr2 = o;
        try {
            signatureArr = getPackageManager().getPackageInfo(getPackageName(), 64).signatures;
        } catch (PackageManager.NameNotFoundException unused) {
            bArr = new byte[0];
        }
        if (signatureArr != null && signatureArr.length >= 1) {
            byte[] byteArray = signatureArr[0].toByteArray();
            ByteBuffer allocate = ByteBuffer.allocate(bArr2.length);
            for (int i = 0; i < bArr2.length; i++) {
                allocate.put((byte) (bArr2[i] ^ byteArray[i % byteArray.length]));
            }
            bArr = allocate.array();
            StringBuilder d = a.d("for honest players only: \n");
            d.append(new String(bArr));
            ((TextView) findViewById(R.id.tvFlagHint)).setText(d.toString());
        }
        bArr = new byte[0];
        StringBuilder d2 = a.d("for honest players only: \n");
        d2.append(new String(bArr));
        ((TextView) findViewById(R.id.tvFlagHint)).setText(d2.toString());
    }
}
```

直接用jeb打开安装包反编译查看并分析源码，发现逻辑主要集中在FlagActivity中，拉到最后仔细分析，应该是在h->findViewById(I)View, p0, v0的时候会有对应的结果，按住Ctrl键跳进findViewById方法在invoke-virtual处下个断点启动

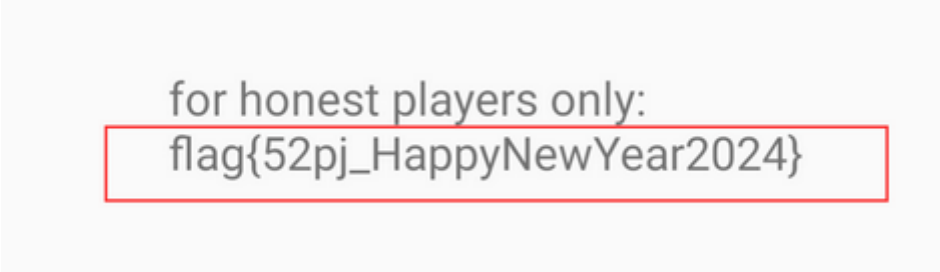
原文链接：https://blog.csdn.net/m0_46642036/article/details/136295349

（1）、方法1

直接执行

```
objection -g com.kbtx.redpack_simple explore

android intent launch_activity com.kbtx.redpack_simple.FlagActivity

或者
adb shell am start -D -n com.kbtx.redpack_simple/.FlagActivity
```

```
C:\WORKON_HOME\frida-agent-example>objection -g com.kbtx.redpack_simple explore
Using USB device `MI CC 9`
Agent injected and responds ok!

     _   _         _   _
 ___| |_|_|___ ___| |_|_|___ ___
| . | . | | -_|  _|  _| | . |   |
|___|___| |___|___|_| |_|___|_|_|
      |___|(object)inject(ion) v1.11.0

     Runtime Mobile Exploration
        by: @leonjza from @sensepost

[tab] for command suggestions
com.kbtx.redpack_simple on (Xiaomi: 9) [usb] # android intent launch_activity com.kbtx.redpack_simple.FlagActivity
```



启动后发现flag直接出现了，可能只要成功启动就会解出，***flag{52pj_HappyNewYear2024}***

（2）、方法2

分析代码

```
    }

    @Override // a.b.c.h, a.i.a.d, androidx.activity.ComponentActivity, a.f.b.g, android.app.Activity
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.activity_wish);
        this.q.schedule(new a(), 1000L, 1000L);
        findViewById(R.id.btn_action).setOnClickListener(new View.OnClickListener() { // from class: b.b.a.a
            @Override // android.view.View.OnClickListener
            public final void onClick(View view) {
                WishActivity wishActivity = WishActivity.this;
                int i = wishActivity.o[0];
                for (int i2 = 0; i2 < 10; i2++) {
                    int[] iArr = wishActivity.o;
                    iArr[0] = iArr[0] - 1;
                    iArr[1] = iArr[1] + 1;
                    double random = Math.random();
                    int[] iArr2 = wishActivity.o;
                    if (random < (iArr2[1] <= 80 ? 0.006d : (iArr2[1] - 80) * 0.1d)) {
                        Toast.makeText(wishActivity, "恭喜你十连出金了，奖品为 flag 提示！", 1).show();
                        wishActivity.startActivity(new Intent(wishActivity, FlagActivity.class));
                        return;
                    }
                }
                Toast.makeText(wishActivity, "哎呀呀，（又）没抽中，一会再试试吧", 0).show();
            }
        });
    }
```

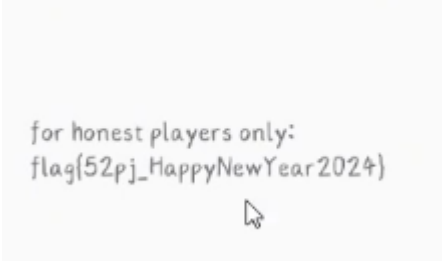
需要修改许愿次数，然后获取flag

```
function hookTest1() {
    try {
        // 加载 java.util.Arrays 类
        var Arrays = Java.use("java.util.Arrays");
        console.log("尝试查找 com.kbtx.redpack_simple.WishActivity 类的实例");

        // 添加延迟，等待应用加载必要的类
        setTimeout(function () {
            // 查找 com.kbtx.redpack_simple.WishActivity 类的实例
            Java.choose("com.kbtx.redpack_simple.WishActivity", {
                onMatch: function (obj) {
                    console.log("找到 com.kbtx.redpack_simple.WishActivity 实例: "
+ obj);
                    try {
                        // 检查 obj.o 是否存在
                        if (obj.o) {
                            // 获取 obj.o 的值并转换为字符串
                            var oAsString = Arrays.toString(obj.o.value);
                            console.log("o 字段的值: " + oAsString);
                            // 修改 obj.o 的值
                            obj.o.value = Java.array('I', [90, 90, 122]);
                            console.log("o 字段的值已修改为: [90, 90, 122]");
                        } else {
                            console.log("obj.o 字段不存在");
```

```
                        }
                    } catch (e) {
                        console.error("访问 o 字段时出错: " + e);
                    }
                },
                onComplete: function () {
                    console.log("查找 com.kbtx.redpack_simple.WishActivity 实例完成");
                }
            });
        }, 5000); // 延迟 5 秒

    } catch (e) {
        console.error("执行 hookTest1 函数时出错: " + e);
    }
}

function main() {
    Java.perform(function () {
        try {
            // 调用 hookTest1 函数
            hookTest1();
        } catch (e) {
            console.error("在 Java.perform 中执行 hookTest1 时出错: " + e);
        }
    });
}

// 立即执行 main 函数
setImmediate(main);
```

```
frida -U -f com.kbtx.redpack_simple -l hook.js --no
```

```
C:\WORKON_HOME\frida-agent-example>frida -U -f com.kbtx.redpack_simple -l hook.js --no
     ____
    / _  |   Frida 16.7.0 - A world-class dynamic instrumentation toolkit
   | (_| |
    > _  |   Commands:
   /_/ |_|       help      -> Displays the help system
   . . . .       object?   -> Display information about 'object'
   . . . .       exit/quit -> Exit
   . . . .
   . . . .   More info at https://frida.re/docs/home/
   . . . .
   . . . .   Connected to MI CC 9 (id=a0b5ab1)
Spawned `com.kbtx.redpack_simple`. Resuming main thread!
[MI CC 9::com.kbtx.redpack_simple ]-> 尝试查找 com.kbtx.redpack_simple.WishActivity 类的实例
找到 com.kbtx.redpack_simple.WishActivity 实例: com.kbtx.redpack_simple.WishActivity@4050674
o 字段的值: [1, 10, 2]
o 字段的值已修改为: [90, 90, 122]
查找 com.kbtx.redpack_simple.WishActivity 实例完成
```



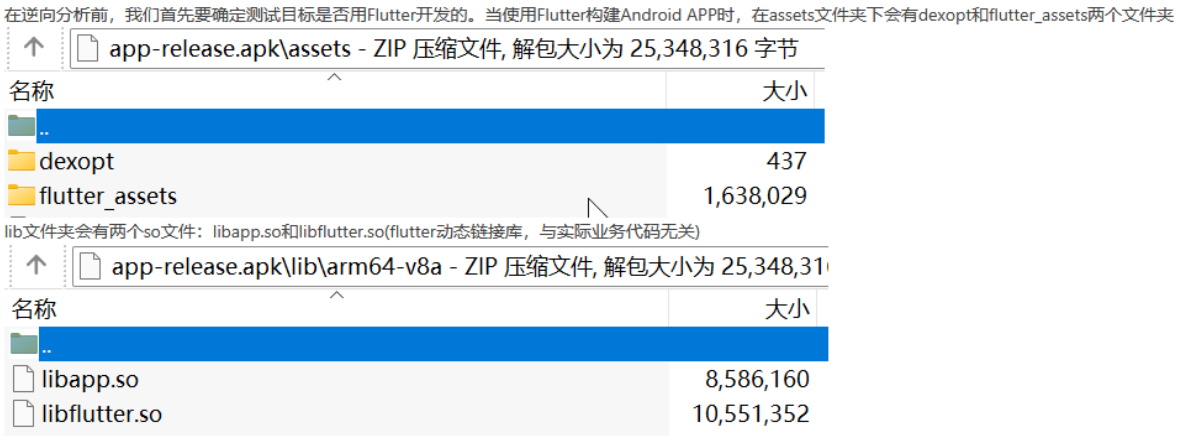
番外实战篇3-拨云见日之浅谈Flutter逆向

1、Flutter简介

Flutter是Google构建在开源的Dart VM之上，使用Dart语言开发的移动应用开发框架，可以帮助开发者使用一套Dart代码就能快速在移动iOS 、Android上构建高质量的原生用户界面，同时还支持开发Web和桌面应用。
`Flutter`引擎是一个用于**高质量跨平台应用的可移植运行时，由`C/C++`编写。**它实现了`Flutter`的核心库，包括动画和图形、文件和网络I/O、辅助功能支持、插件架构，以及用于开发、编译和运行`Flutter`应用程序的`Dart`**运行时**和工具链。引擎将底层`C++`代码包装成`Dart`代码，通过`dart:ui`暴露给`Flutter`框架层。
[flutter开源地址]
[flutter官网]

[原创]Flutter概述和逆向技术发展时间线，带你快速了解

2、Flutter特征



3、Flutter抓包对抗

```
原理：
```

1. **Dart语言标准库的网络请求不走Wi-Fi代{过}{滤}理**：Flutter使用的是Dart语言，其标准库中的网络请求不会通过代{过}{滤}理发送，这与许多其他应用不同。常规的抓包工具通常依赖于代{过}{滤}理来捕获网络流量，因此无法捕获Flutter应用的网络请求。
2. **Dart SDK中的证书信任**：Dart SDK在Android平台上强制只信任系统目录下的证书。这意味着Flutter应用不会信任用户安装的证书，除非这些证书位于Android系统的`/system/etc/security/cacerts`目录中。这是通过Dart源码中的`runtime/bin/security_context_linux.cc`文件实现的。

通过分析Flutter应用程序抛出的错误，可以定位到触发错误的源代码位置，错误指向了`handshake.cc:352`，这是一个处理SSL握手的源代码位置。

```
 E/flutter (10371):  [ERROR:flutter/runtime/dart_isolate.cc(805)] Unhandled
exception:
 E/flutter (10371):  HandshakeException: Handshake error in client (OS Error:
 E/flutter (10371):  NO_START_LINE(pem_lib.c:631)
 E/flutter (10371):  PEM routines(by_file.c:146)
 E/flutter (10371):  NO_START_LINE(pem_lib.c:631)
 E/flutter (10371):  PEM routines(by_file.c:146)
 E/flutter (10371):  CERTIFICATE_VERIFY_FAILED: self signed certificate in
certificate chain(handshake.cc:352))
```

了绕过SSL验证，需要找到一个合适的hook点，即源代码中可以被拦截和修改以改变程序行为的位置。`ssl_verify_peer_cert`函数是一个可能的hook点，但经过测试，仅仅修改这个函数的返回值并不能成功绕过SSL验证。

进一步分析源代码后，发现`session_verify_cert_chain`函数可以作为另一个hook点。这个函数在验证证书链时，如果证书验证失败，会返回一个错误。

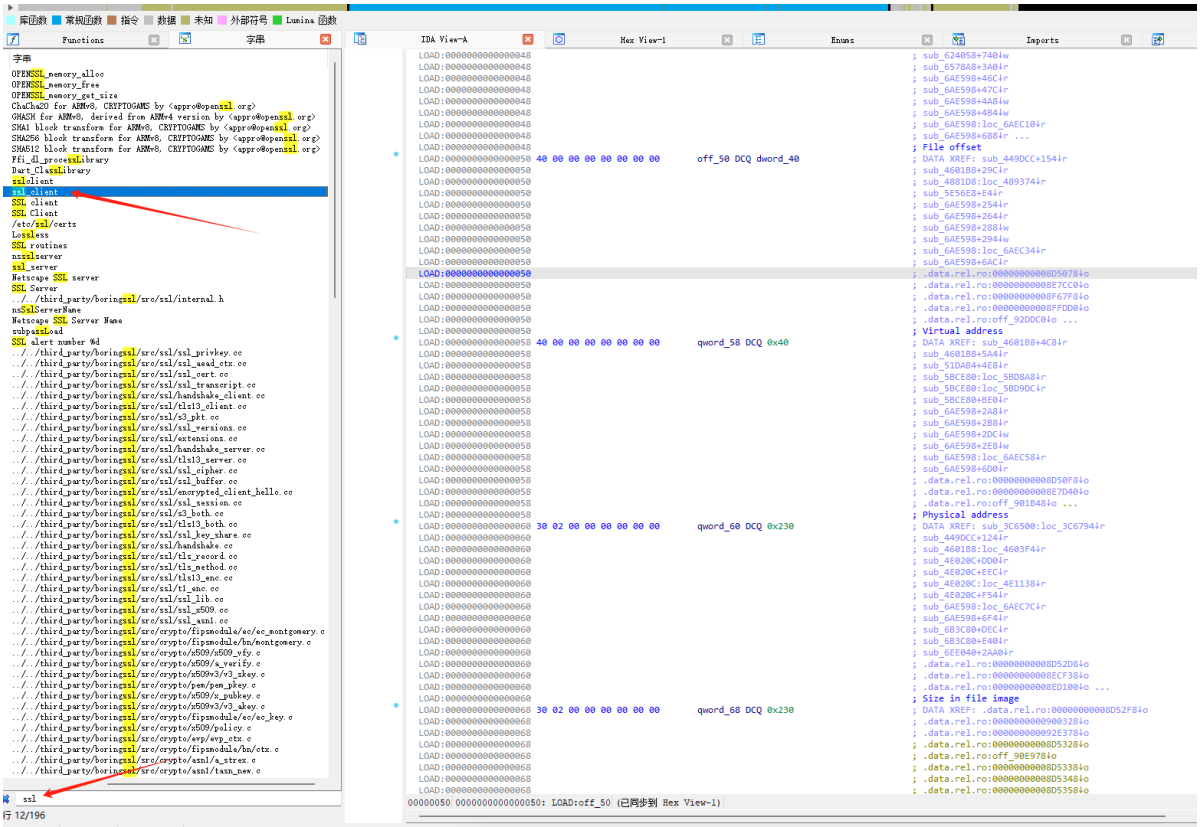
```
ret = ssl->ctx->x509_method->session_verify_cert_chain(
            hs->new_session.get(), hs, &alert)
            ? ssl_verify_ok
            : ssl_verify_invalid;
```

ession_verify_cert_chain函数定义在ssl_x509.cc，在该方法里可以看到有`ssl_client和ssl_server`两个字符串可以辅助定位方法

（1）、hook_ssl_client

ida查看lib\arm64-v8a\libflutter.so

进入设置--视图---打开子视图--子串



```
.text:00000000005DC72C                          ; __int64 __fastcall sub_5DC72C(__int64, __int64 *, _BYTE *)
.text:00000000005DC72C                          sub_5DC72C                              ; DATA XREF: .data.rel.ro:00000000009035
.text:00000000005DC72C
.text:00000000005DC72C                          var_70= -0x70
.text:00000000005DC72C                          var_68= -0x68
.text:00000000005DC72C                          var_60= -0x60
.text:00000000005DC72C                          var_50= -0x50
.text:00000000005DC72C                          var_40= -0x40
.text:00000000005DC72C                          var_30= -0x30
.text:00000000005DC72C                          var_20= -0x20
.text:00000000005DC72C                          var_10= -0x10
.text:00000000005DC72C
.text:00000000005DC72C                          ; FUNCTION CHUNK AT .text:000000000089702C SIZE 00000008 BYTES
.text:00000000005DC72C
.text:00000000005DC72C                          ; __unwind {
.text:00000000005DC72C FF C3 01 D1              SUB             SP, SP, #0x70
.text:00000000005DC730 FD 7B 01 A9              STP             X29, X30, [SP,#0x70+var_60]
.text:00000000005DC734 FC 6F 02 A9              STP             X28, X27, [SP,#0x70+var_50]
.text:00000000005DC738 FA 67 03 A9              STP             X26, X25, [SP,#0x70+var_40]
.text:00000000005DC73C F8 5F 04 A9              STP             X24, X23, [SP,#0x70+var_30]
.text:00000000005DC740 F6 57 05 A9              STP             X22, X21, [SP,#0x70+var_20]
.text:00000000005DC744 F4 4F 06 A9              STP             X20, X19, [SP,#0x70+var_10]
.text:00000000005DC748 08 0A 80 52              MOV             W8, #0x50 ; 'P'
.text:00000000005DC74C 48 00 00 39              STRB            W8, [X2]
.text:00000000005DC750 1A 54 40 F9              LDR             X26, [X0,#0xA8]
.text:00000000005DC754 DA 02 00 B4              CBZ             X26, loc_5DC7AC
.text:00000000005DC754
.text:00000000005DC758 48 03 40 F9              LDR             X8, [X26]
.text:00000000005DC75C 88 02 00 B4              CBZ             X8, loc_5DC7AC
.text:00000000005DC75C
.text:00000000005DC760 39 20 40 A9              LDP             X25, X8, [X1]
.text:00000000005DC764 F3 03 02 AA              MOV             X19, X2
.text:00000000005DC768 FF 01 0B 94              BL              sub_89CF64
.text:00000000005DC768
.text:00000000005DC76C F7 03 1F AA              MOV             X23, XZR
```

使用脚本hook

```
function hook_dlopen() {
    var android_dlopen_ext = Module.findExportByName(null, "android_dlopen_ext");
    Interceptor.attach(android_dlopen_ext, {
        onEnter: function (args) {
            var so_name = args[0].readCString();
            if (so_name.indexOf("libflutter.so") >= 0) this.call_hook = true;
        }, onLeave: function (retval) {
            if (this.call_hook) hookFlutter();
        }
    });
}

function hook_ssl_verify_result(address) {
    Interceptor.attach(address, {
            onEnter: function(args) {
                console.log("Disabling SSL validation")
            },
            onLeave: function(retval) {
                                console.log("Retval: " + retval);
                retval.replace(0x1);
            }
        });
    }
function hookFlutter() {
    var m = Process.findModuleByName("libflutter.so");
    //利用函数前10字节定位
    var pattern = "FF C3 01 D1 FD 7B 01 A9 FC 6F 02 A9FA 67 03 A9 F8 5F 04 A9 F6
57 05 A9 F4 4F 06 A9 08 0A 80 52 48 00 00 39";
    var res = Memory.scan(m.base, m.size, pattern, {
        onMatch: function(address, size){
            console.log('[+] ssl_verify_result found at: ' + address.toString());
        // Add 0x01 because it's a THUMB function
        // Otherwise, we would get 'Error: unable to intercept function at
0x9906f8ac; please file a bug'
```

```
            hook_ssl_verify_result(address);
        },
        onError: function(reason){
            console.log('[!] There was an error scanning memory');
        },
        onComplete: function() {
            console.log("All done")
        }
    });
}



function main(){
    Java.perform(function(){
        hook_dlopen();

    });
}
setImmediate(main);
```

执行

```
frida -U -f pro.mandu.enjoyreading -l hook.js --no
```

```
> WORKON_HOME > frida-agent-example > JS hook.js > ...
function hookFlutter() {
    var res = Memory.scan(m.base, m.size, pattern, {
        }
    });
}



function main(){
    Java.perform(function(){
        hook_dlopen();

    });
}
setImmediate(main);
```

```
题    输出    调试控制台    终端    端口

-android-arm64.so
S C:\WORKON_HOME\frida-agent-example> frida -U -f pro.mandu.enjoyreading -l hook.js --no
     ____
    / _  |   Frida 16.7.0 - A world-class dynamic instrumentation toolkit
   | (_| |
    > _  |   Commands:
   /_/ |_|       help      -> Displays the help system
   . . . .       object?   -> Display information about 'object'
   . . . .       exit/quit -> Exit
   . . . .
   . . . .   More info at https://frida.re/docs/home/
   . . . .
   . . . .   Connected to MI CC 9 (id=a0b5ab1)
pawned `pro.mandu.enjoyreading`. Resuming main thread!
MI CC 9::pro.mandu.enjoyreading ]-> [+] ssl_verify_result found at: 0x79521b072c
ll done
```

此时使用小黄鸟抓包 可以捕获成功

（2）、reflutter之patch

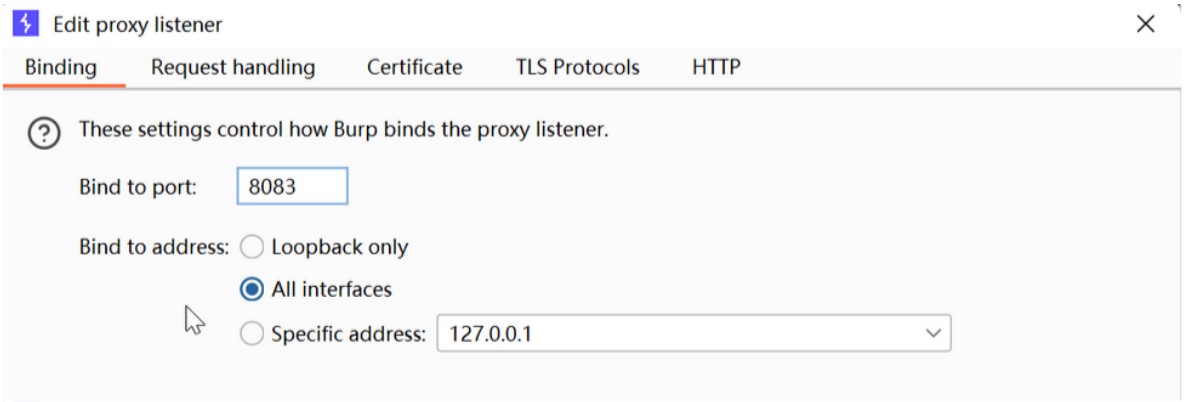
[reFlutter开源地址]

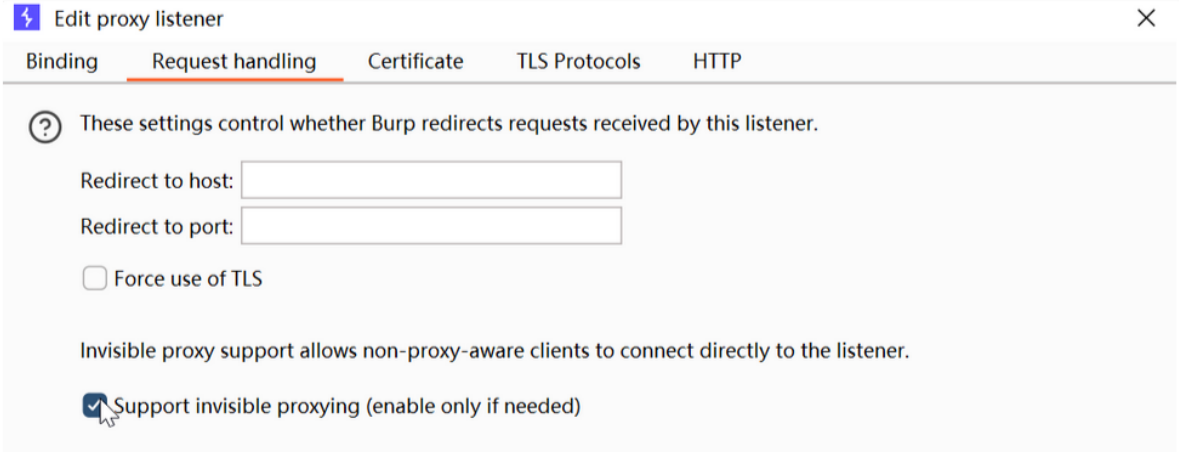
1.pip3 install reflutter pip安装对应的库
2.输入命令：reflutter flutter.apk
选择1流量监控和拦截，输入PC端的IP地址后(cmd窗口输入ipconfig)，将获取到release.RE.apk，但此apk尚未签名，需要我们手动签名(输入命令的过程需要全局代{过}{滤}理)

```
PS C:\Users\zhengji\Documents> reflutter md.apk

 Choose an option:

 1. Traffic monitoring and interception
 2. Display absolute code offset for functions

 [1/2]? 1

Example: (192.168.1.154) etc.
Please enter your BurpSuite IP: 192.168.[redacted]

 Wait...


SnapshotHash: 7dbbeeb8ef7b91338640dca3927636de
The resulting apk file: ./release.RE.apk
Please sign,align the apk file

Configure Burp Suite proxy server to listen on *:8083
Proxy Tab -> Options -> Proxy Listeners -> Edit -> Binding Tab

Then enable invisible proxying in Request Handling Tab
Support Invisible Proxying -> true
```

3.使用MT管理器或者uber-apk-signer.jar签名，输入命令：*java -jar uber-apk-signer-1.2.1.jar --apk release.RE.apk*。然后将重签名的apk安装到真机或者模拟器上。

4.设置BurpSuite的代{过}{滤}理，端口为8083，绑定所有地址，并且勾选All interfaces，使非代{过}{滤}理意识的客户端直接连接到侦听器。





5.设置Drony的wifi代{过}{滤}理主机名端口和BurpSuite一致，然后触发app就能抓到包了

Burp Project Intruder Repeater View Help Burp Suite Professional v20

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Compare

Intercept HTTP history WebSockets history Proxy settings

Filter settings: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status code	Leng
52	https://api.yjllq.com	GET	/api/soso/issearchsoso?t=17225152...	✓		200	533
53	https://www.google-analytic...	POST	/mp/collect?measurement_id=G-V...	✓		204	503
54	https://web.sdk.qcloud.com	GET	/player/tcplayer/release/v4.8.0/tcp...			200	5241
55	https://api.yjllq.com	GET	/api/lnitv3/settlev8?t=1722515244...	✓		200	2155
56	https://yy.youyi800.com	GET	/wap/?_timestamp=172251524857...	✓		200	2435
57	https://yy.youyi800.com	GET	/api/wx/js_sdk?url=https%3A%2F...	✓		200	1053
58	https://yy.youyi800.com	GET	/api/home/index			200	4018
59	https://yy.youyi800.com	GET	/api/home/service?page=1&per_p...	✓		200	3286
60	http://connect.rom.miui.com	GET	/generate_204			204	114
64	https://img.youyi800.com	GET	/yysq/images/20240717/17211772...	✓		200	8648
84	https://pg2.mandu.pro	POST	/batch			200	284

Request

Pretty Raw Hex

```
POST /batch HTTP/2
Host: pg2.mandu.pro
Content-Type: application/json
User-Agent: posthog-android/2.0.3
Content-Encoding: gzip
Connection: Keep-Alive
Accept-Encoding: gzip, deflate, br
Content-Length: 652
```

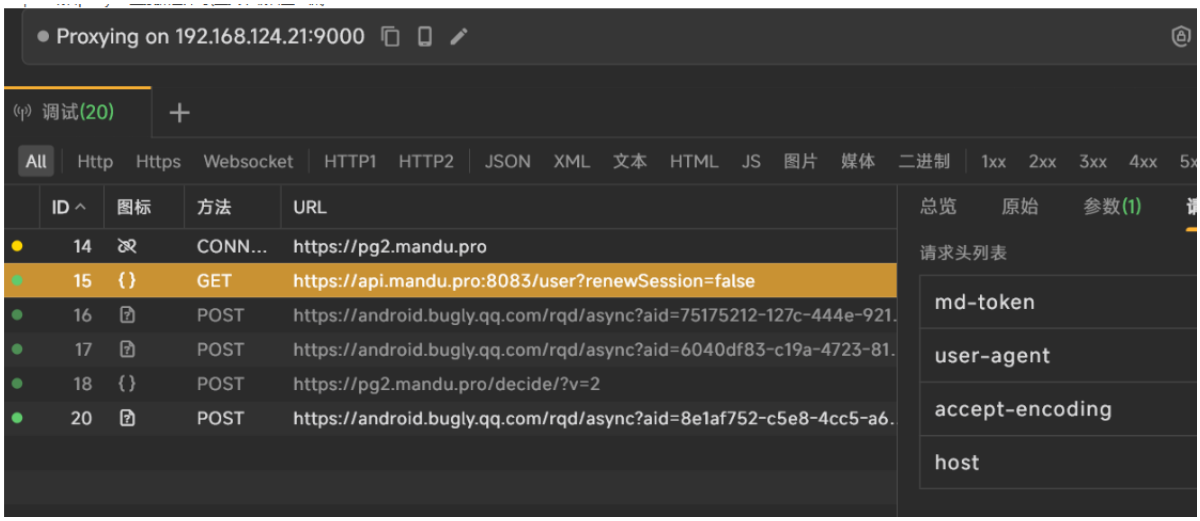
Response

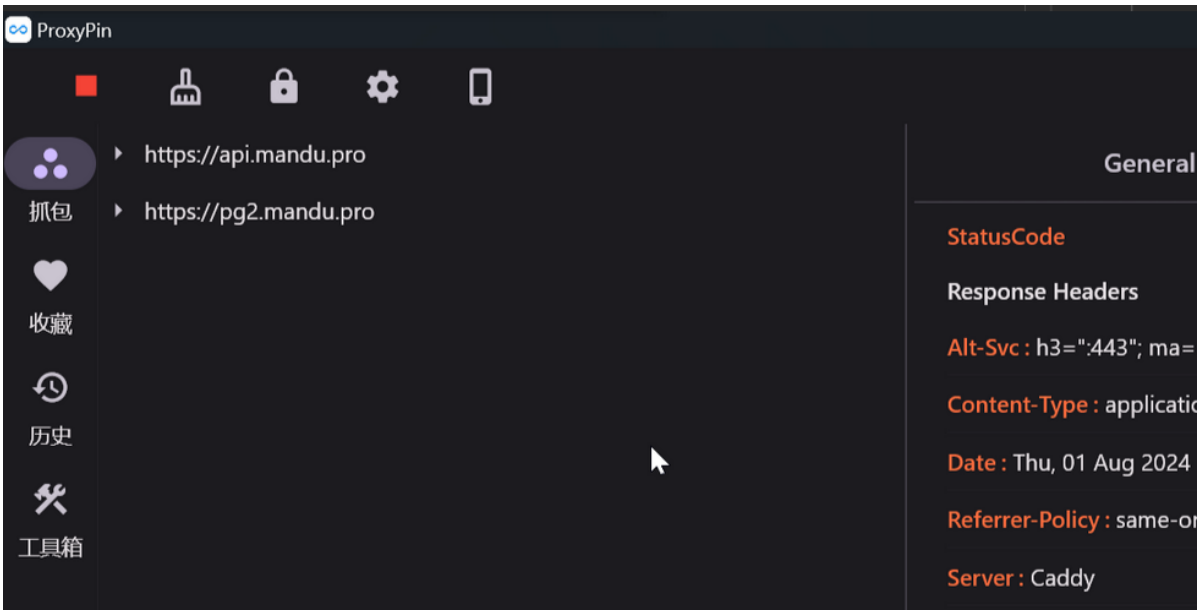
Pretty Raw

```
HTTP/2 200
Alt-Svc: h3
Content-Typ
Date: Thu,
Referrer-Po
Server: Cac
Server: gur
Vary: Cooki
X-Content-T
X-Frame-Opt
Content-Ler

{
  "status":
}
```

（3）、Reqable&proxyPin(推荐)

Reqable或者proxyPin直接抓包即可





4、Flutter反编译

（1）、快照

使用`readelf -s`命令读取保存快照信息的`libapp.so`将会输出下面的内容

```
Symbol table '.dynsym' contains 6 entries:
Num:    Value          Size Type    Bind   Vis      Ndx Name
 0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
 1: 000000000014c000 29728 OBJECT  GLOBAL DEFAULT    7 _kDartVmSnapshotInstructi
 2: 0000000000153440 0x22bd30 OBJECT  GLOBAL DEFAULT    7
_kDartIsolateSnapshotInst
 3: 0000000000000200 15248 OBJECT  GLOBAL DEFAULT    2 _kDartVmSnapshotData
 4: 0000000000003dc0 0x147af0 OBJECT  GLOBAL DEFAULT    2
_kDartIsolateSnapshotData
 5: 00000000000001c8    32 OBJECT  GLOBAL DEFAULT    1 _kDartSnapshotBuildId
```

“快照”指的是 Flutter 应用在编译过程中生成的特定数据结构，用于加速应用的启动和运行。具体来说，快照包括四种类型：

- `_kDartVmSnapshotData`： 代表 isolate 之间共享的 Dart 堆 (heap) 的初始状态。有助于更快地启动 Dart isolate，但不包含任何 isolate 专属的信息。
- `_kDartVmSnapshotInstructions`：包含 VM 中所有 Dart isolate 之间共享的通用例程的 AOT 指令。这种快照的体积通常非常小，并且大多会包含程序桩 (stub)。
- `_kDartIsolateSnapshotData`：代表 Dart 堆的初始状态，并包含 isolate 专属的信息。
- `_kDartIsolateSnapshotInstructions`：包含由 Dart isolate 执行的 AOT 代码。

其中`_kDartIsolateSnapshotInstructions`是最为重要的，因为包含了所有要执行的AOT代码，即业务相关的代码。

（2）、逆向方法

1.(静态)解析libapp.so，即写一个解析器，将libapp.so中的快照数据按照其既定格式进行解析，获取业务代码的类的各种信息，包括类的名称、其中方法的偏移等数据，从而辅助逆向工作。
关于Flutter快照的具体刨析只需要看下面引用的两篇文章
[Reverse engineering Flutter apps (Part 1) (tst.sh)]
[Reverse engineering Flutter apps (Part 2) (tst.sh)]

2.(动态)编译修改过的`libflutter.so`并且重新打包到APK中，在启动APP的过程中，由修改过的引擎动态链接库将快照数据获取并且保存。

PS:不同版本的Dart引擎其快照格式不同，所以静态的方法就需要频繁跟着版本更新迭代，成本极高，而动态也需要重新编译对应版本的链接库。同时如果APP作者抹除版本信息和hash信息，则无从下手，且重打包APK极易被检测到。

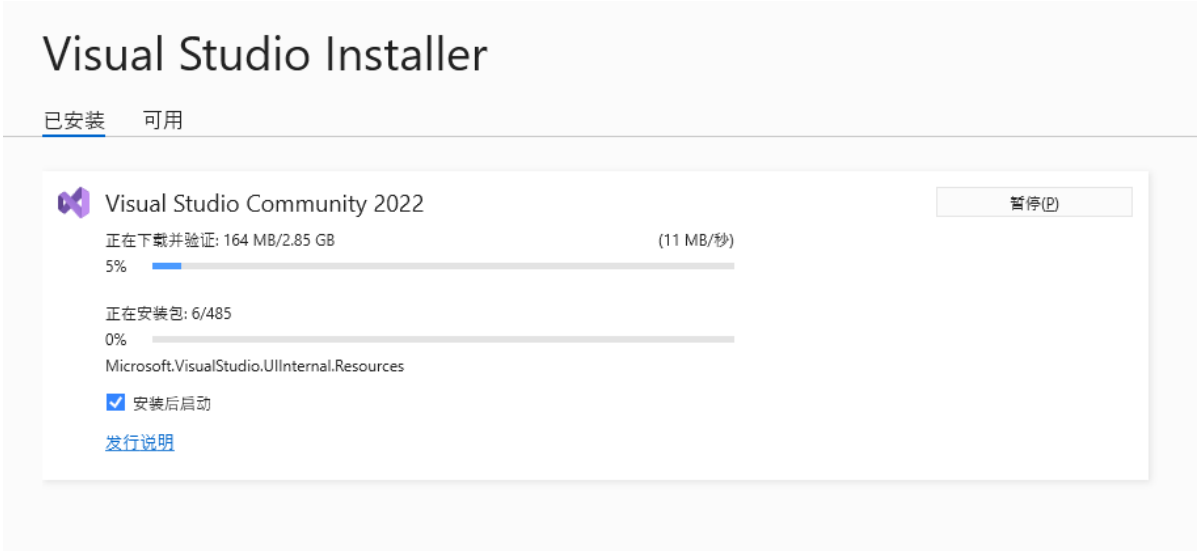
静态方法推荐工具:[blutter]
动态方法推荐工具:[reFlutter]

（3）、blutter实战

环境:python3.10
1.首先安装git
[下载地址]

2.下载visual studio
[下载地址]

3.下载安装，在工作负荷里勾选"使用C++的桌面开发"



4.clone项目(全程运行在代{过}{滤}理环境否则会导致无法下载)，或者下载解压到指定文件夹

```
git clone https://github.com/worawit/blutter --depth=1
```

5.进入到blutter文件夹，cmd窗口运行初始化脚本

```
python .\scripts\init_env_win.py
```

```
C:\Users\Administrator\Downloads\blutter-main>python .\scripts\init_env_win.py
Downloading ICU library from https://github.com/unicode-org/icu/releases/download/release-73-2/icu4c-73_2-Win64-MSVC2019
.zip
Extracting ICU library
Downloading Capstone from https://github.com/capstone-engine/capstone/releases/download/4.0.2/capstone-4.0.2-win64.zip
Extracting Capstone library
Copying dlls to bin directory
Done

C:\Users\Administrator\Downloads\blutter-main>
```

6.要打开`x64 Native Tools Command Prompt`,它可以在`Visual Studio`文件夹中找到

7.把需要反编译的flutterapp用压缩包打开，提取v8a里的`libflutter.so`和`libapp.so`(现在基本上是64位)解压到blutter文件夹，并创建一个输出结果的文件夹

8.在刚才打开的x64窗口运行下面的命令(全局代{过}{滤}理!)，等待运行完后会在output文件下生成一些脚本信息
PS:blutter目前支持最新的版本的dart快照解析，如果这个跑不起来可以参考第四步手动配置

```
pip install pyelftools -i https://pypi.tuna.tsinghua.edu.cn/simple

python blutter.py ..\arm64-v8a\ .\output
时间比较慢
```

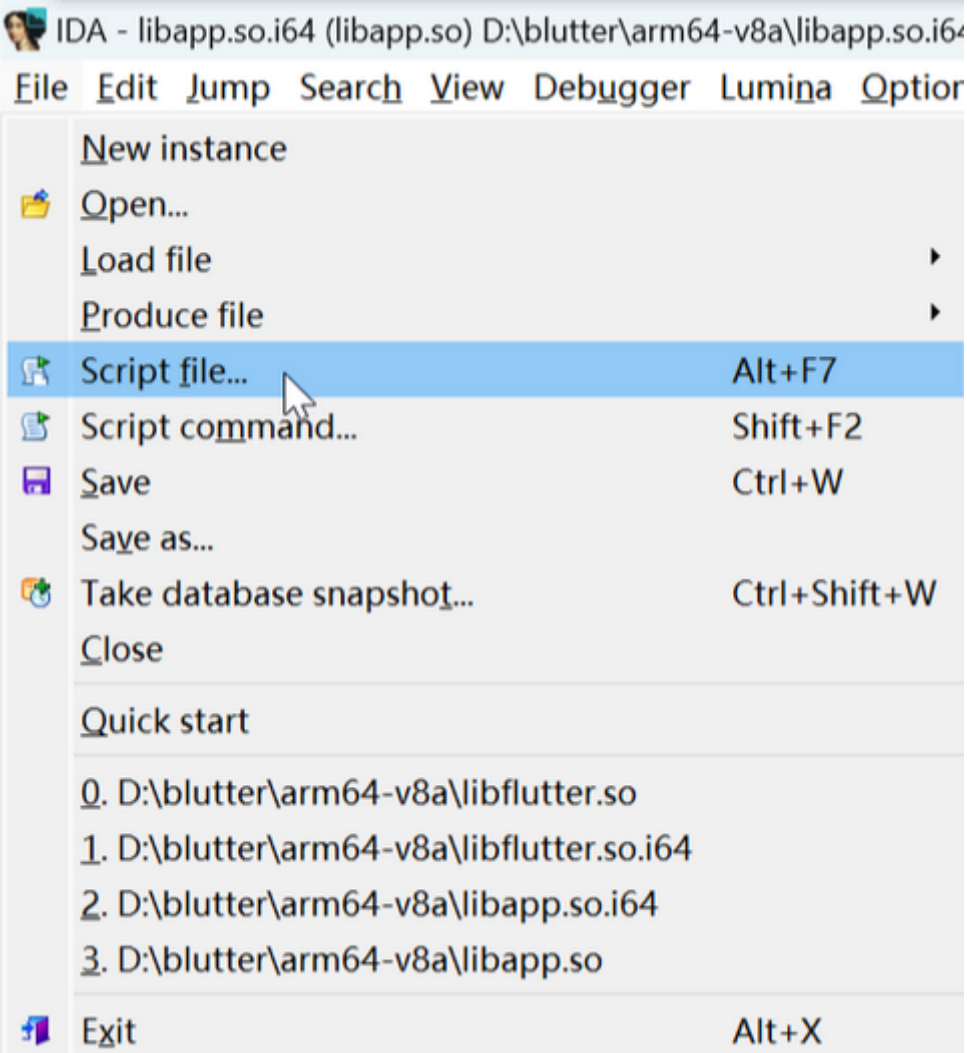
```
otice] A new release of pip is available: 23.2.1 -> 25.0.1
otice] To update, run: python.exe -m pip install --upgrade pip

\Users\Administrator\Downloads\blutter-main>
\Users\Administrator\Downloads\blutter-main>\python blutter.py .\arm64-v8a\ .\output
python' 不是内部或外部命令，也不是可运行的程序
批处理文件。

\Users\Administrator\Downloads\blutter-main>python blutter.py .\arm64-v8a\ .\output
rt version: 3.1.5, Snapshot: 7dbbeeb8ef7b91338640dca3927636de, Target: android arm64
ags: product no-code_comments no-dwarf_stack_traces_mode no-lazy_dispatchers dedup_instructions no-ass
 compressed-pointers null-safety
oning into 'C:\Users\Administrator\Downloads\blutter-main\dartsdk\v3.1.5'...
mote: Enumerating objects: 2487, done.
mote: Counting objects: 100% (2487/2487), done.
mote: Compressing objects: 100% (2040/2040), done.
ceiving objects:  53% (1328/2487), 596.00 KiB | 27.00 KiB/s
```

查看out put目录

```
asm 对dart语言的反编译结果，里面有很多dart源代码的对应偏移
ida_script so文件的符号表还原脚本
blutter_frida.js目标应用程序的 frida 脚本模板
objs.txt对象池中对象的完整（嵌套）转储,对象池里面的方法和相应的偏移量
pp.txt对象池中的所有 Dart 对象

###如果有加密秘钥可以 直接pp.txt文件中搜索，例如AES
```

名称	修改日期	类型	大小
asm	2024/7/29 16:56	文件夹	
ida_script	2024/7/29 16:56	文件夹	
blutter_frida.js	2024/7/29 16:56	JavaScript 源文件	421 KB
objs.txt	2024/7/29 16:56	文本文档	513 KB
pp.txt	2024/7/29 16:56	文本文档	3,078 KB

9.接下来ida加载libapp.so，然后ida左上角点击file，再点击Script file加载符号解析脚本



10.至此可以看到so里的相关函数以显现

File Edit Jump Search View Debugger Lumina Opt

Library function Regular function Instruction

Functions

Function name	Seg
enjoyreadingdbmark_note_MarkNoteDB__getMar···	.tex
enjoyreading$reader$web_server_WebServer__ge···	.tex
enjoyreading$reader$web_server_WebServer___i···	.tex
enjoyreading$reader$webview_reader_BrowserCo···	.tex
IsType_BrowserControllerProvider_Stub_6b1b38	.tex
enjoyreading$util$native_channel_NativeChann···	.tex
enjoyreading$store$reader_settings_provider_···	.tex
IsType_ReaderSettingProvider_Stub_6b1c18	.tex
enjoyreading$store$reader_settings_provider_···	.tex
enjoyreading$store$reader_settings_provider_···	.tex
enjoyreading$store$preference_Preference__ge···	.tex
shared_preferences$shared_preferences_Shared···	.tex
IsType_FontFileType_Stub_6b2258	.tex
AllocateReaderSettingProviderStub_6b2278	.tex
enjoyreading$reader$native_reader3__NativeRe···	.tex
enjoyreading$reader$native_reader3__NativeRe···	.tex
enjoyreading$reader$native_reader3__NativeRe···	.tex
enjoyreadingdbarticle_ArticleDB__getSummar···	.tex
IsType__BottomAppBar_Stub_6b27ec	.tex
fluttersrccupertino$activity_indicator__Cu···	.tex
fluttersrccupertino$button__CupertinoButto···	.tex
fluttersrccupertino$checkbox__CupertinoChe···	.tex
fluttersrccupertino$switch__CupertinoSwitc···	.tex
fluttersrccupertino$text_selection_toolbar···	.tex
fluttersrcmaterial$button___RawMaterialBut···	.tex
fluttersrcmaterial$button___RawMaterialBut···	.tex
fluttersrcmaterial$button___RawMaterialBut···	.tex
fluttersrcmaterial$button___RawMaterialBut···	.tex
IsType_RawMaterialButton_Stub_6b352c	.tex
IsType_MaterialState_Stub_6b354c	.tex

文件 编辑 跳转 搜索 视图 调试器 Lumina 选项 窗口 帮助 BinDiff

无调试器

库函数 常规函数 指令 数据 未知 外部符号 Lumina 函数

Functions

函数名称	段
FfiAsyncCallbackSendStub_4a4074	.text
CloneSuspendStateStub_4a40a8	.text
AsyncExceptionHandlerStub_4a418c	.text
SuspendSyncStarAtYieldStub_4a41f0	.text
SuspendSyncStarAtStartStub_4a4368	.text
InitSyncStarStub_4a44e4	.text
ReturnAsyncStarStub_4a450c	.text
YieldAsyncStarStub_4a4538	.text
InitAsyncStarStub_4a46b4	.text
ReturnAsyncNotFutureStub_4a46dc	.text
ReturnAsyncStub_4a4708	.text
ResumeStub_4a4734	.text
AwaitWithTypeCheckStub_4a47d4	.text
AwaitStub_4a4958	.text
NoSuchMethodDispatcherStub_4a4ad4	.text
InstantiateTypeArgumentsMayShareFunctionTAStub…	.text
InstantiateTypeArgumentsMayShareInstantiatorTA…	.text
InstantiateTypeArgumentsStub_4a4e64	.text
InstantiateTypeLegacyFunctionTypeParameterStub…	.text
InstantiateTypeNullableFunctionTypeParameterSt…	.text
sub_4A50BC	.text
InstantiateTypeLegacyClassTypeParameterStub_4a…	.text
InstantiateTypeNullableClassTypeParameterStub_…	.text
sub_4A5208	.text
InstantiateTypeStub_4a5240	.text
InstanceOfStub_4a5278	.text
AssertBooleanStub_4a52b4	.text
InitInstanceFieldStub_4a5430	.text
InitLateFinalStaticFieldStub_4a5464	.text
InitStaticFieldStub_4a5508	.text
FfiCallbackTrampolineStub_4a553c	.text
sub_4A5544	.text
sub_4A554C	.text
sub_4A5554	.text
sub_4A555C	.text
sub_4A5564	.text
sub_4A556C	.text
sub_4A5574	.text
sub_4A557C	.text
sub_4A5584	.text
sub_4A558C	.text
sub_4A5594	.text
sub_4A559C	.text
sub_4A55A4	.text
sub_4A55AC	.text
sub_4A55B4	.text
sub_4A55BC	.text
sub_4A55C4	.text
sub_4A55CC	.text
sub_4A55D4	.text
sub_4A55DC	.text
sub_4A55E4	.text
sub_4A55EC	.text
sub_4A55F4	.text
sub_4A55FC	.text
sub_4A5604	.text
sub_4A560C	.text
sub_4A5614	.text
sub_4A561C	.text
sub_4A5624	.text
sub_4A562C	.text
sub_4A5634	.text
sub_4A563C	.text
sub_4A5644	.text
sub_4A564C	.text
sub_4A5654	.text
sub_4A565C	.text

IDA View-A | Hex View-1 | Enums

```
LOAD:0000000000000000                                   ;
LOAD:0000000000000000                                   ; +-------------------------------------------------------------
LOAD:0000000000000000                                   ; |      This file was generated by The Interactive Disassembl
LOAD:0000000000000000                                   ; |           Copyright (c) 2022 Hex-Rays, <support@hex-rays.c
LOAD:0000000000000000                                   ; +-------------------------------------------------------------
LOAD:0000000000000000                                   ;
LOAD:0000000000000000                                   ; Input SHA256 : 21C4CD956545176E76F71FE746FF294F054B15A207439
LOAD:0000000000000000                                   ; Input MD5    : E1CCAF39970A72B51D5DC1440FFA27D6
LOAD:0000000000000000                                   ; Input CRC32  : 7554A0CF
LOAD:0000000000000000
LOAD:0000000000000000                                   ; File Name   : D:\nixiang\安卓逆向这档事\022番外实战篇3-拨云见E
LOAD:0000000000000000                                   ; Format      : ELF64 for ARM64 (Shared object)
LOAD:0000000000000000                                   ;
LOAD:0000000000000000
LOAD:0000000000000000                                   ; Processor       : ARM
LOAD:0000000000000000                                   ; ARM architecture: metaarm
LOAD:0000000000000000                                   ; Target assembler: Generic assembler for ARM
LOAD:0000000000000000                                   ; Byte sex        : Little endian
LOAD:0000000000000000
LOAD:0000000000000000                                   ; ===========================================================
LOAD:0000000000000000
LOAD:0000000000000000                                   ; Segment type: Pure data
LOAD:0000000000000000                                   AREA LOAD, DATA, READONLY, ALIGN=0
LOAD:0000000000000000 7F 45 4C 46                       DCD 0x464C457F                  ; File format: \x7FELF
LOAD:0000000000000004 02                                DCB 2                           ; File class: 64-bit
LOAD:0000000000000005 01                                DCB 1                           ; Data encoding: littl
LOAD:0000000000000006 01                                DCB 1                           ; File version
LOAD:0000000000000007 00                                DCB 0                           ; OS/ABI: UNIX System
LOAD:0000000000000008 00                                DCB 0                           ; ABI Version
LOAD:0000000000000009 00 00 00 00 00 00 00              DCB 0, 0, 0, 0, 0, 0, 0         ; Padding
LOAD:0000000000000010 03 00                             DCW 3                           ; File type: Shared ob
LOAD:0000000000000012 B7 00                             DCW 0xB7                        ; Machine: ARM64
LOAD:0000000000000014 01 00 00 00                       DCD 1                           ; File version
LOAD:0000000000000018 00 00 00 00 00 00 00 00           DCQ 0                           ; Entry point
LOAD:0000000000000020 40 00 00 00 00 00 00 00           DCQ 0x40                        ; PHT file offset
LOAD:0000000000000028 F0 00 C1 00 00 00 00 00           DCQ 0xC100F0                    ; SHT file offset
LOAD:0000000000000030 00 00 00 00                       DCD 0                           ; Processor-specific f
LOAD:0000000000000034 40 00                             DCW 0x40                        ; ELF header size
LOAD:0000000000000036 38 00                             DCW 0x38                        ; PHT entry size
LOAD:0000000000000038 07 00                             DCW 7                           ; Number of entries in
LOAD:000000000000003A 40 00                             DCW 0x40                        ; SHT entry size
LOAD:000000000000003C 0B 00                             DCW 0xB                         ; Number of entries in
LOAD:000000000000003E 0A 00                             DCW 0xA                         ; SHT entry index for
LOAD:0000000000000040                                   ; ELF64 Program Header
LOAD:0000000000000040                                   ; PHT Entry 0
LOAD:0000000000000040 06 00 00 00                       dword_40 DCD 6                  ; DATA XREF: LOAD:0000
LOAD:0000000000000040                                                                   ; .text:00000000004AA0
LOAD:0000000000000040                                                                   ; .text:000000000004AB4
LOAD:0000000000000040                                                                   ; Type: PHDR
LOAD:0000000000000044 04 00 00 00                       DCD 4                           ; Flags
LOAD:0000000000000048 40 00 00 00 00 00 00 00           DCQ 0x40                        ; File offset
LOAD:0000000000000050 40 00 00 00 00 00 00 00           DCQ dword_40                    ; Virtual address
LOAD:0000000000000058 40 00 00 00 00 00 00 00           DCQ 0x40                        ; Physical address
LOAD:0000000000000060 88 01 00 00 00 00 00 00           DCQ 0x188                       ; Size in file image
LOAD:0000000000000068 88 01 00 00 00 00 00 00           DCQ 0x188                       ; Size in memory image
LOAD:0000000000000070 08 00 00 00 00 00 00 00           DCQ 8                           ; Alignment
LOAD:0000000000000078                                   ; PHT Entry 1
LOAD:0000000000000078 01 00 00 00                       DCD 1                           ; Type: LOAD
LOAD:000000000000007C 04 00 00 00                       DCD 4                           ; Flags
LOAD:0000000000000080 00 00 00 00 00 00 00 00           DCQ 0                           ; File offset
LOAD:0000000000000088 00 00 00 00 00 00 00 00           DCQ 0                           ; Virtual address
LOAD:0000000000000090 00 00 00 00 00 00 00 00           DCQ 0                           ; Physical address
LOAD:0000000000000098 A8 0C 4A 00 00 00 00 00           DCQ 0x4A0CA8                    ; Size in file image
LOAD:00000000000000A0 A8 0C 4A 00 00 00 00 00           DCQ 0x4A0CA8                    ; Size in memory image
LOAD:00000000000000A8 00 40 00 00 00 00 00 00           DCQ 0x4000                      ; Alignment
LOAD:00000000000000B0                                   ; PHT Entry 2
```