网易易盾应用加固
技术分享

反编译安全威胁

攻击方式

- 文件反编译：DEX文件、 SDK文件、SO文件、资源文件
- 代码分析：Java代码、C\C++代码、JS\HTML代码
- 逆向破解：调试、抓包、HOOK注入、绕过签名校验等

安全威胁

- 逆向分析：代码调试，漏洞挖掘，协议分析
- 二次打包： APP盗版仿冒，插入广告、病毒木马，修改资源等
- 功能破解：VIP，会员，内购破解，去广告等

攻击案例

仿冒应用


17:00
病毒查杀
危险
发现4个风险项，建议处理
铁路12306
恶意软件
处理
12306订票助手
支付风险
处理
12306抢票专家
支付风险
处理
12306火车订票系统
支付风险
处理
17:03
风险项详情
12306抢票专家
支付风险
需立即处理
风险等级
高
感染病毒
a.expense.regtaobao.a.[盗信僵尸]
风险详情
该病毒启动后私自发送短信注册淘宝帐号，监听二次确认短信内容自动发送验证码，并将用户手机信息发送到159****1368,可能会给您的手机安全造成一定的威胁。
卸载

攻击案例

二次打包+破解会员

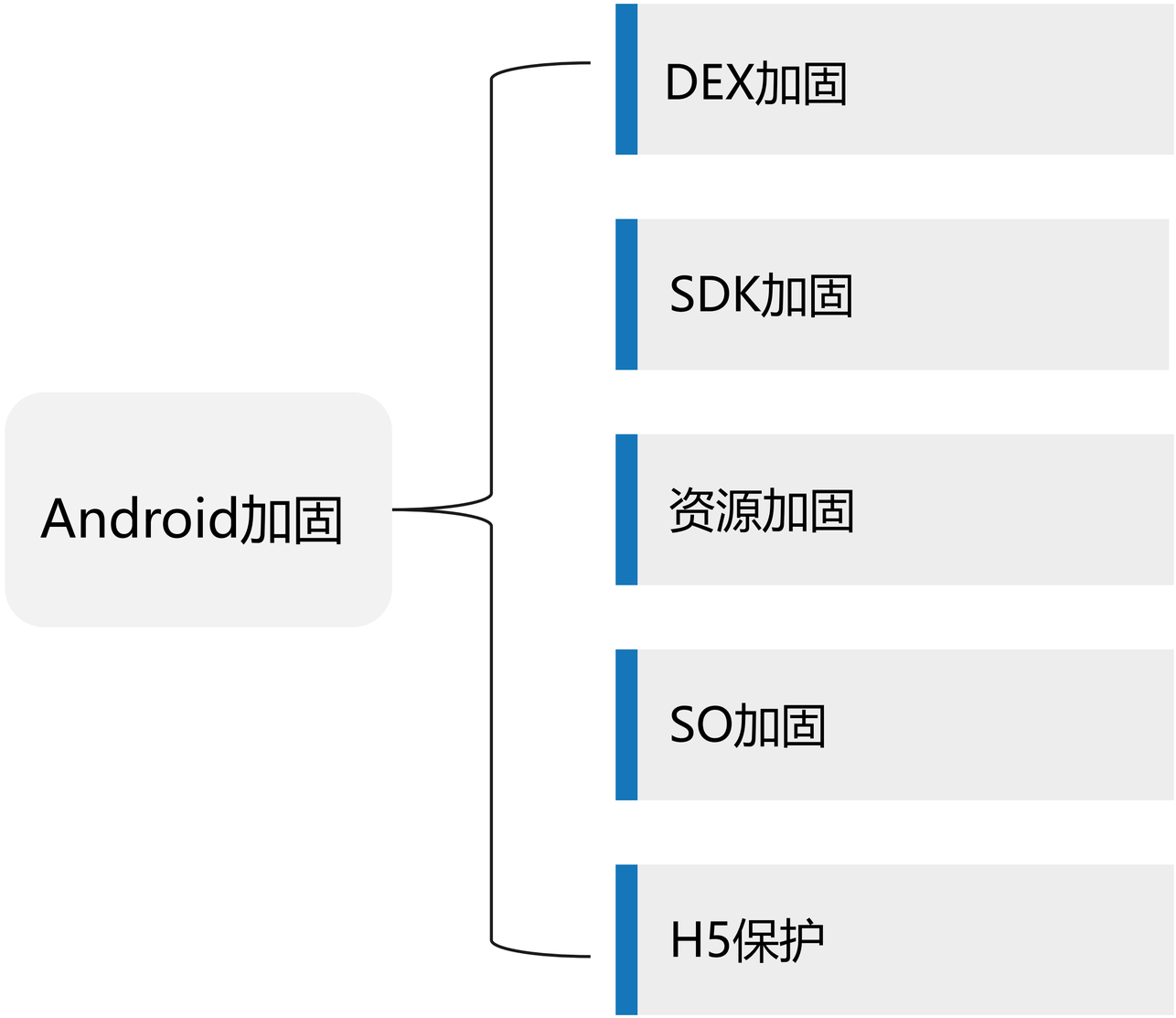

```
public class ReadLeftTimeRecord
{
  private long mDayReadTimes;
  private long mDayReadWords;
  private long mLeftTime;
  private final long mTimestamp;
  private final long mTradeEndTime;

  public ReadLeftTimeRecord(long paramLong1, long paramLong2, long paramLong3)
  {
    this.mTimestamp = paramLong1;
    this.mLeftTime = paramLong2;
    this.mTradeEndTime = paramLong3;
  }

  public ReadLeftTimeRecord(long paramLong1, long paramLong2, long paramLong3, long paramLong4, long paramLong5)
  {
    this(paramLong1, paramLong2, paramLong3);
    this.mDayReadTimes = paramLong4;
    this.mDayReadWords = paramLong5;
  }

  public native ReadLeftTimeRecord copy();

  public native long getDayReadTimes();

  public native long getDayReadWords();

  public long getLeftTime()
  {
    return 9999L;
  }

  public native long getTimestamp();

  public long getTradeEndTime()
  {
    return 99999L;
  }

  public boolean hasPaidService()
  {
    return true;
  }

  public native void setDayReadTimes(long paramLong);

  public native void setDayReadWords(long paramLong);

  public native void setLeftTime(long paramLong);
}
```

Android应用加固

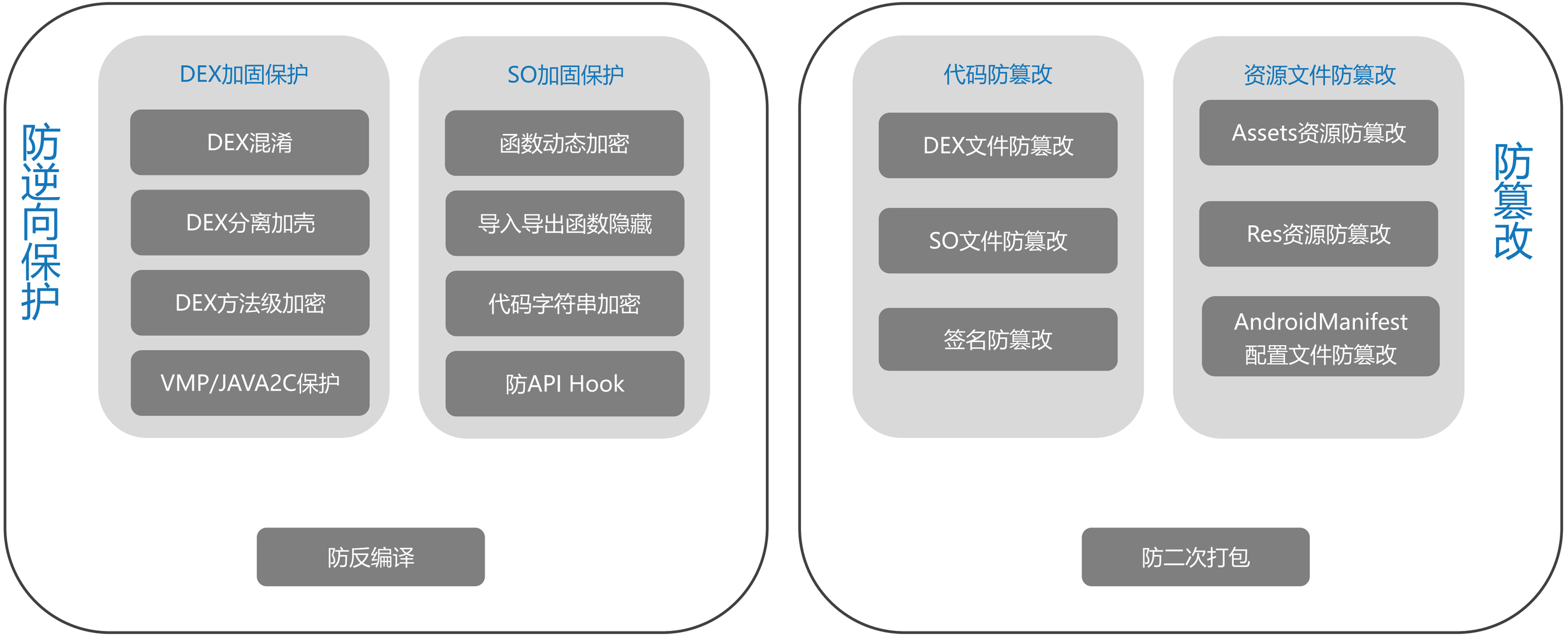
Android—加固功能清单

功能	说明
DEX加固	对DEX文件进行加壳防护，防止被静态反编译工具破解获取源码
防二次打包	应用在被非法二次打包后不能正常运行
防调试器	防止通过使用调试器工具对应用进行非法破解
内存防 dump	防止运行时在内存中dump数据
资源文件保护	加密资源文件, 防止APK资源文件被破解
H5文件混淆	对JS/HTML代码文件进行保护，防止破解分析
SDK加固	对Jar/AAR文件进行保护，防止反编译获取源码
SO 加密保护	对SO进行加壳，保护native代码不被逆向分析

网易易盾解决方案



Android加固主要功能



Dex加固

Dex加固效果—保护classes.dex

加固前

名称	大小	压缩后大小	类型	修	CRC32
..			文件夹		
assets			文件夹		
com			文件夹		
fabric			文件夹		
lib			文件夹		
META-INF			文件夹		
properties			文件夹		
res			文件夹		
AndroidManifest.xml	57,644	9,387	XML 文档		A13CE97A
classes.dex	8,134,124	3,232,664	DEX 文件		25EBE4AE
classes2.dex	3,909,028	1,556,037	DEX 文件		480280FA
miui_push_version	93	88	文件		579AEF8E
push_version	44	45	文件		0357FAE5
resources.arsc	1,327,864	1,327,864	ARSC 文件		BA3BA0C7

加固后

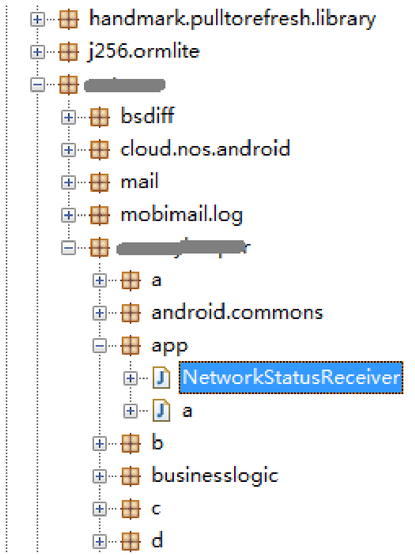
名称	大小	压缩后大小	类型	修改时间	CRC32
..			文件夹		
res			文件夹		
properties			文件夹		
META-INF			文件夹		
lib			文件夹		
fabric			文件夹		
com			文件夹		
assets			文件夹		
simplelogger.properties	177	98	Properties 源文件	1980/1/1 0:00	D226F51D
resources.arsc	1,327,864	1,327,864	ARSC 文件	1980/1/1 0:00	BA3BA0C7
push_version	44	45	文件	1980/1/1 0:00	0357FAE5
miui_push_version	93	88	文件	1980/1/1 0:00	579AEF8E
classes.dex	4,779,468	4,697,079	DEX 文件	2017/2/28 17:44	173264D5
AndroidManifest.xml	57,848	8,920	XML 文档	2017/2/28 17:44	7EAD1070

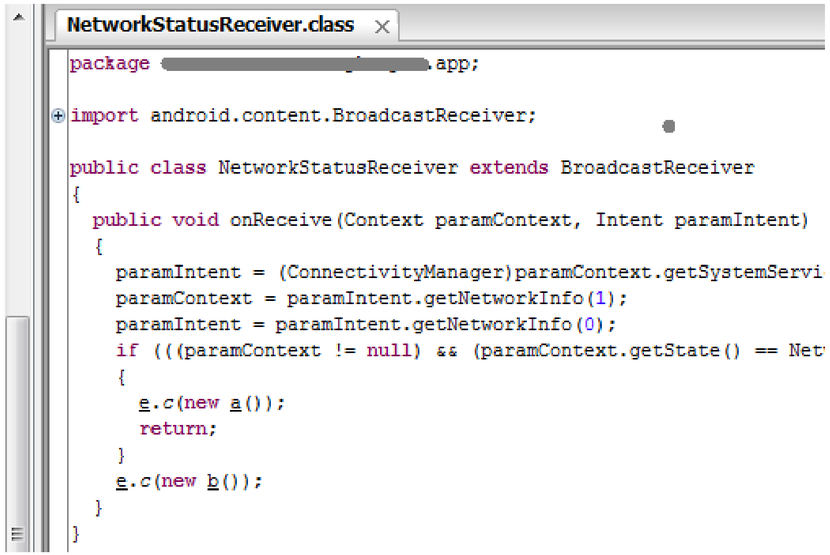
加固前的classes.dex和classes2.dex，直接暴露给了分析者。

加固后只有一个classes.dex，用baksmali.jar/jeb等工具查看，无法查看原app代码逻辑，同时保护classes.dex与classes2.dex。

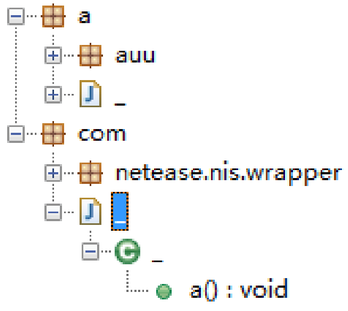
Dex加固效果—防止classes.dex被逆向分析

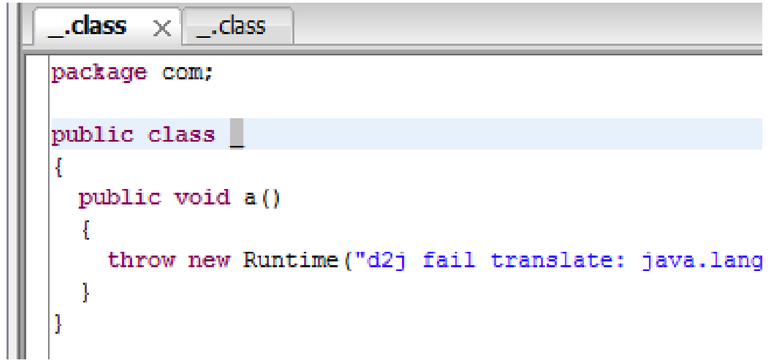
加固前





加固后





加固前很容易反编译出jar包文件查看源码
加固后让反编译失败，并无从查看源码。

Dex加固效果—代码逻辑混淆替换

加固前

```
public void onCreate() {
    h.a("app.onCreate");
    super.onCreate();
    [redacted]rApplication.d = this;
    [redacted]rApplication.a = System.currentTimeMillis();
    [redacted]rApplication.b = [redacted]Application.a;
    this.e = Thread.getDefaultUncaughtExceptionHandler();
    Thread.setDefaultUncaughtExceptionHandler(((Thread$UncaughtExceptionHandler)this));
    b.a.a.a.c.a(this.c, new i[]{new Crashlytics()});
    [redacted].android.commons.a.a.a(((Context)this));
    [redacted].android.commons.a.a.a(false);
    [redacted].android.commons.a.a.b("release".equals("dev"));
    [redacted].android.commons.a.a.a("[redacted]");
    [redacted].android.commons.a.a.c("[redacted]");
```

加固前重要初始化函数onCreate逻辑反编译后可见

加固后

```
public void onCreate() {
    try {
        super.onCreate();
        if(MyApplication.c != null) {
            Long v0_1 = Long.valueOf(System.currentTimeMillis());
            MyJni.run(MyApplication.b.getBaseContext(), MyApplication.c);
            MyApplication.c.onCreate();
            new StringBuilder().append(a.c("HhwWHCRQACwDBkg=")).append(Lor
        }

        Context v0_2 = this.getApplicationContext();
        MyApplication.a = v0_2;
        MyJni.c(v0_2);
    }
    catch(Exception v0) {
        v0.printStackTrace();
    }
```

加固后onCreate逻辑反编译后原逻辑已经不可见

DEX加固效果—VMP保护效果

```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130903040);
  this.tv_hello = ((TextView)findViewById(2131230720));
  this.btn_crash = ((Button)findViewById(2131230721));
  this.btn_copy = ((Button)findViewById(2131230722));
  this.btAdd = ((Button)findViewById(2131230723));
  this.btUpdate = ((Button)findViewById(2131230724));
  this.btDelete = ((Button)findViewById(2131230725));
  this.btClear = ((Button)findViewById(2131230726));
  this.tv_hello.setText(MyApplication.ApplicationLog);
  this.btAdd.setOnClickListener(new View.OnClickListener()
  {
    public void onClick(View paramAnonymousView)
    {
      new ContactsMgr(MainActivity.this).queryPhoneNumber("abc");
    }
  });
  this.btDelete.setOnClickListener(new View.OnClickListener()
  {
    public void onClick(View paramAnonymousView)
    {
      paramAnonymousView = new ContactsMgr(MainActivity.this);
      try
      {
        paramAnonymousView.deleteContact("大学同学0");
        return;
      }
      catch (Exception paramAnonymousView)
      {
        paramAnonymousView.printStackTrace();
      }
    }
  });
```

```
public native void copyFile(String paramString1, String paramString2);

public native void msg(String paramString);

protected native void onCreate(Bundle paramBundle);

private native String getContactID(String paramString);

public native void addContact(String paramString1, String paramString2);

public native void clearContacts();

public native void deleteContact(String paramString)
  throws Exception;

public native Map<Integer, String> queryPhoneNumber(String paramString);

public native void updateContact(String paramString1, String paramString2);
```

DEX加固效果—VMP保护指令替换

```
enum Opcode {
    OP_REM_FLOAT                    = 0x0,
    OP_ADD_LONG_2ADDR               = 0x1,
    OP_INT_TO_SHORT                 = 0x2,
    OP_INVOKE_SUPER_RANGE           = 0x3,
    OP_USHR_LONG                    = 0x4,
    OP_INVOKE_INTERFACE_RANGE       = 0x5,
    OP_SHR_LONG_2ADDR               = 0x6,
    OP_INVOKE_VIRTUAL_RANGE         = 0x7,
    OP_SPUT_BOOLEAN                 = 0x8,
    OP_MOVE_RESULT_WIDE             = 0x9,
    OP_INVOKE_DIRECT_RANGE          = 0xa,
    OP_MUL_FLOAT                    = 0xb,
    OP_INT_TO_DOUBLE                = 0xc,
    OP_RETURN_VOID_BARRIER          = 0xd,
    OP_AGET_BYTE                    = 0xe,
    OP_IF_GEZ                       = 0xf,
    OP_REM_LONG_2ADDR               = 0x10,
    OP_INVOKE_SUPER_QUICK_RANGE     = 0x11,
    OP_ADD_INT_2ADDR                = 0x12,
    OP_MUL_INT_2ADDR                = 0x13,
    OP_SUB_LONG_2ADDR               = 0x14,
    OP_DIV_INT                      = 0x15,
    OP_MOVE_OBJECT_FROM16           = 0x16,
    OP_IPUT_CHAR                    = 0x17,
    OP_SHR_INT_LIT8                 = 0x18,
    OP_AND_INT_2ADDR                = 0x19,
    OP_IF_NE                        = 0x1a,
    OP_DOUBLE_TO_LONG               = 0x1b,
    OP_NEG_DOUBLE                   = 0x1c,
```

指令替换

```
0x3 = 0x43
0x5f = 0xe2
0xc0 = 0x7d
0xbf = 0x10
0xb8 = 0x59
0xe5 = 0x21
0xcf = 0x3f
0x15 = 0x61
0x73 = 0x60
0x1d = 0xba
0xfa = 0xd6
0xe9 = 0xde
0xf = 0x79
0xe0 = 0x2a
0x32 = 0x95
0x52 = 0x9d
0xfe = 0xa4
0x13 = 0x97
0xa2 = 0x55
0x41 = 0xb9
0x40 = 0x62
0x1e = 0xd8
0x42 = 0xa2
0x57 = 0xd2
0x3b = 0xf
0xdf = 0xc4
0xb4 = 0x94
0x7f = 0xa0
0x8 = 0x16
```

DEX加固效果—VMP执行流程

DEX加固效果—Java2c保护效果

```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130968607);
  n = this;
  paramBundle = (LinearLayout)findViewById(2131493022);
  Object localObject = (LinearLayout)findViewById(2131493030);
  LinearLayout localLinearLayout1 = (LinearLayout)findViewById(2131493026);
  LinearLayout localLinearLayout2 = (LinearLayout)findViewById(2131493028);
  LinearLayout localLinearLayout3 = (LinearLayout)findViewById(2131493036);
  LinearLayout localLinearLayout4 = (LinearLayout)findViewById(2131493024);
  LinearLayout localLinearLayout5 = (LinearLayout)findViewById(2131493034);
  LinearLayout localLinearLayout6 = (LinearLayout)findViewById(2131493032);
  a(paramBundle, SysInfoActivity.class, 0);
  a((LinearLayout)localObject, BatteryActivity.class, 1);
  a(localLinearLayout1, AnyLocationActivity.class, 2);
  a(localLinearLayout2, DirList.class, 3);
  a(localLinearLayout3, SpeedTest.class, 4);
  a(localLinearLayout4, CaptureActivity.class, 5);
  a(localLinearLayout5, ProcessActivity.class, 6);
  a(localLinearLayout6, SettingsActivity.class, 7);
  if (com.cn.zh.device.c.a.a("DeviceInfoSetting", "firstLaunch") == 0)
  {
    paramBundle = Integer.toString(1);
    localObject = n.getSharedPreferences("DeviceInfoSetting", 0).edit();
    ((SharedPreferences.Editor)localObject).putString("firstLaunch", paramBundle);
    ((SharedPreferences.Editor)localObject).commit();
    Toast.makeText(this, getString(2131099812), 0).show();
  }
  this.q = new a();
  this.p = ((LinearLayout)findViewById(2131493021));
  new Thread(new com.cn.zh.device.settings.a(this, this)).start();
  CrashHandler.getInstance();
  CrashHandler.init(getApplicationContext(), null);
}

public boolean onKeyDown(int paramInt, KeyEvent paramKeyEvent)
{
  boolean bool = false;
  long l;
  if (paramInt == 4)
  {
    l = System.currentTimeMillis();
    if ((this.r == 0L) || (l - this.r >= 2000L))
    {
      Toast.makeText(this, getString(2131099721), 0).show();
      this.r = System.currentTimeMillis();
      bool = true;
    }
  }
  else
  {
    return bool;
  }
  finish();
  this.r = l;
  return false;
}
```

```
private native void a(LinearLayout paramLinearLayout, Class<?> paramClass, int paramInt);

public static native Context c();

public final void a(int paramInt)
{
}

public final native boolean b(String paramString);

protected native void onCreate(Bundle paramBundle);

public native boolean onKeyDown(int paramInt, KeyEvent paramKeyEvent);

protected native void onPause();

protected native void onResume();
```

DEX加固效果—Java2c保护效果

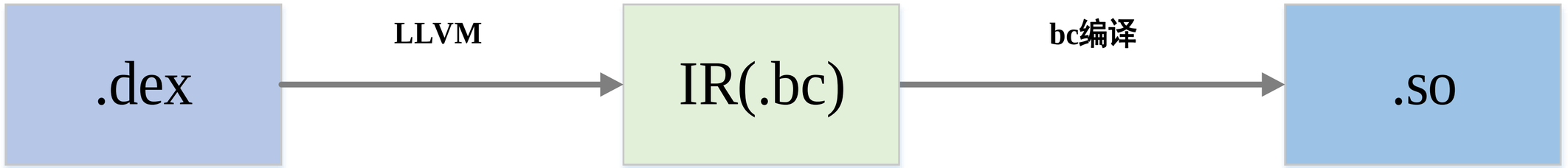
```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130968607);
  n = this;
  paramBundle = (LinearLayout)findViewById(2131493022);
  Object localObject = (LinearLayout)findViewById(2131493030);
  LinearLayout localLinearLayout1 = (LinearLayout)findViewById(2131493026);
  LinearLayout localLinearLayout2 = (LinearLayout)findViewById(2131493028);
  LinearLayout localLinearLayout3 = (LinearLayout)findViewById(2131493036);
  LinearLayout localLinearLayout4 = (LinearLayout)findViewById(2131493024);
  LinearLayout localLinearLayout5 = (LinearLayout)findViewById(2131493034);
  LinearLayout localLinearLayout6 = (LinearLayout)findViewById(2131493032);
  a(paramBundle, SysInfoActivity.class, 0);
  a((LinearLayout)localObject, BatteryActivity.class, 1);
  a(localLinearLayout1, AnyLocationActivity.class, 2);
  a(localLinearLayout2, DirList.class, 3);
  a(localLinearLayout3, SpeedTest.class, 4);
  a(localLinearLayout4, CaptureActivity.class, 5);
  a(localLinearLayout5, ProcessActivity.class, 6);
  a(localLinearLayout6, SettingsActivity.class, 7);
  if (com.cn.zh.device.c.a.a("DeviceInfoSetting", "firstLaunch") == 0)
  {
    paramBundle = Integer.toString(1);
    localObject = n.getSharedPreferences("DeviceInfoSetting", 0).edit();
    ((SharedPreferences.Editor)localObject).putString("firstLaunch", paramBundle);
    ((SharedPreferences.Editor)localObject).commit();
    Toast.makeText(this, getString(2131099812), 0).show();
  }
  this.q = new a();
  this.p = ((LinearLayout)findViewById(2131493021));
  new Thread(new com.cn.zh.device.settings.a(this, this)).start();
  CrashHandler.getInstance();
  CrashHandler.init(getApplicationContext(), null);
}

public boolean onKeyDown(int paramInt, KeyEvent paramKeyEvent)
{
  boolean bool = false;
  long l;
  if (paramInt == 4)
  {
    l = System.currentTimeMillis();
    if ((this.r == 0L) || (l - this.r >= 2000L))
    {
      Toast.makeText(this, getString(2131099721), 0).show();
      this.r = System.currentTimeMillis();
      bool = true;
    }
  }
  else
  {
    return bool;
  }
  finish();
  this.r = l;
  return false;
}
```

Java2c

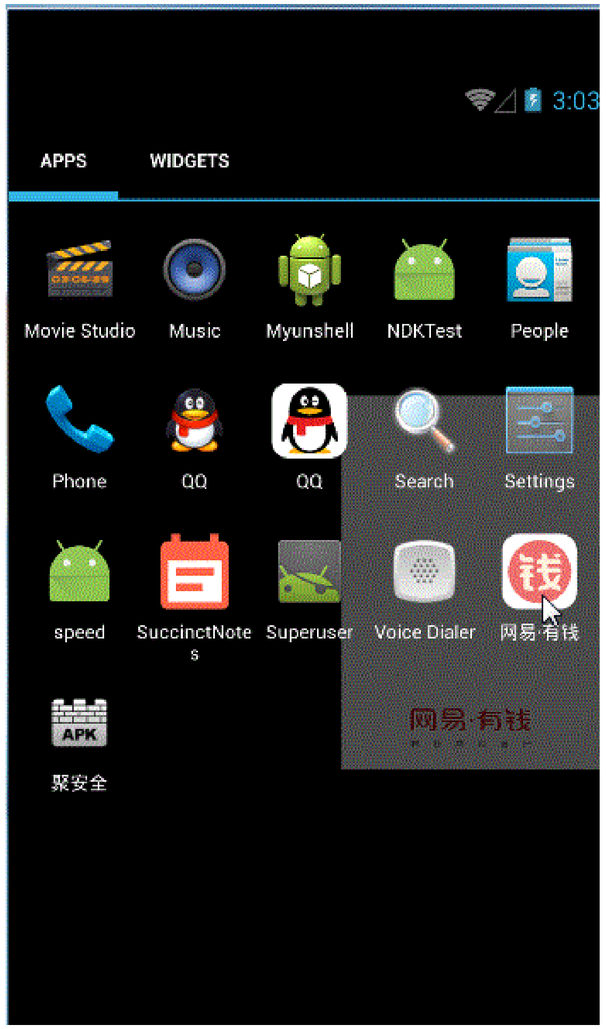
```
.text:0000AA88                         EXPORT Java_com_cn_zh_device_MainActivity_onCreate__Landroid_os_Bundle_2
.text:0000AA88 Java_com_cn_zh_device_MainActivity_onCreate__Landroid_os_Bundle_2
.text:0000AA88                                                 ; DATA XREF: LOAD:stru_1E8↑o
.text:0000AA88 ; __unwind {
.text:0000AA88                 STMFD   SP!, {R4-R11,LR}
.text:0000AA8C                 ADD     R11, SP, #0x1C
.text:0000AA90                 SUB     SP, SP, #0x29C
.text:0000AA94                 SUB     R3, R11, #-var_68
.text:0000AA98                 ADD     R12, R3, #0x30
.text:0000AA9C                 ADD     LR, R3, #0x2C
.text:0000AAA0                 ADD     R4, R3, #0x28
.text:0000AAA4                 ADD     R5, R3, #0x24
.text:0000AAA8                 ADD     R6, R3, #0x20
.text:0000AAAC                 ADD     R7, R3, #0x1C
.text:0000AAB0                 ADD     R8, R3, #0x18
.text:0000AAB4                 ADD     R9, R3, #0x14
.text:0000AAB8                 ADD     R10, R3, #0x10
.text:0000AABC                 STR     R0, [R11,#var_6C]
.text:0000AAC0                 ADD     R0, R3, #0x34
.text:0000AAC4                 ADD     R3, R3, #0x38
.text:0000AAC8                 STR     R0, [R11,#var_70]
.text:0000AACC                 MOV     R0, #0xD
.text:0000AAD0                 STR     R0, [R11,#var_74]
.text:0000AAD4                 MOV     R0, #0x100
.text:0000AAD8                 STR     R0, [R11,#var_78]
.text:0000AADC                 MOV     R0, #0x1000
.text:0000AAE0                 STR     R0, [R11,#var_7C]
.text:0000AAE4                 LDR     R0, [R11,#var_74]
.text:0000AAE8                 STR     R1, [R11,#var_80]
.text:0000AAEC                 LDR     R1, [R11,#var_78]
.text:0000AAF0                 STR     R2, [R11,#var_84]
.text:0000AAF4                 LDR     R2, [R11,#var_7C]
.text:0000AAF8                 STR     R9, [R11,#var_88]
.text:0000AAFC                 STR     R3, [R11,#var_8C]
.text:0000AB00                 STR     R8, [R11,#var_90]
.text:0000AB04                 STR     R10, [R11,#var_94]
.text:0000AB08                 STR     R12, [R11,#var_98]
.text:0000AB0C                 STR     LR, [R11,#var_9C]
.text:0000AB10                 STR     R4, [R11,#var_A0]
.text:0000AB14                 STR     R5, [R11,#var_A4]
.text:0000AB18                 STR     R6, [R11,#var_A8]
.text:0000AB1C                 STR     R7, [R11,#var_AC]
.text:0000AB20                 BL      j_dvmInitReferenceTable
.text:0000AB24                 LDR     R1, =(aJava2cJavaComC_5 - 0x1315C8)
```

Java代码彻底转换成了本地指令

DEX加固效果—JAVA2C原理



DEX加固效果—防二次打包，让打包党却步



对加固后的APK进行二次打包，
安装运行后，直接闪退

DEX加固效果—防止动态调试

```
Debugger: attached to process /system/bin/app_process (pid=2230)
502F8000: loaded /data/app-lib/[illegible]
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGTRAP signal (Trace trap) (exc.code 5, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
403B8E84: got SIGINT signal (Interrupt) (exc.code 2, tid 2230)
50318D0A: got SIGSEGV signal (Segmentation violation) (exc.code b, tid 2230)
Command "MakeCode" failed
50318D0A: got SIGSEGV signal (Segmentation violation) (exc.code b, tid 2230)
Debugger: thread 2238 has exited (code 0)
```

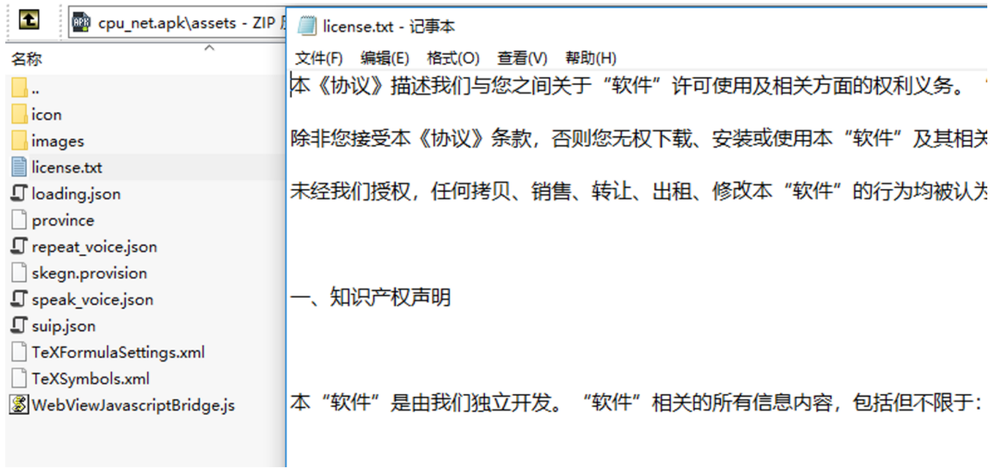
使用IDA Pro调试APP自动退出

加固后APK一旦被调试，APP异常退出，阻断调试。

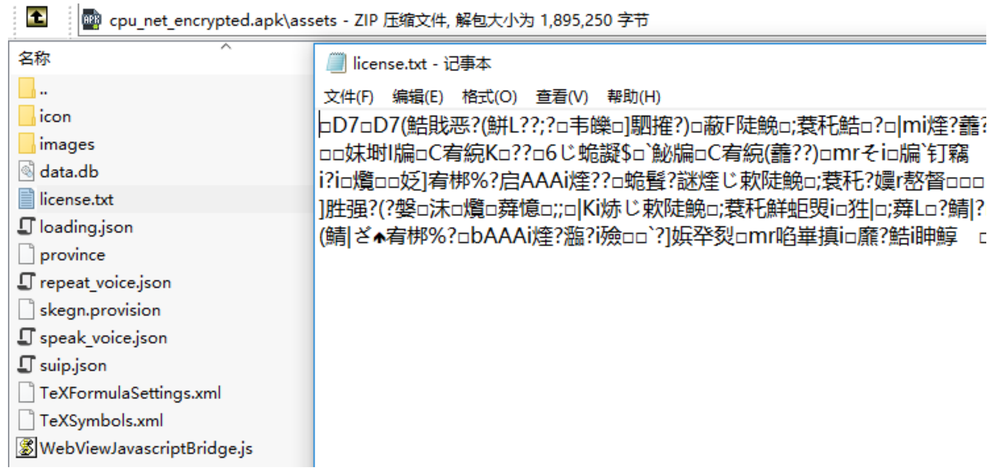
资源加固

APK加固效果—assets资源透明加解密

加固前



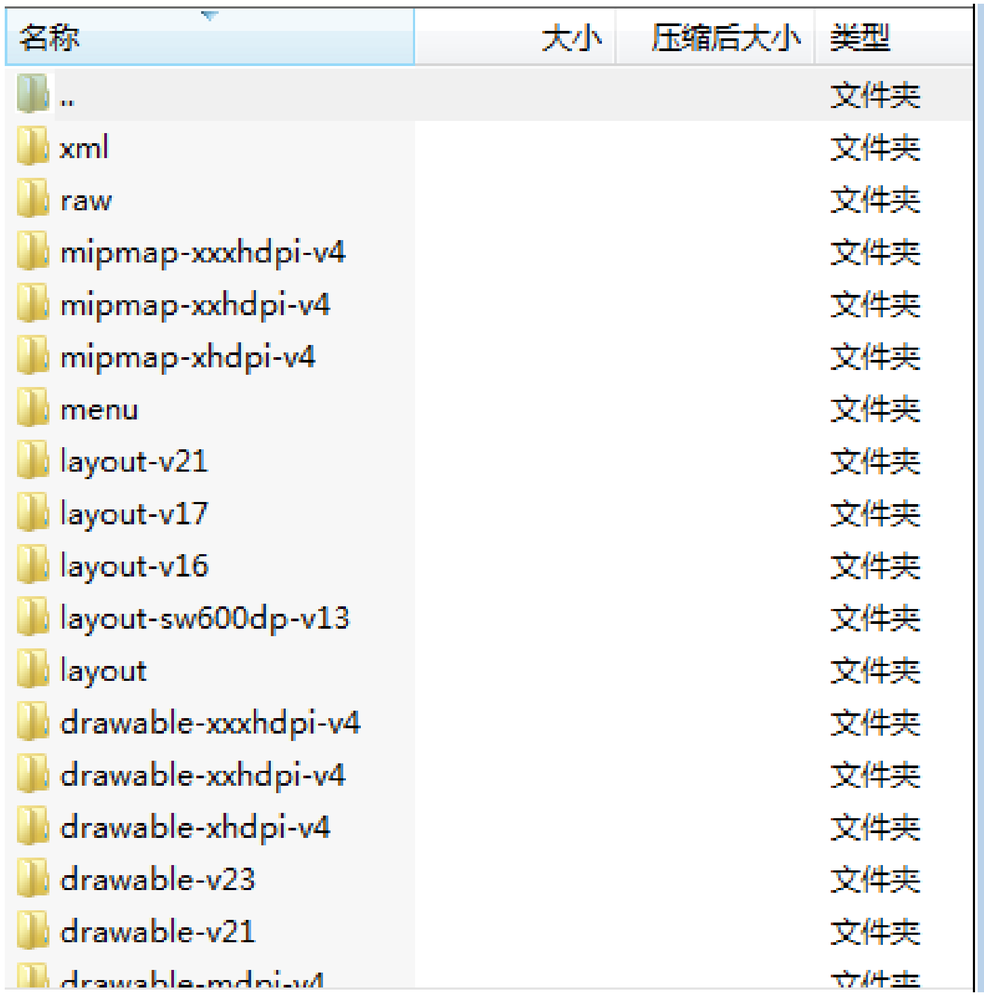
加固后



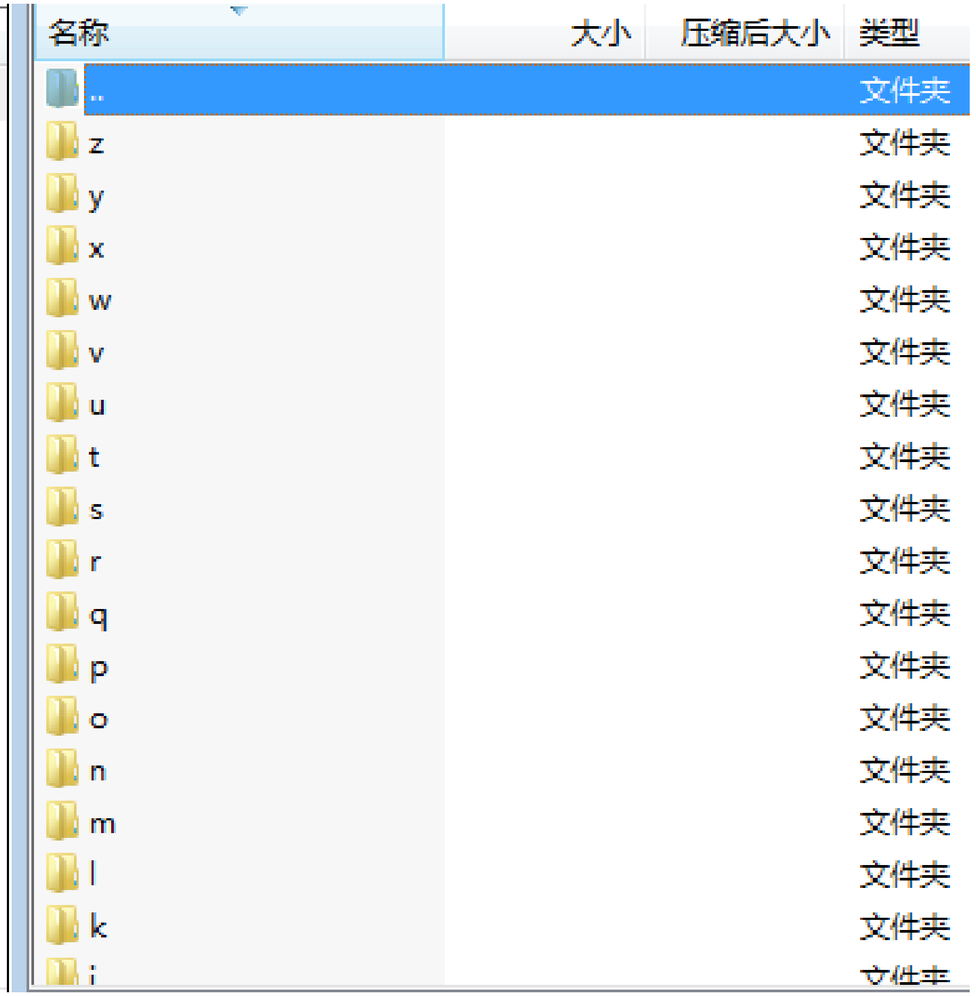
加固后APK里的assets资源文件被加密保护，使用时解密

APK加固效果—res资源混淆保护

加固前



名称	大小	压缩后大小	类型
..			文件夹
xml			文件夹
raw			文件夹
mipmap-xxxhdpi-v4			文件夹
mipmap-xxhdpi-v4			文件夹
mipmap-xhdpi-v4			文件夹
menu			文件夹
layout-v21			文件夹
layout-v17			文件夹
layout-v16			文件夹
layout-sw600dp-v13			文件夹
layout			文件夹
drawable-xxxhdpi-v4			文件夹
drawable-xxhdpi-v4			文件夹
drawable-xhdpi-v4			文件夹
drawable-v23			文件夹
drawable-v21			文件夹
drawable-mdpi-v4			文件夹



名称	大小	压缩后大小	类型
..			文件夹
z			文件夹
y			文件夹
x			文件夹
w			文件夹
v			文件夹
u			文件夹
t			文件夹
s			文件夹
r			文件夹
q			文件夹
p			文件夹
o			文件夹
n			文件夹
m			文件夹
l			文件夹
k			文件夹
j			文件夹

加固后

加固后APK里的res资源文件混淆，分析难度加大

H5保护效果—JS保护

保护前

```
if ($.mobile_download.isMobile()) {
    var language_arr = ['de','es','br','fr','it','nl'];
    var path_name = window.location.pathname;
    var language = path_name.substr(1,2);
    var link_url = '';
    $('a[href^="http://                    "],a[href^="http:/                              ]').each(function(){
        var download_url = $(this).attr('href');
        if(typeof(download_url) != 'undefined' ){
            if(download_url.indexOf('.dmg') > -1 || download_url.indexOf('.exe') > -1){
                if($.mobile_download.in_Array(language,language_arr)){
                    $(this).attr('href', 'https:/                              e.com/'+language+'/get-downloac
                }else{
                    language = 'en';
                    $(this).attr('href', 'https:/                              com/get-download-link-to-your
                }
                $.mobile_download.click($(this),language);
            }

        }
    });
}else{
    var filetypes = /\.(zip|exe|dmg)$/i;
```

保护后

保护方式：

- 字符串加密
- 混淆/去log/变量名混淆/函数名混淆
- 压缩
- 游戏保护/平台识别
- 防篡改/防加速

H5保护效果—html保护

保护前

```
<div class="header-btn-content">
  <ul class="if_btnWin">
    <li class="art_hidden"><a href="http://[redacted]m/vi
&#39;navigation&#39;, document.location.pathname]);" class="btn_outline_tr
    <li class="art_show"><a href="http://[redacted]om/[redacted]
document.location.pathname]);" class="btn_outline_try">Download</a></li>
    <li class="art_hidden"><a href="https://[redacted]e
&#39;navigation&#39;, document.location.pathname]);" class="btn_outline_bu
  </ul>
  <ul class="if_btnMac" style="display: none;">
    <li class="art_hidden"><a href="http://[redacted]/vi
&#39;navigation&#39;, document.location.pathname]);" class="btn_outline_tr
    <li class="art_show"><a href="http://[redacted]m/vide
&#39;navigation&#39;, document.location.pathname]);" class="btn_outline_tr
    <li class="art_hidden"><a href="https://[redacted]e
&#39;navigation&#39;, document.location.pathname]);" class="btn_outline_bu
  </ul>
</div>
<div class="logo"><a href="https://[redacted]/" cl
<div class="nav clickMenu_show_nav">
  <ul class="ul-block">
    <li><a href="https://[redacted]/">Product</a><
    <li><a href="https://[redacted]/support.html">
    <li><a href="https://[redacted]/[redacted]
    <li><a href="https://[redacted]/resource.html"
  </ul>
  <ul class="nav_pro">
```

保护方式

- 加密后内容分块
- 乱序插入干扰信息
- 压缩处理，去除注释无用属性等
- 平台识别，检测到非移动平台，不显示网页

保护后

```
FFQzAght="+PG1ldGEgaHR0cC1lcXVpdj0iQ";co=l('dyhsKGMpKTs=');EQOAUGjp="xLmJhaWR1LmNvbS8iPjxsaW5rIHJlbD0iZG5zLXByZWZldGNoIiBocmVmPSJodHRwczovL3Q";mhQyy60E=
"sx\rxSIxDC1xDC3xNAKvFFxETBxEMxESCxGSwdx!x#whxg_0x(dun.163.com";ZoMCrThj="tYWxpZ246Y2VudGVyO3Bvc2l0aW9uOmFic29sdXRlO2JvdHRvbToxNDBweDtoZW1naHQ";WbYYYvxu=
"29udGVudC1UeXBlIiBjb250ZW50PSJ0ZXh0L2h0bWw7IGNoYXJzZXQ";jYzXmAZF="tYWxpZ246bGVmdH0jcXJjb2R1IC5xcmNvZGUtdGV4dCBie2NvbG9yOiM2NjY7Zm9udC13ZW1naHQ";eval(l(a));
zNvZNgrZ="6NjBweDt3aWR0aDoxMDAlfSNxcmNvZGV7ZGlzcGxheTppbmxpbmUtYmxvY2t9I3FyY29kZSAucXJjb2R1LW10ZW17ZmxvYXQ";eYCCHPGR=
"zLmJhaWR1LmNvbS8iPjxsaW5rIHJlbD0iZG5zLXByZWZldGNoIiBocmVmPSJodHRwczovL3Q";GPxOPcpU="PCFETONUWVBFIGh0bWw+PGh0bWwgY2xhc3M9ImN5ZS1lbmFibGVkIGN5ZS1ubSI+PGhlYWQ";
kQVbyjFO=
"6bGVmdH0jcXJjb2R1IC5xcmNvZGUtaXR1bS0ye21hcmdpbi1sZWZ0OjMzcHh9I3FyY29kZSAucXJjb2R1LW1tZ3t3aWR0aDo2MHB4O2hlaWdodDo2MHB4fSNxcmNvZGUgLnFyY29kZS1pdGVtLTEgLnFyY29kZS1pbWd7YmFja2dyb3VuZDp1cmwoaHR0cHM6Ly9zczEuYmRzdGFOaWMuY29tLzVlTjFianE4Q";HbrqMOxr=
"VVZbTJ6Z29ZMOsvci93d3cvY2FjaGUvc3RhdGljL3Byb3RvY29sL2h0dHBzL2hvbWUvaW1nL3FyY29kZS9udW9taV8zNjV1YWJkLnBuZykgMCAwIG5vLXJlcGVhdH1AbWVkaWEgb25seSBzY3J1ZW4gYW5kICgtd2Via2l0LW1pbi1kZXZpY2UtcG14ZWwtcmF0aW86Mi17I3FyY29kZSAucXJjb2R1LW10ZW0tMSAucXJjb2R1LW1tYWdlOnVybChodHRwczovL3NzMS5iZHN0YXRpYy5jb20vNWVOMWJqcThBQ";qSVHsKtI="9IiMyOTMyZTEiPjxsaW5rIHJlbD0ic2hvcnRjdXQ";ZoMpoTGw="29ue2hlaWdodDolMHB4O3Bvc2l0aW9uOmFic29sdXRlO2JvdHRvbTo0N3B4O3R1eHQ";MQkhmXNC=
```

SDK/SO加固

SDK加固

从代码安全、文件安全等方面对SDK进行保护。对抗反编译手段，防止恶意篡改SDK、窃取用户隐私信息等，有效提升SDK保护的强度。

- SDK接口保持不变，对接入者透明
- 支持JAR包、AAR包
- 可灵活指定加固保护的类和函数
- 加固体积增量几乎无影响

SDK加固效果-Java代码保护

对JAR包的保护，我们采取将重要逻辑代码抽离保护的方案，在运行时再动态修复回去，这样接入者在开发阶段仍然可以依赖JAR包作为库文件直接引用接口函数，保持了接入者开发的透明。但是实际代码又是抽离的，我们测试时只选取了xxxxxxxx.jar中类com.test.app.view.ExampleShell的接口进行处理，那么接入者在打开JAR包时只能看到如下的效果：

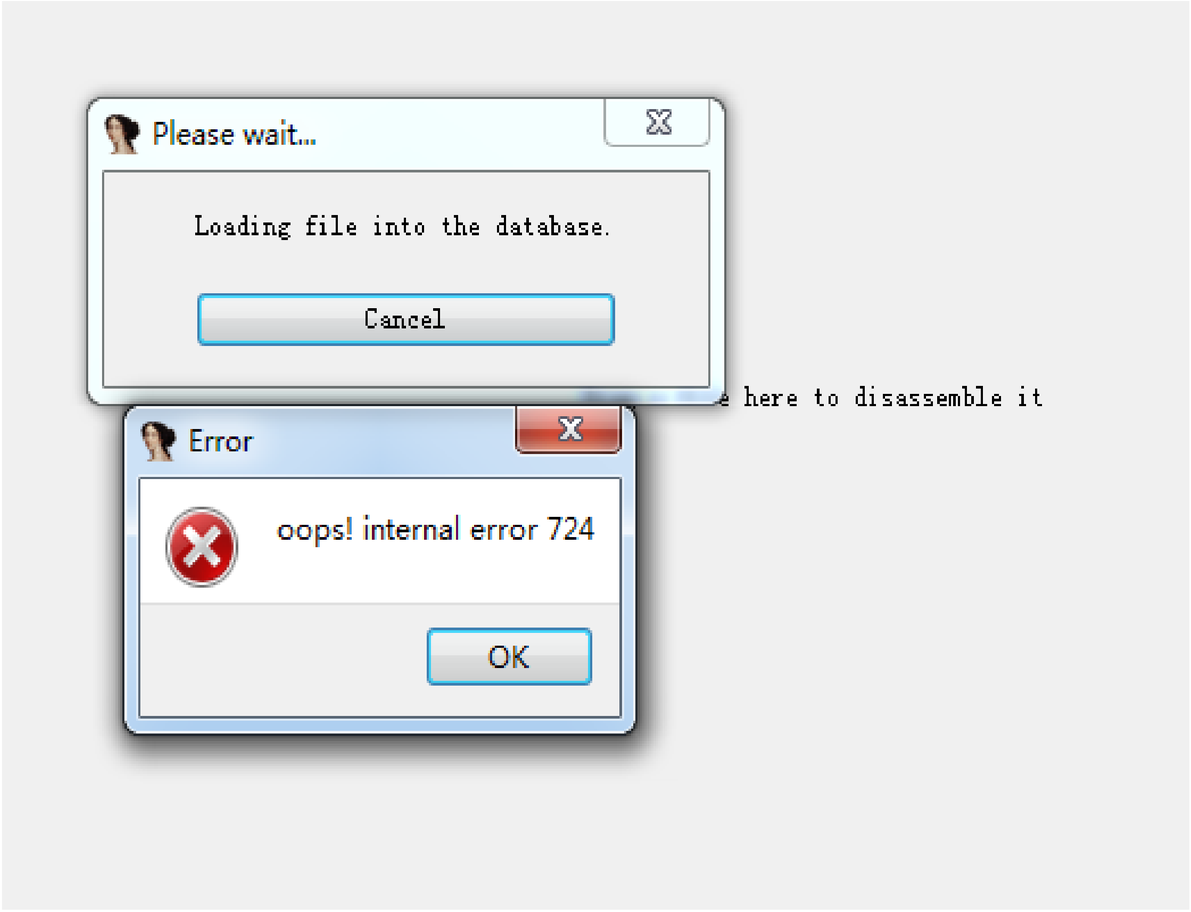
```
public static void InitShellApplicationContext(Context arg0, InitShellApplicationContextListener arg1)
{
  Utils.rL(new Object[] { 0, arg0, arg1, Integer.valueOf(13), Long.valueOf(1573811114836L) });
}

public static boolean IsInitializedComplete()
{
  Object object = Utils.rL(new Object[] { 0, Integer.valueOf(14), Long.valueOf(1573811114837L) });
  return ((Boolean)object).booleanValue();
}
```

函数的指令代码被隐藏保护了。相应的，其他类和函数也可以依此方案处理。

由于代码逻辑被隐藏掉了，接入者看不到代码逻辑，也就无法分析甚至阉割SDK的功能了。

SO加固—阻断IDA分析so及so代码加密

SO加固后，IDA将无法正常打开SO进行分析，IDA会显示如下报错:



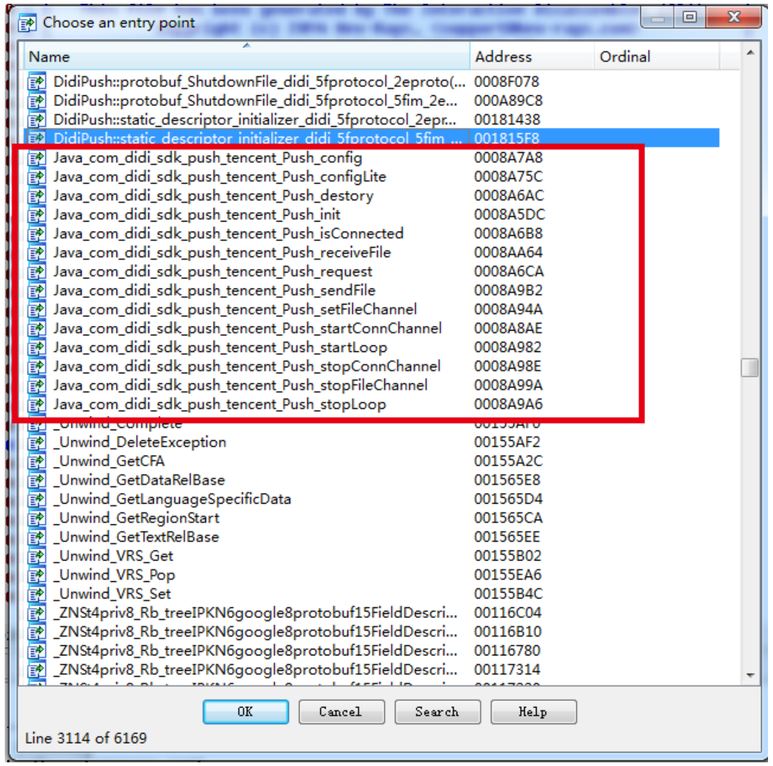
SO加固后，IDA将无法正常打开SO进行分析，IDA会显示如下报错。

左边是原始SO，右边是加固后的SO，加固前后已经没有任何相似性，代码和字符串都被加密。

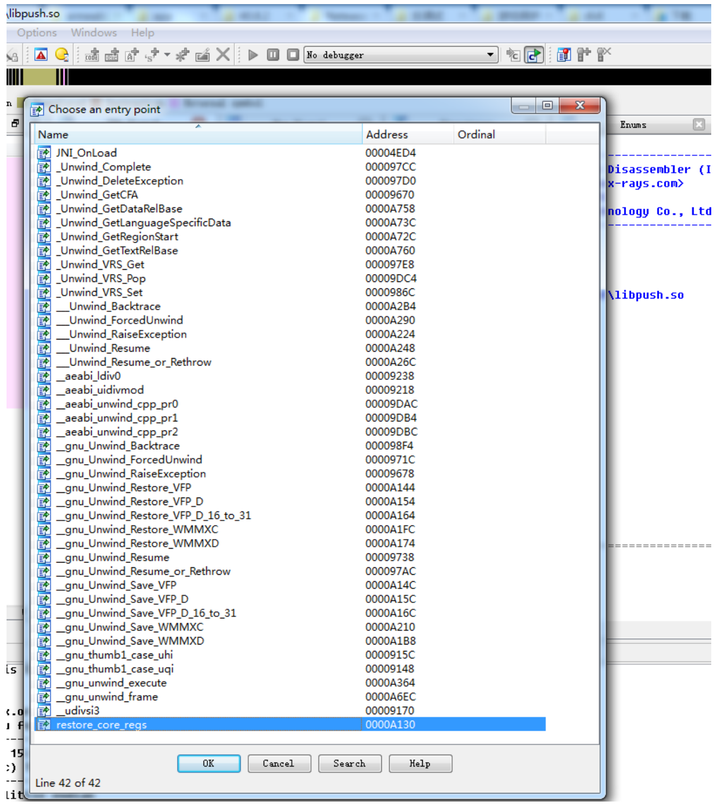
SO加固—SO导出函数隐藏

SO调用接口常常以Java_xxx_xxx_xxx形式导出，从接口名字即可大致看出该函数的功能。

SO加固可将该函数完全隐藏。(此处将反IDA静态分析功能去掉做示例对比)



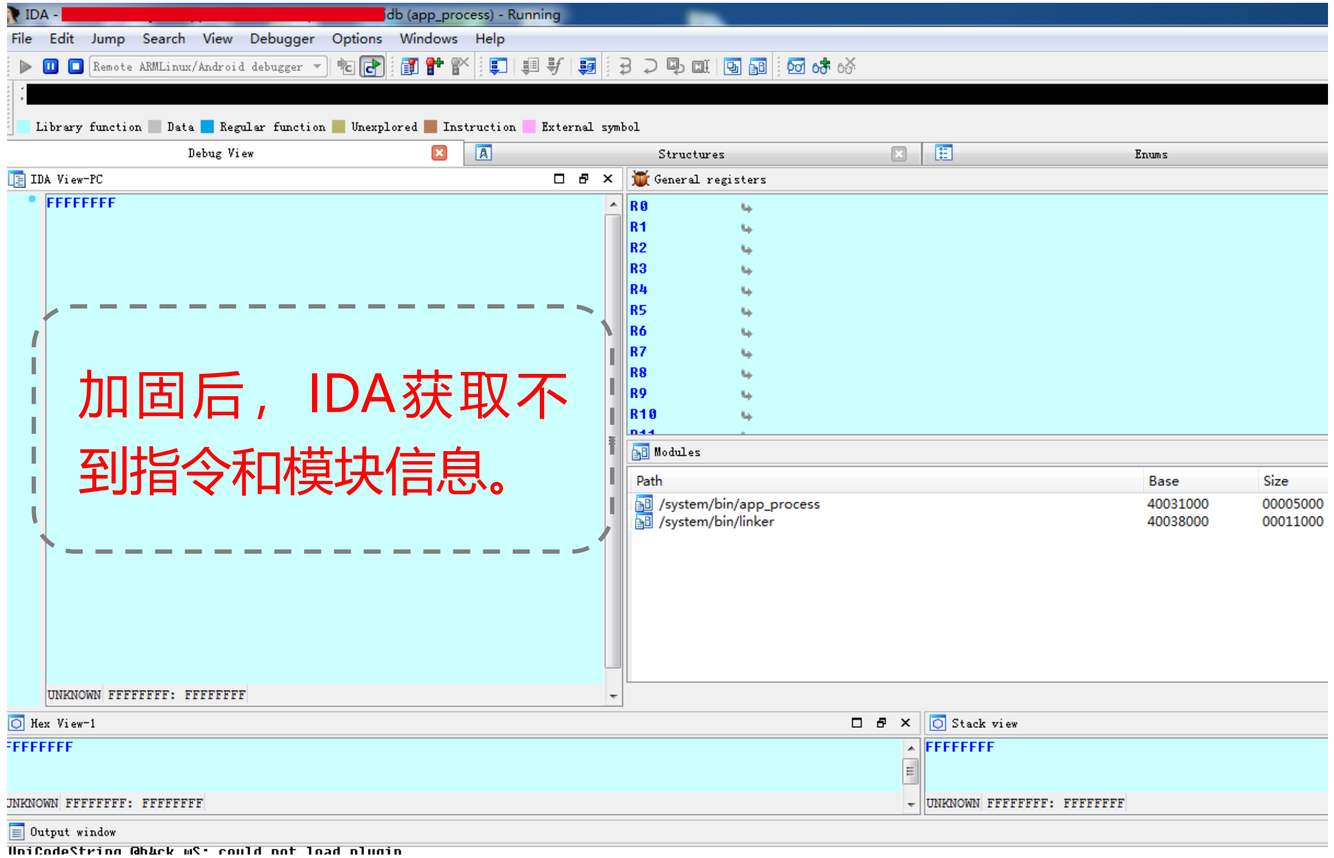
加固前，红框里的函数是Java层的调用接口



加固后，这些函数都被隐藏掉了

SO加固—SO导出函数隐藏

SO加固反调试效果，IDA attach到进程上后，IDA面板都为空获取不到指令和模块信息。



SO加固保护优点

优点	介绍
无需源码	只需要编译好的so即可加保护，相比于有些厂商提交源码的so保护方式，更为安全，有效保护厂商的技术机密
关键函数动态加密	一些关键函数运行完后，对其进行动态加密，提升安全性
防系统API HOOK	Hook一些的系统API，比如strcmp,strcpy等字符串操作函数就可以获取到很多关键信息。加保护后，这些hook将获取不到任何信息
全平台支持	支持armeabi、armeabi-v7a、x86、arm64-v8a、x86_64五大平台
无法脱壳	直接对ELF结构进行改造，仅在运行时存在可运行的代码，其它结构全部重构，任何时刻都不存在原始so的内存