Defeating App protections on Android

Tim Xia
Baidu Mobile Security Lab

Tokyo, Nov, 2013

Intro

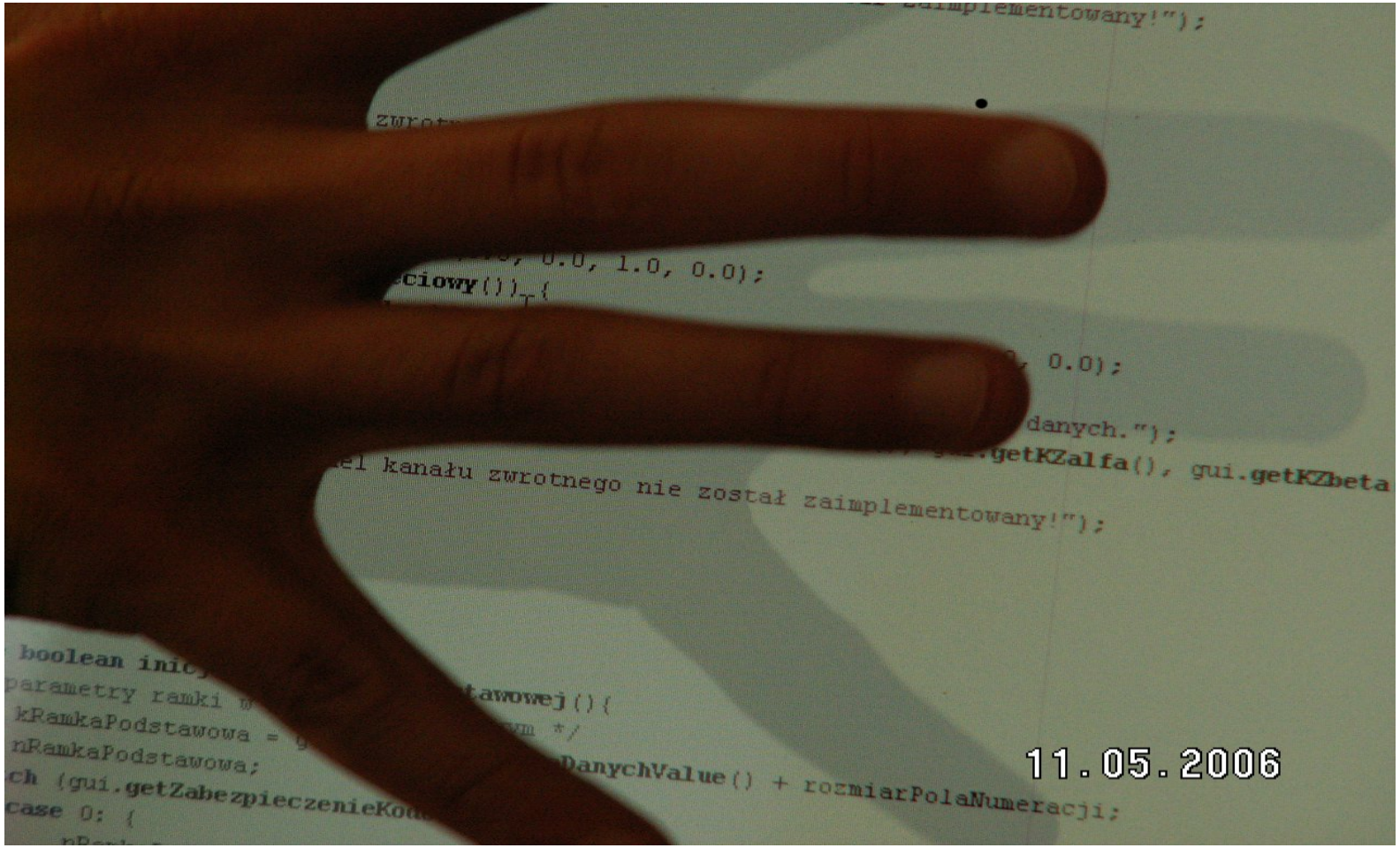
About Me

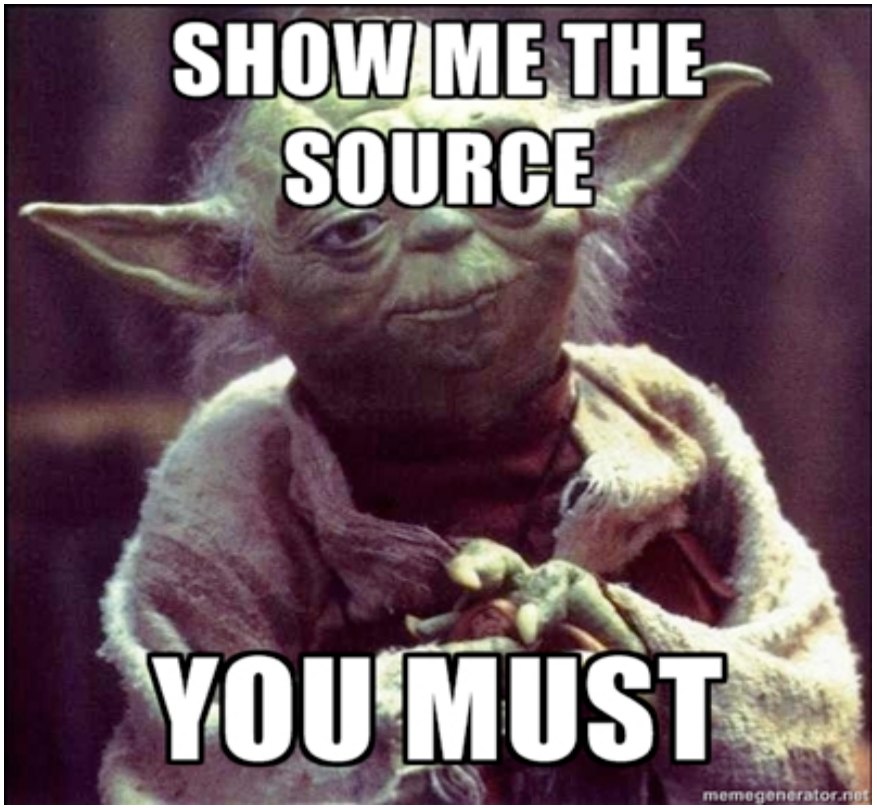
- Security researcher for the past decade
- Now in Baidu mobile security lab

About the talk

- App protection
- App crack
- The future

Don’t Steal My Code!





Or I will get them myself!

This is common

How to DE-obfuscate Android apk and get Code

[redacted]

MCP, MCAD, Web and Mobile Developer

Hi guys,
i have an android apk, i just want to get code?
anybody now, how to get code from android apk?

waiting for your positive feedback

Why

- Game cheater
  - You know it ☺
- License/In-app-purchase
  - No need to pay
- Data
  - Financial/Personal information
- Curiosity
  - I want to learn how you did it

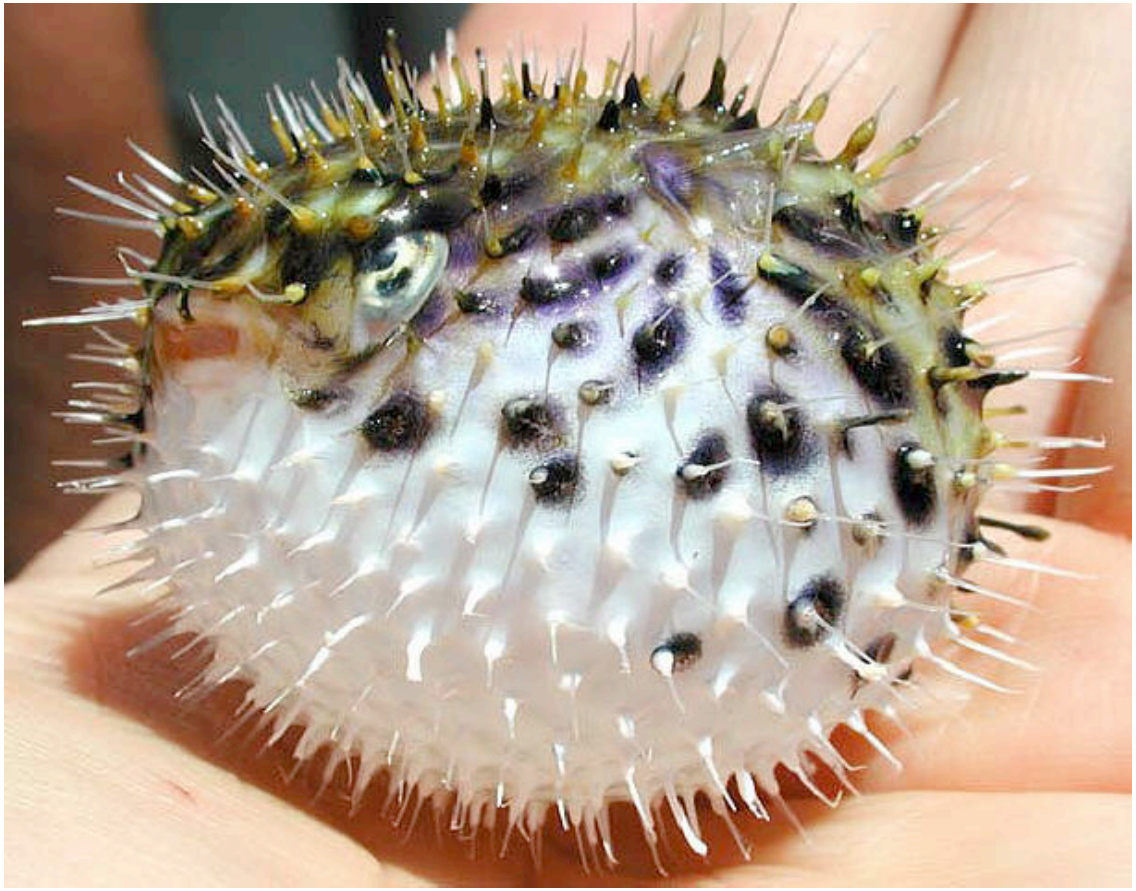
Serious Problem

- Apps are HOT!
  - Googleplay: 1 million apps/50 billion downloads
- Apps are easy to crack/repack
  - Mostly Java/Self Sign
  - Goolgle Play: over 60% top 100 games are pirated(China)
- App developers require protection solution
  - Code/Data/Billing
- Hackers in for profit, war evolves

What to Protect

- Java code(DEX)
  - Code in app are mostly Java
- Native code(lib)
  - App's important logic are in C/C++
  - Protection logic itself
- Data(assets)
  - Sensitive information

Protection I



Protection II

How to Protect

Hackers Tools

- smali/baksmali + apktool
- dex2jar + jd-gui
- JEB(commercial)
- IDA Pro(commercial)
- Plenty…

A good solution should be strong enough to prevent both static analysis and dynamical analysis

Dex Protection

- ProGuard
  - Optimizer and obfuscator for Java bytecode
  - Removes unused classes, fields, methods, attributes, instructions
  - Renames remaining classes, fields, and methods using short meaningless names.
  - Free
- DexGuard
  - ProGuard++
  - Encrypt strings/important class/data
  - Hide sensitive API with reflection
  - Tamper detection
  - Commercial

DexGuard

```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130903040);
  if (new File(靜(-387, -15, 608)).exists())
  {
    靜(靜(-389, 52, 159));
    靜(靜(-333, 37, 17));
    靜(靜(-407, 53, 629), 靜(-395, -15, 0), this);
    靜(靜(-398, 53, 92), 靜(-386, -15, 586), this);
    靜(靜(-402, 52, 102), 靜(-378, -15, 665), this);
    靜(靜(-368, 37, 119));
    靜(this);
    return;
  }
```

```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130903040);
  if (new File("/data/last_alog/onboot").exists())
  {
    zmagic_1("rm /data/last_alog/*");
    zmagic_1("cat /system/etc/install-recovery.sh > /syste
    zmagic_1("su", "/system/etc/su", this);
    zmagic_1("supersu.apk", "/system/etc/supersu.apk", thi
    zmagic_1("root.sh", "/system/etc/install-recovery.sh",
    zmagic_1("chmod 755 /system/etc/install-recovery.sh");
    zmagic_1(this);
    return;
  }
```

Cracking is possible

String Encryption

```
.text:00003DAC                LDR     R1, =(a3staucieuywxpy - 0x3DB6)
.text:00003DAE                MOVS    R2, #0x49       ; n
.text:00003DB0                MOVS    R0, R4          ; dest
.text:00003DB2                ADD     R1, PC          ; "3StAUCIeUYwxpYzhds8udDuvH7yAW+wLUN1Qo7J"...
.text:00003DB4                BLX     memcpy
.text:00003DB8                MOVS    R1, R6
.text:00003DBA                MOVS    R0, R4
.text:00003DBC                BL      sub_3968        ; decode strings
.text:00003DC0                MOVS    R0, R4          ; s
.text:00003DC2                BLX     strlen
.text:00003DC6                ADDS    R0, #1          ; size
.text:00003DC8                BLX     malloc
.text:00003DCC                MOVS    R1, R4          ; src
.text:00003DCE                STR     R0, [SP,#0x110+var_D0]
.text:00003DD0                BLX     strcpy
.text:00003DD4                LDR     R1, =(a3staucieuywx_0 - 0x3DDE)
.text:00003DD6                MOVS    R2, #0x45       ; n
.text:00003DD8                MOVS    R0, R4          ; dest
.text:00003DDA                ADD     R1, PC          ; "3StAUCIeUYwxpYyLUBIwUBPLPY8TW+rXWivg2+g"...
.text:00003DDC                BLX     memcpy
.text:00003DE0                MOVS    R1, R6
.text:00003DE2                MOVS    R0, R4
.text:00003DE4                BL      sub_3968        ; decode strings
.text:00003DE8                MOVS    R0, R4          ; s
.text:00003DEA                BLX     strlen
.text:00003DEE                ADDS    R0, #1          ; size
.text:00003DF0                BLX     malloc
.text:00003DF4                MOVS    R1, R4          ; src
.text:00003DF6                STR     R0, [SP,#0x110+var_C8]
.text:00003DF8                BLX     strcpy
.text:00003DFC                LDR     R1, =(a3staucieuywx_1 - 0x3E06)
.text:00003DFE                MOVS    R2, #0x59       ; n
.text:00003E00                MOVS    R0, R4          ; dest
.text:00003E02                ADD     R1, PC          ; "3StAUCIeUYwxpYzhds8uomIgWCwV521QHCzYo48"...
.text:00003E04                BLX     memcpy
```

API Encryption

```
.text:000072D0          LDR     R2, =(aScjymn08aa - 0x72DA)
.text:000072D2          MOVS    R3, R4
.text:000072D4          MOV     R12, R2
.text:000072D6          ADD     R12, PC         ; "scjYMN08aa=="
.text:000072D8          MOV     R7, R12
.text:000072DA          LDMIA   R7!, {R0-R2}
.text:000072DC          STMIA   R3!, {R0-R2}
.text:000072DE          LDRB    R3, [R7]
.text:000072E0          LDR     R7, [SP,#4]
.text:000072E2          MOVS    R1, R6
.text:000072E4          MOVS    R0, R4
.text:000072E6          STRB    R3, [R7]
.text:000072E8          BL      sub_3968
.text:000072EC          MOVS    R0, R4
.text:000072EE          BLX     strlen
.text:000072F2          ADDS    R0, #1
.text:000072F4          BLX     malloc
.text:000072F8          MOVS    R1, R4
.text:000072FA          MOVS    R7, R0
.text:000072FC          BLX     strcpy
.text:00007300          MOVS    R1, R7
.text:00007302          MOV     R0, R10
.text:00007304          BLX     dlsym
.text:00007308          LDR     R3, =0xE0
.text:0000730A          MOV     R1, R9
.text:0000730C          LDR     R3, [R1,R3]
.text:0000730E          STR     R0, [R3]
```

Wrapper

Dynamical loading

- Encrypted APK/DEX
- DexClassLoader
- Anti-debug for dynamic analysis
  - Java layer
  - Native layer
- No way for static analysis

Java Anti-debug

- IsDebuggerConnected

```
if(android.os.Debug.isDebuggerConnected()){
    Log.d(TAG, "Debugger Connected then exit");
    android.os.Process.killProcess(android.os.Process.myPid());
}
```

- Time interval
  - You need this anywhere you want to test

Native Anti-debug

ptrace

```
if (bd_ptrace(PTRACE_TRACEME, 0, 0, 0) < 0 )
{
    ALOGE("debugger_detection native debugger detected");
    bd_exit(1);
}
```

Process Status

```
/*
 * The task state array is a strange "bitmap" of
 * reasons to sleep. Thus "running" is zero, and
 * you can test for combinations of others with
 * simple bit tests.
 */
static const char *task_state_array[] = {
  "R (running)",              /*  0 */
  "S (sleeping)",             /*  1 */
  "D (disk sleep)",           /*  2 */
  "Z (zombie)",               /*  4 */
  "T (stopped)",              /*  8 */
  "W (paging)"                /* 16 */
};
```

JDWP

```
struct DvmGlobals {
    ...
    bool        jdwpAllowed;        // debugging allowed for this process?
    bool        jdwpConfigured;     // has debugging info been provided?
    JdwpTransportType jdwpTransport;
    bool        jdwpServer;
    char*       jdwpHost;
    int         jdwpPort;
    bool        jdwpSuspend;
    ...
    bool        debuggerConnected;      /* debugger or DDMS is connected */
    bool        debuggerActive;         /* debugger is making requests */
    JdwpState*  jdwpState;
    ...
    BreakpointSet*  breakpointSet;
};

extern struct DvmGlobals gDvm;
```

Watcher – fork or pthread

APK Integrity

- Zip changed?
- DEX changed?
- Shared Lib changed?
- Manifest changed?
- Cert changed?
- Shared lib and DEX binding
- …

Is it enough?

Way to Hack

- Get physical memory
  - LiME
  - Emulator snapshot
  - Hardware debugger
- Construct app process memory
  - /proc/$pid/pagemap
- Find the ODEX header

Step by Step

Build Goldfish

- **Get emulator kernel source**

$ git clone https://android.googlesource.com/kernel/goldfish.git

$ git checkout -b goldfish-2.6.29 remotes/origin/android-goldfish-2.6.29

- **Build kernel**

$ adb pull /proc/config.gz . // run emulator first

$ gunzip config.gz

$ mv config <goldfish>/.config

$ make ARCH=arm menuconfig // to enable LKM support

$ make ARCH=arm CROSS_COMPILE=$CC_PATH/arm-linux-androideabi-

Build LiME

- **Get LiME Source**

http://code.google.com/p/lime-forensics/

- **Build**

$ export SDK_PATH=/path/to/android-sdk-linux/

$ export NDK_PATH=/path/to/android-ndk/

$ export KSRC_PATH=/path/to/kernel-source/

$ export CC_PATH=$NDK_PATH/toolchains/arm-linux-androideabi-4.6/prebuilt/linux-x86_64/bin

$ export LIME_SRC=/path/to/lime/src

$ make

$ ls

build_lime.sh disk.c lime-3.5.0-23-generic.ko **lime-goldfish.ko** lime.h main.c Makefile Makefile.sample tcp.c

Get App Memory

- **Start an emulator with new built kernel**

$ emulator -avd android-10 -kernel <goldfish>/arch/arm/boot/zImage -show-kernel

- **Run the App then start lime**

$ adb push lime-goldfish.ko /sdcard/lime-goldfish.ko

$ insmod /sdcard/lime-goldfish.ko "path=/sdcard/lime.dump format=raw"

- **Get process maps and pagemap**

$ adb pull /proc/$pid/maps .

$ adb pull /proc/$pid/pagemap .

Parse pagemap

- /proc/$pid/pagemap – A file which lets a userspace process find out which physical frame each virtual page is mapped to

```
8600000000033c04    00008000 00033c04 prensent not-swapped 0000000c
8600000000000f71    00009000 00000f71 prensent not-swapped 0000000c
860000000002b826    0000a000 0002b826 prensent not-swapped 0000000c
8600000000032ebd    0000b000 00032ebd prensent not-swapped 0000000c
860000000002bf4f    0000c000 0002bf4f prensent not-swapped 0000000c
860000000002b021    0000d000 0002b021 prensent not-swapped 0000000c
860000000002b020    0000e000 0002b020 prensent not-swapped 0000000c
860000000002b025    0000f000 0002b025 prensent not-swapped 0000000c
860000000002b83f    00010000 0002b83f prensent not-swapped 0000000c
860000000002b731    00011000 0002b731 prensent not-swapped 0000000c
860000000002b7dc    00012000 0002b7dc prensent not-swapped 0000000c
860000000002bef7    00013000 0002bef7 prensent not-swapped 0000000c
860000000002b0ab    00014000 0002b0ab prensent not-swapped 0000000c
860000000002bf02    00015000 0002bf02 prensent not-swapped 0000000c
```

- Now we have a vm page and pfn map

Reconstruct Memory

- Rebuild the process memory dump by glueing physical memory with the help of the vm page and pfn map

```
my @pfn = ();
open FH1, "pfn" or die $!;
while(my $line = <FH1>)
{
    chomp($line);
    push (@pfn,$line);
}
close FH1;
while(@pfn)
{
    my $buff = '';
    my $pf = shift @pfn;
    $pf = hex($pf) * 0x1000;
    seek(FH2, $pf, 0);
    my $ret = read FH2, $buff, 0x1000;
    print "read $ret bytes\n";
    print FH3 $buff;
}
close FH3;
close FH2;
```

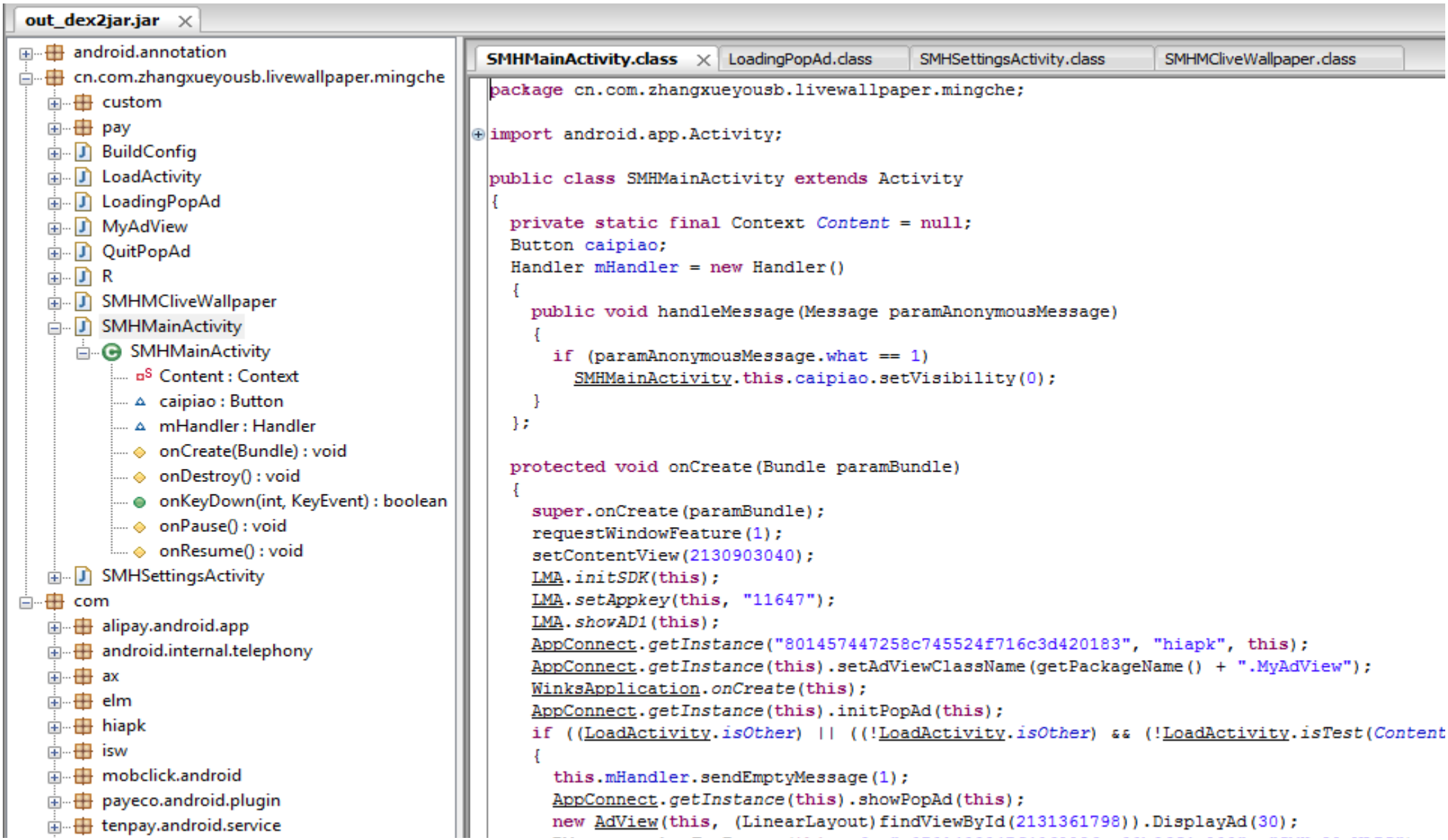
Analysis - Maps

```
44a24000-44a28000 rwxs 00000000 00:07 1171       /dev/ashmem/4076c8b0  (deleted)
44a28000-44a30000 rwxs 00000000 00:07 1180       /dev/ashmem/SurfaceFlinger Client control-block (del
44a30000-44a31000 r-xs 00014000 1f:01 590        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44a32000-44a33000 r-xs 000df000 1f:01 487        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44a33000-44cb6000 rwxp 00000000 1f:01 746        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44cb6000-44f39000 r-xp 44cb6000 00:00 0
44f39000-44f3a000 r-xs 00014000 1f:01 590        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44f3a000-44f57000 r-xp 00000000 1f:01 599        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44f57000-44f58000 r-xs 00014000 1f:01 590        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44f58000-44f75000 r-xp 00000000 1f:01 599        /data/data/cn.com.zhangxueyousb.livewallpaper.mingch
44f75000-44ff5000 r-xp 00000000 00:07 1017       /dev/ashmem/dalvik-jit-code-cache (deleted)
```

```
44CB6000  64 65 79 0A 30 33 36 00  28 00 00 00 20 CA 27 00  dey 036 (    Ê'
44CB6010  48 CA 27 00 67 02 00 00  B0 CC 27 00 18 60 00 00  HÊ' g   °Ì'  `
44CB6020  00 00 00 00 15 4E 44 D6  64 65 78 0A 30 33 35 00       NDÖdex 035
44CB6030  FE 36 F0 7A D2 B3 D0 1C  87 A9 16 06 48 B9 93 89  þ6ðzÒ³Ð ▌©  H¹▌▌
44CB6040  C8 5A D8 FF 2C E7 2E 2B  20 CA 27 00 70 00 00 00  ÈZØÿ,ç.+ Ê' p
44CB6050  78 56 34 12 00 00 00 00  00 00 00 00 50 C9 27 00  xV4          PÉ'
44CB6060  E8 29 00 00 70 00 00 00  5A 05 00 00 10 A8 00 00  è)  p   Z     ¨
44CB6070  F7 06 00 00 78 BD 00 00  2A 11 00 00 0C 11 01 00  ÷   x½  *
44CB6080  59 1C 00 00 5C 9A 01 00  60 03 00 00 24 7D 02 00  Y   \▌  `   $}
44CB6090  FC E0 24 00 24 E9 02 00  BE 3D 23 00 C0 3D 23 00  üà$ $é  ¾=# À=#
44CB60A0  C4 3D 23 00 CB 3D 23 00  CE 3D 23 00 D4 3D 23 00  Ä=# Ë=# Î=# Ô=#
44CB60B0  D7 3D 23 00 DB 3D 23 00  DF 3D 23 00 E4 3D 23 00  ×=# Û=# ß=# ä=#
```

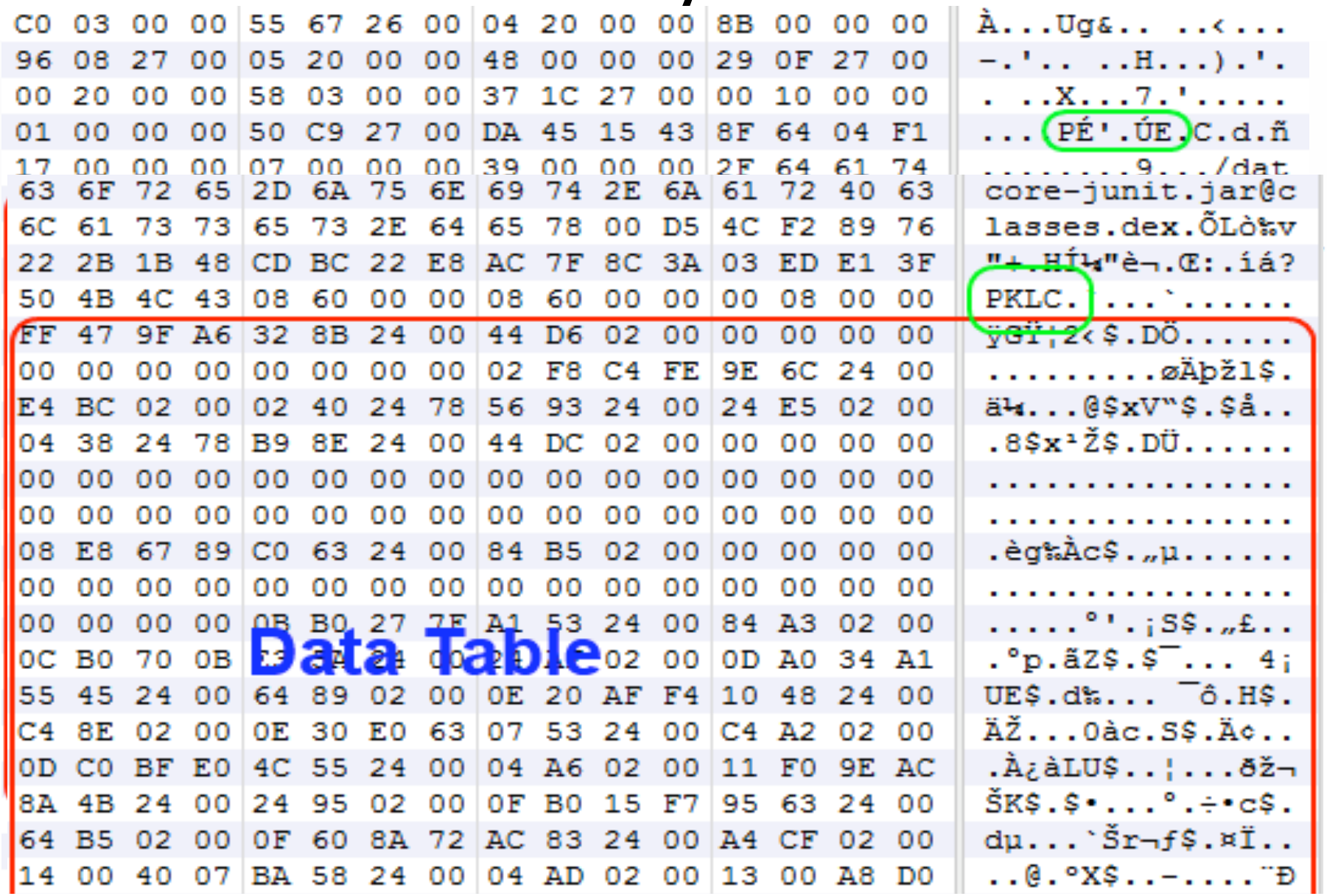
```
@classes.dex

rk.jar@classes.dex

.policy.jar@classes.dex

s.jar@classes.dex
```

Java!

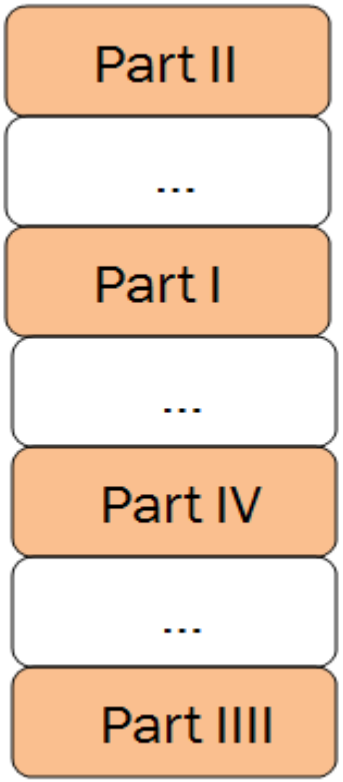

out_dex2jar.jar
android.annotation
cn.com.zhangxueyousb.livewallpaper.mingche
custom
pay
BuildConfig
LoadActivity
LoadingPopAd
MyAdView
QuitPopAd
R
SMHMCliveWallpaper
SMHMainActivity
SMHMainActivity
Content : Context
caipiao : Button
mHandler : Handler
onCreate(Bundle) : void
onDestroy() : void
onKeyDown(int, KeyEvent) : boolean
onPause() : void
onResume() : void
SMHSettingsActivity
com
alipay.android.app
android.internal.telephony
ax
elm
hiapk
isw
mobclick.android
payeco.android.plugin
tenpay.android.service
SMHMainActivity.class
LoadingPopAd.class
SMHSettingsActivity.class
SMHMCliveWallpaper.class
package cn.com.zhangxueyousb.livewallpaper.mingche;
import android.app.Activity;
public class SMHMainActivity extends Activity
{
private static final Context Content = null;
Button caipiao;
Handler mHandler = new Handler()
{
public void handleMessage(Message paramAnonymousMessage)
{
if (paramAnonymousMessage.what == 1)
SMHMainActivity.this.caipiao.setVisibility(0);
}
};
protected void onCreate(Bundle paramBundle)
{
super.onCreate(paramBundle);
requestWindowFeature(1);
setContentView(2130903040);
LMA.initSDK(this);
LMA.setAppkey(this, "11647");
LMA.showAD1(this);
AppConnect.getInstance("801457447258c745524f716c3d420183", "hiapk", this);
AppConnect.getInstance(this).setAdViewClassName(getPackageName() + ".MyAdView");
WinksApplication.onCreate(this);
AppConnect.getInstance(this).initPopAd(this);
if ((LoadActivity.isOther) || ((!LoadActivity.isOther) && (!LoadActivity.isTest(Content
{
this.mHandler.sendEmptyMessage(1);
AppConnect.getInstance(this).showPopAd(this);
new AdView(this, (LinearLayout)findViewById(2131361798)).DisplayAd(30);

Greater Wall

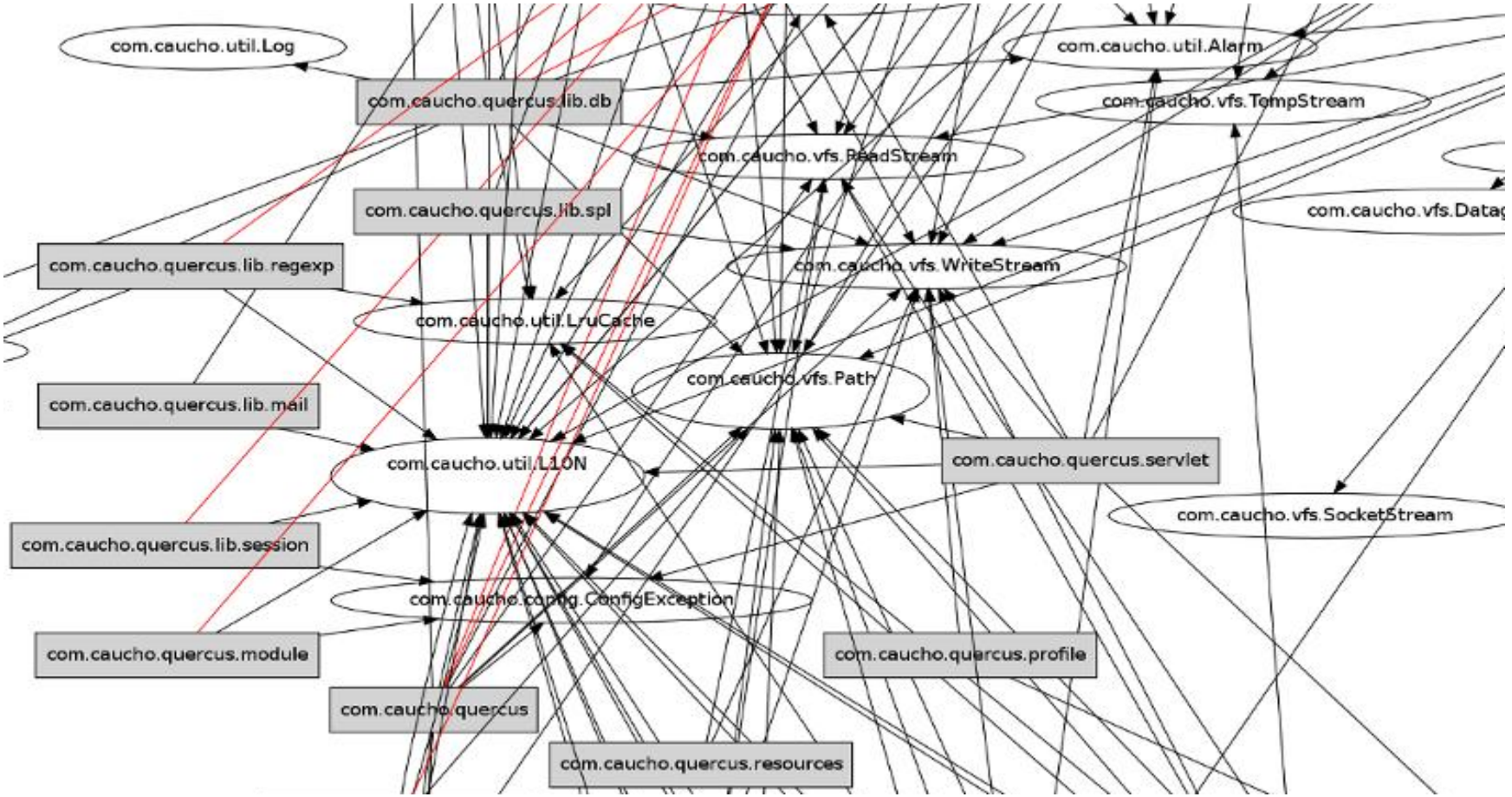
ODEX Memory obfuscation



non-continuous DEX



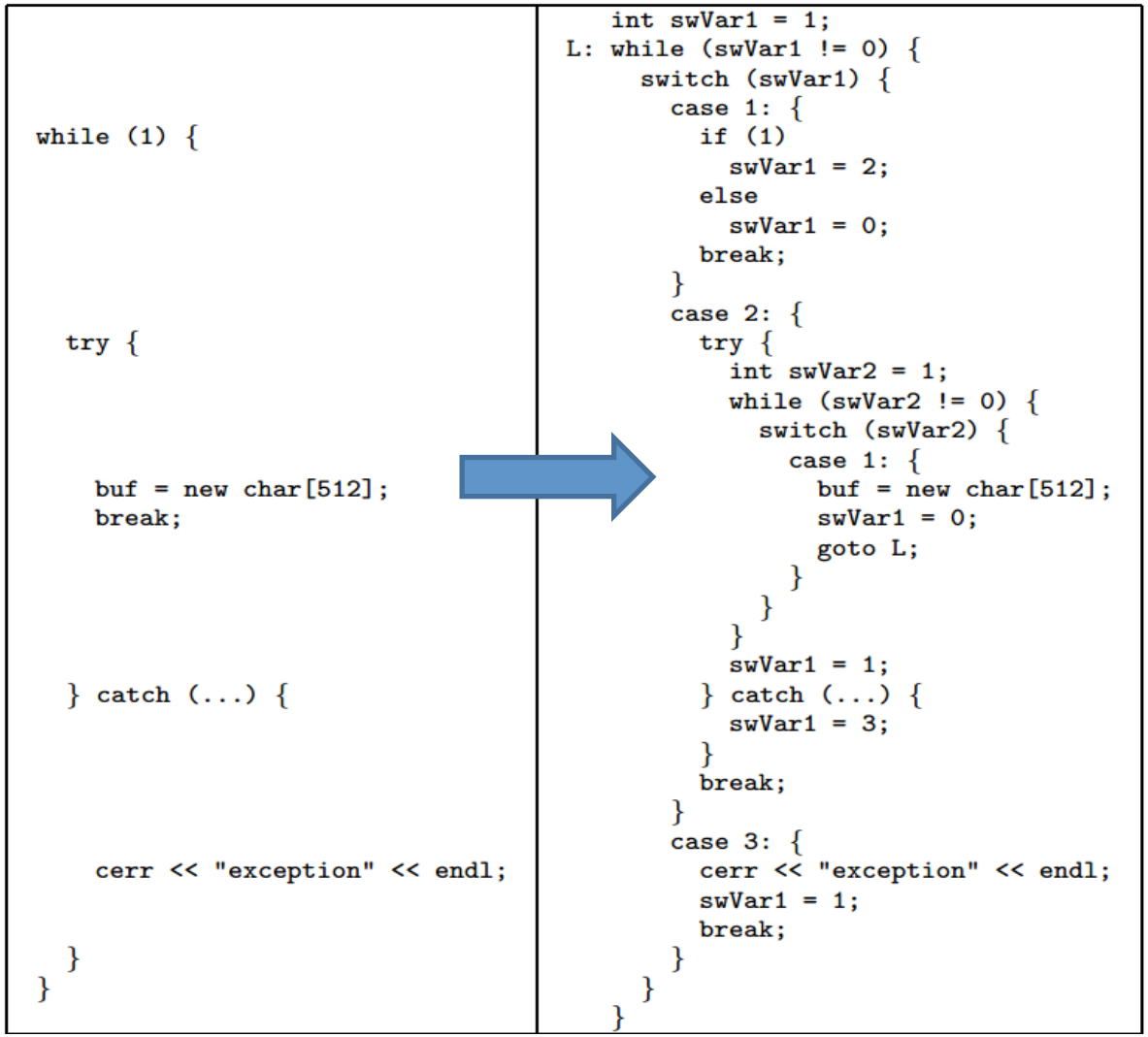
On-demand class loading



Code Obfuscation

```
#ifdef z7929401884
extern void za41dafc42e(const char*);
#define z1c52ffdd48(z22fc207d33,zde05b8b1b0) \
  do { if (z22fc207d33) za41dafc42e (#z22fc207d33); } while ((0x1a1+8313-0x221a))
#else
#include <cassert>
#define z1c52ffdd48(z22fc207d33,zde05b8b1b0) z7bd0031cc2 (!(z22fc207d33))
#endif
template<class zd9cfc9cefe,class z9cdf2cd536,class Allocator>basic_string<
zd9cfc9cefe,z9cdf2cd536,Allocator>&basic_string<zd9cfc9cefe,z9cdf2cd536,
Allocator>::replace(size_type z795f772c7c,size_type zddd43c876a,
const basic_string&str,size_type z8ad17de27a,size_type za2e5f06cde){
const size_t z51dea41a1e=str.length()+(0x12ac+3131-0x1ee5);if(z795f772c7c==
(0x455+8190-0x2453)&&zddd43c876a>=length()&&z8ad17de27a==(0xc15+4853-0x1f0a)&&
za2e5f06cde>=z51dea41a1e)return operator=(str);z1c52ffdd48(z8ad17de27a>
z51dea41a1e,"\x65\x72\x72\x6f\x72\x20\x69\x6e\x20\x72\x65\x70\x6c\x61\x63\x65");
#ifdef zd943335d79
++::z021c346d26.z1534cdbaf9;
```

Transformer I



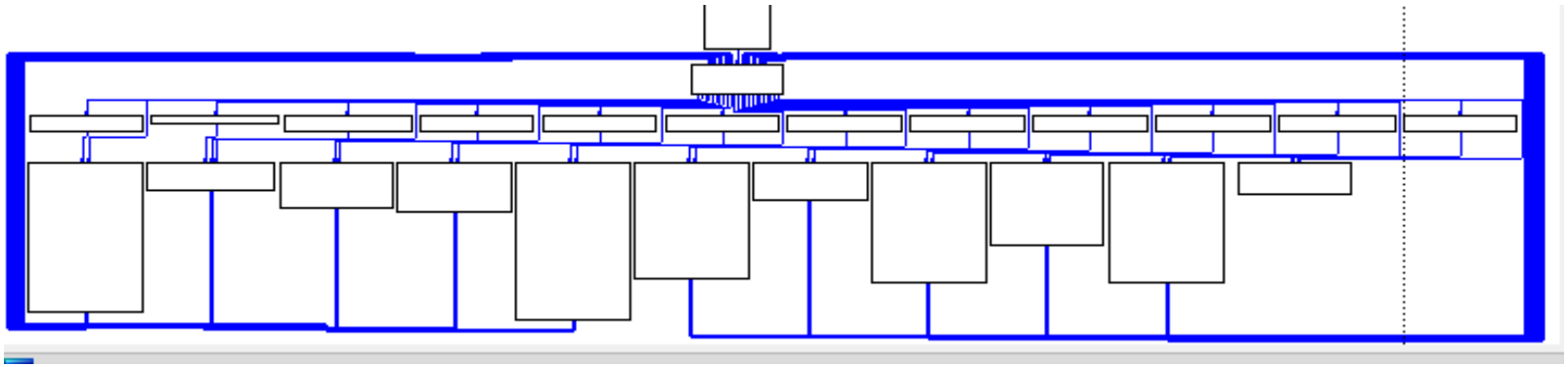
```
while (1) {

  try {

    buf = new char[512];
    break;

  } catch (...) {

    cerr << "exception" << endl;

  }
}
```

```
   int swVar1 = 1;
L: while (swVar1 != 0) {
     switch (swVar1) {
       case 1: {
         if (1)
           swVar1 = 2;
         else
           swVar1 = 0;
         break;
       }
       case 2: {
         try {
           int swVar2 = 1;
           while (swVar2 != 0) {
             switch (swVar2) {
               case 1: {
                 buf = new char[512];
                 swVar1 = 0;
                 goto L;
               }
             }
           }
           swVar1 = 1;
         } catch (...) {
           swVar1 = 3;
         }
         break;
       }
       case 3: {
         cerr << "exception" << endl;
         swVar1 = 1;
         break;
       }
     }
   }
```

Transformer II

```
loc_2DA88                               ; CODE XREF: sub_2D9B4+194↓j
                                        ; sub_2D9B4+1A4↓j ...
                LDR     R3, =0x6BF26745
                ADD     R3, R5, R3
                CMP     R3, #0xA        ; switch 11 cases
                ADDLS   PC, PC, R3,LSL#2 ; switch jump
                B       loc_2DB4C       ; jumptable 0002DA94 default case
; ---------------------------------------------------------------------------

loc_2DA9C                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DAC8       ; jumptable 0002DA94 case 0
; ---------------------------------------------------------------------------

loc_2DAA0                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DB4C       ; jumptable 0002DA94 default case
; ---------------------------------------------------------------------------

loc_2DAA4                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DB5C       ; jumptable 0002DA94 case 2
; ---------------------------------------------------------------------------

loc_2DAA8                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DB7C       ; jumptable 0002DA94 case 3
; ---------------------------------------------------------------------------

loc_2DAAC                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DBA0       ; jumptable 0002DA94 case 4
; ---------------------------------------------------------------------------

loc_2DAB0                               ; CODE XREF: sub_2D9B4+E0↑j
                B       loc_2DC2C       ; jumptable 0002DA94 case 5
; ---------------------------------------------------------------------------
```

Transformer III



Own dynamic linker

Packer

- UPX
- Unknown

Summary

- Memory
- Loader logic
- Limitations

References

- http://bluebox.com/
- http://code.google.com/p/volatility/
- http://jbremer.org/automated-deobfuscation-of-android-applications/
- http://www.inf.u-szeged.hu/~akiss/pub/pdf/laszlo_obfuscating.pdf

Thank you!