Writeup of Android02(android) in WhaleCTF

原创

C0ss4ck 于 2018-02-05 00:45:07 发布 497 收藏
分类专栏： Android_of_CTF 文章标签： CTF reverse Android java

本文链接：https://blog.csdn.net/cossack9989/article/details/79256097
版权

Android_of_CTF 专栏收录该内容
1 篇文章 0 订阅
订阅专栏

先甩链接Android02

老套路，直接把Android02扔进AndroidKiller，得到整个工程的反汇编代码。随后用jd-gui查看MainActivity的java代码——

```
package com.tencent.crasms;

import android.app.Activity;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity
  extends Activity
{
  Button a;

  static
  {
    System.loadLibrary("msyk");
  }

  protected void onCreate(Bundle paramBundle)
  {
    super.onCreate(paramBundle);
    setContentView(2130968602);
    this.a = ((Button)findViewById(2131427420));
    this.a.setOnClickListener(new View.OnClickListener()
    {
      public void onClick(View paramAnonymousView)
      {
        try
        {
          paramAnonymousView = SmsManager.getDefault();
          MainActivity.this.rx3sdfx(paramAnonymousView);
          label12:
          Toast.makeText(MainActivity.this.getApplicationContext(), "ok", 0).show();
          return;
        }
        catch (Exception paramAnonymousView)
        {
          break label12;
        }
      }
    });
  }

  public native void rx3sdfx(SmsManager paramSmsManager);
}
```

观察这个程序，发现要取得SmsManager的默认实例，显然这个默认实例应当出现在rx3sdfx函数内，然而此处该函数缺失反编译代码。陷入沉思……

又观察到loadLibrary("msyk")，于是找到libmysk.so文件，索性用IDA反汇编下好了；扔进IDA，反编译得rx3sdfx的C风格伪代码——

```
int __fastcall rx3sdfx(int a1, int a2, int a3)
{
  int v3; // r9@1
  int v4; // r8@1
  char *v5; // r3@2
  int v6; // r11@5
  int result; // r0@9
  int v8; // [sp+18h] [bp-4F8h]@1
  int v9; // [sp+1Ch] [bp-4F4h]@4
  int v10; // [sp+24h] [bp-4ECh]@1
  int v11; // [sp+28h] [bp-4E8h]@1
  int v12; // [sp+2Ch] [bp-4E4h]@1
  char v13; // [sp+30h] [bp-4E0h]@1
  char v14; // [sp+31h] [bp-4DFh]@1
  char v15; // [sp+32h] [bp-4DEh]@1
  int s; // [sp+64h] [bp-4ACh]@1
  int v17; // [sp+68h] [bp-4A8h]@1
  int v18; // [sp+6Ch] [bp-4A4h]@1
  int v19; // [sp+70h] [bp-4A0h]@1
  int v20; // [sp+74h] [bp-49Ch]@1
  int v21; // [sp+78h] [bp-498h]@1
  int v22; // [sp+7Ch] [bp-494h]@1
  char v23; // [sp+E4h] [bp-42Ch]@1
  char v24; // [sp+E5h] [bp-42Bh]@1
  char v25; // [sp+E6h] [bp-42Ah]@1
  char v26; // [sp+E7h] [bp-429h]@1
  char v27; // [sp+E8h] [bp-428h]@1
  char v28; // [sp+E9h] [bp-427h]@1
  char v29; // [sp+EAh] [bp-426h]@1
  char v30; // [sp+EBh] [bp-425h]@1
  char v31; // [sp+ECh] [bp-424h]@1
  char v32; // [sp+EDh] [bp-423h]@1
  char v33; // [sp+EEh] [bp-422h]@1
  char v34; // [sp+EFh] [bp-421h]@1
  char v35; // [sp+F0h] [bp-420h]@1
  char v36; // [sp+F1h] [bp-41Fh]@1
  char v37; // [sp+F2h] [bp-41Eh]@1
  char v38; // [sp+F3h] [bp-41Dh]@1
  char v39; // [sp+F4h] [bp-41Ch]@1
  char v40; // [sp+F5h] [bp-41Bh]@1
  char v41; // [sp+F6h] [bp-41Ah]@1
  char v42; // [sp+F7h] [bp-419h]@1
  char v43; // [sp+F8h] [bp-418h]@1
  char v44; // [sp+F9h] [bp-417h]@1
  char v45; // [sp+FAh] [bp-416h]@1
  char v46; // [sp+FBh] [bp-415h]@1
  char v47; // [sp+FCh] [bp-414h]@1
  char v48; // [sp+FDh] [bp-413h]@1
  char v49; // [sp+FEh] [bp-412h]@1
  char v50; // [sp+FFh] [bp-411h]@1
  char v51; // [sp+100h] [bp-410h]@1
  char v52; // [sp+101h] [bp-40Fh]@1
  char v53; // [sp+102h] [bp-40Eh]@1
  char v54; // [sp+103h] [bp-40Dh]@1
  char v55; // [sp+104h] [bp-40Ch]@1
  char v56; // [sp+105h] [bp-40Bh]@1
  char v57; // [sp+106h] [bp-40Ah]@1
  char v58; // [sp+107h] [bp-409h]@1
  char v59; // [sp+108h] [bp-408h]@1
  char v60; // [sp+109h] [bp-407h]@1
  char v61; // [sp+10Ah] [bp-406h]@1
  char v62; // [sp+10Bh] [bp-405h]@1
  char v63; // [sp+10Ch] [bp-404h]@1
```

```
    char v63; // [sp+10Ch] [bp-404h]@1
    char v64; // [sp+10Dh] [bp-403h]@1
    char v65; // [sp+10Eh] [bp-402h]@1
    char v66; // [sp+10Fh] [bp-401h]@1
    char v67; // [sp+110h] [bp-400h]@1
    char v68; // [sp+111h] [bp-3FFh]@1
    char v69; // [sp+112h] [bp-3FEh]@1
    char v70; // [sp+113h] [bp-3FDh]@1
    char v71; // [sp+114h] [bp-3FCh]@1
    char v72; // [sp+115h] [bp-3FBh]@1
    char v73; // [sp+116h] [bp-3FAh]@1
    char v74; // [sp+117h] [bp-3F9h]@1
    char v75; // [sp+118h] [bp-3F8h]@1
    char v76; // [sp+119h] [bp-3F7h]@1
    char v77; // [sp+11Ah] [bp-3F6h]@1
    char v78; // [sp+11Bh] [bp-3F5h]@1
    char v79; // [sp+11Ch] [bp-3F4h]@1
    char v80; // [sp+11Dh] [bp-3F3h]@1
    char v81; // [sp+11Eh] [bp-3F2h]@1
    char v82; // [sp+11Fh] [bp-3F1h]@1
    char v83; // [sp+120h] [bp-3F0h]@1
    char v84; // [sp+121h] [bp-3EFh]@1
    char v85; // [sp+122h] [bp-3EEh]@1
    char v86; // [sp+123h] [bp-3EDh]@1
    char v87; // [sp+124h] [bp-3ECh]@1
    char v88; // [sp+125h] [bp-3EBh]@1
    char v89; // [sp+126h] [bp-3EAh]@1
    char v90; // [sp+127h] [bp-3E9h]@1
    char v91; // [sp+128h] [bp-3E8h]@1
    char v92; // [sp+129h] [bp-3E7h]@1
    char v93; // [sp+12Ah] [bp-3E6h]@1
    char v94; // [sp+12Bh] [bp-3E5h]@1
    char v95; // [sp+12Ch] [bp-3E4h]@1
    char v96; // [sp+12Dh] [bp-3E3h]@1
    char v97; // [sp+12Eh] [bp-3E2h]@1
    char v98; // [sp+12Fh] [bp-3E1h]@1
    char v99; // [sp+130h] [bp-3E0h]@1
    char v100; // [sp+131h] [bp-3DFh]@1
    char v101; // [sp+132h] [bp-3DEh]@1
    char v102; // [sp+133h] [bp-3DDh]@1
    char v103; // [sp+134h] [bp-3DCh]@1
    char v104; // [sp+135h] [bp-3DBh]@1
    char v105; // [sp+136h] [bp-3DAh]@1
    char v106; // [sp+137h] [bp-3D9h]@1
    char v107; // [sp+138h] [bp-3D8h]@1
    char v108; // [sp+139h] [bp-3D7h]@1
    char v109; // [sp+13Ah] [bp-3D6h]@1
    char v110; // [sp+13Bh] [bp-3D5h]@1
    char v111; // [sp+13Ch] [bp-3D4h]@1
    char v112; // [sp+13Dh] [bp-3D3h]@1
    char v113; // [sp+13Eh] [bp-3D2h]@1
    char v114; // [sp+13Fh] [bp-3D1h]@1
    char v115; // [sp+140h] [bp-3D0h]@1
    char v116; // [sp+141h] [bp-3CFh]@1
    char v117; // [sp+142h] [bp-3CEh]@1
    char v118; // [sp+143h] [bp-3CDh]@1
    char v119; // [sp+144h] [bp-3CCh]@1
    char v120; // [sp+145h] [bp-3CBh]@1
    char v121; // [sp+146h] [bp-3CAh]@1
    char v122; // [sp+147h] [bp-3C9h]@1
```

```
  char v123; // [sp+148h] [bp-3C8h]@1
  char v124; // [sp+149h] [bp-3C7h]@1
  char v125; // [sp+14Ah] [bp-3C6h]@1
  char v126; // [sp+14Bh] [bp-3C5h]@1
  char v127; // [sp+14Ch] [bp-3C4h]@1
  char v128; // [sp+14Dh] [bp-3C3h]@1
  char v129; // [sp+14Eh] [bp-3C2h]@1
  char v130; // [sp+14Fh] [bp-3C1h]@1
  char v131; // [sp+150h] [bp-3C0h]@1
  char v132; // [sp+151h] [bp-3BFh]@1
  char v133; // [sp+152h] [bp-3BEh]@1
  int v134; // [sp+1E4h] [bp-32Ch]@1
  int v135; // [sp+1E8h] [bp-328h]@1
  char v136; // [sp+1ECh] [bp-324h]@1
  char v137; // [sp+1EDh] [bp-323h]@1
  char v138; // [sp+1EEh] [bp-322h]@1
  int v139; // [sp+2E4h] [bp-22Ch]@1
  int v140; // [sp+2E8h] [bp-228h]@1
  int v141; // [sp+2ECh] [bp-224h]@1
  int v142; // [sp+2F0h] [bp-220h]@1
  int v143; // [sp+2F4h] [bp-21Ch]@1
  int v144; // [sp+2F8h] [bp-218h]@1
  int v145; // [sp+2FCh] [bp-214h]@1
  int v146; // [sp+300h] [bp-210h]@1
  int v147; // [sp+304h] [bp-20Ch]@1
  char v148; // [sp+308h] [bp-208h]@1
  char v149; // [sp+309h] [bp-207h]@1
  int v150; // [sp+4E4h] [bp-2Ch]@1

  v3 = a1;
  v8 = a3;
  v150 = _stack_chk_guard;
  j_j_memset(&s, 0, 0x80u);
  s = 0x190F050A;
  v17 = 0x440F0204;
  v18 = 0xE070E1F;
  v19 = 0x504031B;
  LOBYTE(v20) = 18;
  BYTE3(v20) = 6;
  v21 = 0x50A2618;
  v22 = 0x190E0C0A;
  *(&v20 + 1) = '8D';
  j_j_memset(&v10, 0, 0x40u);
  v10 = 0xF050E18;
  v11 = 0x1F130E3F;
  v12 = 0x18180E26;
  v13 = 0xA;
  v14 = 12;
  v15 = 14;
  j_j_memset(&v23, 0, 0x100u);
  v23 = 67;
  v26 = 0xA;
  v27 = 29;
  v28 = 0xA;
  v31 = 0xA;
  v32 = 5;
  v33 = 12;
  v35 = 56;
  v36 = 31;
  v37 = 25;
```

```
    v37 = 25;
    v38 = 2;
    v39 = 5;
    v40 = 12;
    v41 = 80;
    v44 = 0xA;
    v45 = 29;
    v46 = 0xA;
    v49 = 0xA;
    v50 = 5;
    v51 = 12;
    v53 = 56;
    v54 = 31;
    v55 = 25;
    v24 = 39;
    v25 = 1;
    v29 = 'D';
    v30 = 7;
    v34 = 'D';
    v42 = 39;
    v43 = 1;
    v47 = 'D';
    v48 = 7;
    v52 = 'D';
    v56 = 2;
    v57 = 5;
    v58 = 12;
    v59 = 80;
    v60 = 39;
    v61 = 1;
    v62 = 0xA;
    v63 = 29;
    v64 = 0xA;
    v65 = 'D';
    v66 = 7;
    v67 = 0xA;
    v68 = 5;
    v69 = 12;
    v70 = 'D';
    v71 = 56;
    v72 = 31;
    v73 = 25;
    v74 = 2;
    v75 = 5;
    v76 = 12;
    v77 = 80;
    v78 = 39;
    v79 = 0xA;
    v80 = 5;
    v81 = 15;
    v82 = 25;
    v83 = 4;
    v84 = 2;
    v85 = 15;
    v86 = 'D';
    v87 = 0xA;
    v88 = 27;
    v89 = 27;
    v90 = 'D';
    v91 = 59;
```

```
v92 = 14;
v93 = 5;
v94 = 15;
v95 = 2;
v96 = 5;
v97 = 12;
v98 = 34;
v99 = 5;
v100 = 31;
v101 = 14;
v102 = 5;
v103 = 31;
v104 = 80;
v105 = 39;
v106 = 0xA;
v107 = 5;
v108 = 15;
v109 = 25;
v110 = 4;
v111 = 2;
v112 = 15;
v114 = 0xA;
v115 = 27;
v116 = 27;
v117 = 'D';
v118 = 59;
v119 = 14;
v120 = 5;
v121 = 15;
v122 = 2;
v123 = 5;
v113 = 'D';
v124 = 12;
v125 = 34;
v126 = 5;
v127 = 31;
v128 = 14;
v129 = 5;
v130 = 31;
v131 = 80;
v132 = 66;
v133 = 61;
j_j_memset(&v139, 0, 0x200u);
v139 = 0x1802033F;
v140 = 0x4033B4B;
v141 = 0x34B0E05;
v142 = 0x94B180A;
v143 = 0x4B050E0E;
LOWORD(v144) = 0xA03;
BYTE2(v144) = 8;
v145 = 0x4B470F0E;
v146 = 0x53512538;
v147 = 0x595A5258;
v148 = 0x5A;
v149 = 0x5E;
j_j_memset(&v134, 0, 0x100u);
v136 = 0x5F;
v134 = 0x5B52585A;
v135 = 0x5C535B5A;
```

```
v137 = 0x5A;
v138 = 0x53;
s ^= 0x6B6B6B6Bu;
v17 ^= 0x6B6B6B6Bu;
v18 ^= 0x6B6B6B6Bu;
v19 ^= 0x6B6B6B6Bu;
v20 ^= 0x6B6B6B6Bu;
v21 ^= 0x6B6B6B6Bu;
v22 ^= 0x6B6B6B6Bu;
v4 = (*(*v3 + 24))(v3, &s);
j_j_memset(&s, 0, 0x80u);
if ( v4 )
{
  v10 ^= 0x6B6B6B6Bu;
  v11 ^= 0x6B6B6B6Bu;
  v12 ^= 0x6B6B6B6Bu;
  v13 ^= 0x6Bu;
  v14 ^= 0x6Bu;
  v15 ^= 0x6Bu;
  v5 = &v23;
  do
  {
    *v5 ^= 0x6B6B6B6Bu;
    v5 += 4;
  }
  while ( v5 != &v131 );
  v131 ^= 0x6Bu;
  v132 ^= 0x6Bu;
  v133 ^= 0x6Bu;
  v9 = (*(*v3 + 132))(v3, v4, &v10, &v23);
  j_j_memset(&v10, 0, 0x40u);
  j_j_memset(&v23, 0, 0x100u);
  if ( v9 )
  {
    v134 ^= 0x6B6B6B6Bu;
    v135 ^= 0x6B6B6B6Bu;
    v136 ^= 0x6Bu;
    v137 ^= 0x6Bu;
    v138 ^= 0x6Bu;
    v6 = (*(*v3 + 668))(v3, &v134);
    j_j_memset(&v134, 0, 0x100u);
    v139 ^= 0x6B6B6B6Bu;
    v140 ^= 0x6B6B6B6Bu;
    v141 ^= 0x6B6B6B6Bu;
    v142 ^= 0x6B6B6B6Bu;
    v143 ^= 0x6B6B6B6Bu;
    v144 ^= 0x6B6B6B6Bu;
    v145 ^= 0x6B6B6B6Bu;
    v146 ^= 0x6B6B6B6Bu;
    v147 ^= 0x6B6B6B6Bu;
    v148 ^= 0x6Bu;
    v149 ^= 0x6Bu;
    if ( (*(*v3 + 668))(v3, &v139) )
      _JNIEnv::CallVoidMethod(v3, v8, v9, v6);
    j_j_memset(&v139, 0, 0x200u);
  }
  (*(*v3 + 92))(v3, v4);
}
result = (*(*v3 + 912))(v3);
if ( result )
```

```
    result = (*(*v3 + 68))(v3);
  if ( v150 != _stack_chk_guard )
    j_j___stack_chk_fail(result);
  return result;
}
```

连虚拟机都懒得开……就直接手算吧……

得到最终内存中应该出现的数据：

- v139~v149 : This Phone has been hacked, SN:8391215
- v134~v138 : 13901087418
- v10~v15 : sendTextMessage
- s~v22 : android/telephone

于是得到题目要求的恶意短信内容与SN；提交，pass~