对某APK的一次分析(转from看雪)

转载

starflier 于 2014-03-18 14:53:32 发布 721 收藏
分类专栏： android编程


android编程 专栏收录该内容
13 篇文章 0 订阅
订阅专栏

一直是在Windows上学习软件安全的，这几天在使用一个安卓上的软件，带有注册功能，突然心血来潮，就想拿来分析一下，其实我Java都没学过，也是从来没有在安卓上写过软件，实现一个对话框都要百度，查N久的SDK文档。再加上分析的第一个软件就是经过混淆处理的，于是乎自然而然的就会遇到很多困难，折腾了几天原来的激情也没有了，就此作罢，但是因为查资料花了很大力气，所以但是还是这几天得到的一些东西记下来吧，让其他的人能够不用再浪费那么多时间。

一， 工具 分析当然少不了得力的工具，像在Windows上，要是少了OD和IDA等众多出色的工具，那就举步维艰了，这里列出几个我试过的分析安卓Apk的工具:

1, apktool,smali,baksmali:

https://code.google.com/p/android-apktool/

apktool是基于smali的项目，提供APK代码，资源反编译+编译一条龙服务，还是很方便的，不过看到有人说这是土八路用的玩意儿，。反编译来的代码是smali文件，详细的语法请参考https://code.google.com/p/smali/.下面是效果图:

2, dextojar + jd-gui:

http://code.google.com/p/dex2jar/

http://java.decompiler.free.fr/?q=jdgui

首先用把编译好的dex文件转为class文件，然后用jd-gui可以转为java源文件，不过有时错误很多，看到有人说是因为dex2jar不能正确处理循环导致的，效果如下:

3 ,dextojar + DJ 这个跟上面那个差不多，基于dextojar,所以也会继承上面的错误,不过JD反编译处理的代码带很多goto和标号，值得一提的是，DJ虽然是个收费软件，但是其目录下的JAD文件实现了它的全部功能，可以直接命令行操作，效果如下:

百度基本上搜出的都是上面那几个了，下面还有些google的结果。

4,ddx: 网址忘记存了，大家google一下吧，这个是个和smali独立的项目,语法和smali类似,都是虚拟字节码,效果如下:

5,ded

http://siis.cse.psu.edu/ded/index.html#banner

这个是我比较推荐的工具,也是提供直接从apk到java源代码的一条龙服务，而且可读性非常好，那时相当的舒服,不过有些方法无法反编译，这就要借助上面其他的工具了，如果如下:

6,AndroChef Java Decompiler

这个忘记说了，也是很好用的。

好，工具介绍完了，下面来实战一下。

一.5 调试工具

增加一节，有点乱七八糟，但是想到什么就马上去实现，这也是个优点，不是么？呵呵。世界就是那么神奇。

调试的话推荐NetBean + DDMS,没什么好说的，apktool网站上摆着一个smali调试，但是发布的版本和视频教程里面不一样，不懂他是何居心。难道是为了apk的安全着想?。所以没有测试成功，还有 古河 兄说的AndBug也试了一下，没有脚本而单独使用工具自身提供的几个命令的话，那个还是纠结的，又不能单步，在方法上下断还不一定成功，参数不合规范立马罢工，这个还是比较顽皮的。

二， 实战

其实这也没有什么好说的，都直接反编译成源代码了，爱干啥干啥，虽然我没学过Java，但是这一点也不能妨碍我理解它呀，都是高级语言，长得差不多。。

这里跟Windows不同的是，程序设计的时候是面向对象的，反编译以后呈现的结果也还是面向对象的。

程序可以选择一个Key文件进行导入，找到它的onitemclick()事件:

```
public final void onItemClick(AdapterView var1, View var2, int var3,
      long var4)
  {
    w var6 = (w) this.b.b.elementAt(var3);
    //如果点击的是上一层目录
    if (var6.b.equals(".."))
    {
      this.c((new File(this.b.a)).getParent());
      this.e.setText(this.b.a);
      this.e.requestLayout();
      this.m.notifyDataSetChanged();
    }
    //如果点击的是目录
    else if (var6.a)
    {
      this.c((new File(this.b.a, var6.b)).getPath());
      this.e.setText(this.b.a);
      this.e.requestLayout();
      this.m.notifyDataSetChanged();
    }
    else
    {
      this.d.dismiss();  //销毁对话框
      if (this.g != null)
      {
        File var7 = new File(this.b.a, var6.b);
        this.g.a(var7.getPath());  //调用a处理
      }
    }

  }
```

代码经过混淆处理，名字变成abcdefg没有意义了有些不好懂,但是借助IDE慢慢分析改过来就行了。选择目录以后调用this.g.a(string),这里前面有private ac g = null;可是ac是个接口而已,

```
public interface ac {

   void a(String var1);
}
```

，这就有点为难了，我的办法是查看所有实现了这个接口的类，很低级的哈,然后发现一个可疑的类，其中方法为：

```
public final void a(String filepath)
  {
    try
    {
      Context context = this.a.getContext();
      FileInputStream fileinput = new FileInputStream(filepath);
      byte[] buffer = new byte[1024];
      int nbufferread = fileinput.read(buffer);
      fileinput.close();
      if (nbufferread > 0)
      {
        ///license.dat
        //把key的内容写进/license.dat
        FileOutputStream fileoutput = new FileOutputStream(ai.getKeyFilepath(context, "",true));  //静态方法
        fileoutput.write(buffer, 0, nbufferread);
        fileoutput.flush();
        fileoutput.close();
      }

      d.a(this.a.getContext(), GkMainView.a(this.a).c());
    }
    catch (Exception var8)
    {
      d.a(this.a.getContext(), var8);
    }

  }
```

现在我们再看d.a(Context context, boolean ifnull);

```
public static void a(Context  context, boolean  ifnull) throws Throwable
   {
```

```
        Object n0;
        byte traChinese, sinChinese;
        boolean bRegist;
        char firstbyte;
        String szWarning, r7, r8, szifregist, r11, r18, szIMEI, szMISI, _szIMEI, _szMISI, r27, r29, r59, $r73;
        WebView webview;
        AlertDialog.Builder alerdialogbuilder;
        n0 = null;
        traChinese = (byte) (byte) 1;
        bRegist = (new ai()).a(context);      //主要就是这个返回了
        firstbyte = context.getString(2131099663).charAt(0);

        if (firstbyte != '\u5173')  //简体中文
        {
            sinChinese = (byte) (byte) 0;
        }
        else
        {
            sinChinese = (byte) (byte) 1;
        }

        if (firstbyte != '\u95dc')  //繁体中文
        {
            traChinese = (byte) (byte) 0;
        }

        r7 = "";

        if (sinChinese == (byte) 0)  //不是简体中文
        {
}
.........
}
```

下面是根据是否注册及本机语言呈现不同的用户界面而已，现在让我们把精力放到主要的那个方法那里，a(Context context),

```
public final boolean a(Context context) throws Throwable
  {
    boolean bresult = true;
    int var3 = 0;

    while (true)  //while( i < 2)
    {
      if (var3 >= 2)
      {
        bresult = false;
        break;
      }

      String szIMEI = cn.maocai.components.am.GetIMSI_(context);
      //IEMI码不为空的话
      if (szIMEI != null && szIMEI.length() > 0)
      {
        boolean bresult_;
        if (var3 == 0)
        {
          bresult_ = bresult;
        }
        else
        {
          bresult_ = false;
        }

        //第一遍的时候寻找/data/data/.../license.dat
        //第二遍为SD卡
        if (this.a(getKeyFilepath(context, szIMEI, bresult_), szIMEI))
        {
          break;
        }
```

```
      }

      String szIMSI = cn.maocai.components.am.getIMSI(context);
      if (szIMSI != null && szIMSI.length() > 0)
      {
        boolean var6;
        if (var3 == 0)
        {
          var6 = bresult;
        }
        else
        {
          var6 = false;
        }
                                                                    //这里
        if (this.a(getKeyFilepath(context, szIMSI, var6), szIMSI))
        {
          break;
        }
      }

      ++var3;
    }

    return bresult;
  }
```

现在再看:

```
private boolean a(String keyfilepath, String szdevicesID) throws java.lang.Throwable
  {

    boolean bResult, $z4;
    java.io.FileInputStream fileinput1, keyfileinput;
    int navailable, i5;
    byte[] bytekeybuffer, decryptKeybuf;
    X509EncodedKeySpec x509key;
    PublicKey publickey;
    Cipher cipher;
    String sttraKey, r29;
    Throwable r38;
    bResult = false;

    //如果在黑名单中就一切免谈了
    label_6:
    if (ai.checkblacklist(szdevicesID) == false)
    {
      fileinput1 = null;

      label_5:
      {
        label_4:
        {
          label_3:
          {
            label_2:
            {
              try
              {
                //处理一下文件名
                if ((new File(keyfilepath)).exists() == false)
                {
                  if (keyfilepath.endsWith(".dat") != false)
                  {
                    break label_6;
                  }
                  else
                  {
                    keyfilepath = (new StringBuilder())
                        .append(keyfilepath)
```

```
                        .append(".dat")
                        .toString();
                    }
                  }

                  keyfileinput = new FileInputStream(keyfilepath);
                }
                catch (Exception $r33)
                {
                  break label_2;
                }
                catch (Throwable $r37)
                {
                  r38 = $r37;
                  keyfileinput = fileinput1;
                  break label_3;
                }

                try
                {
                  navailable = keyfileinput.available();
                  //可读字节不能为0
                  if (navailable <= 0)
                  {
                    break label_5;
                  }
                  else
                  {
                    bytekeybuffer = new byte[navailable];

                    keyfileinput.read(bytekeybuffer);
                    keyfileinput.close();
                    //得到公钥
                    x509key = new X509EncodedKeySpec(statsbytes);
                    publickey = KeyFactory.getInstance("RSA").generatePublic(x509key);
                    cipher = Cipher.getInstance("RSA");
                    //解密模式
                    cipher.init(javax.crypto.Cipher.DECRYPT_MODE ,publickey);

                    //解密字串
                    decryptKeybuf = cipher.doFinal(bytekeybuffer);
                    //解密密后的密码转成字符串
                    sttraKey = new String(
                        decryptKeybuf,
                        0,
                        decryptKeybuf.length,
                        "UTF-8");
                    i5 = sttraKey.indexOf("GameKiller\nIMEI\n");

                    if (i5 >= 0)
                    {
                      a = sttraKey.substring(i5);
                    }

                    r29 = (new StringBuilder(
                        "GameKiller\nIMEI\n"))
                        .append(szdevicesID)
                        .append("\n")
                        .toString();
                    $z4 = a.startsWith(r29);
                    break label_4;
                  }
.....
返回 $z4;
}
//看到这里是不是感觉被忽悠了哈，嘿嘿，他用私钥加密注册信息生成key然后程序用公钥解密后对比,所以写注册机还是算了吧。
```

三，安卓平台上的软件安全

感觉革命尚未成功，同志仍须努力,这个问题需要解决的东西应该很多吧，包括google该做的事情。开发者请参考ZhWeir兄的帖子

【原创】APK反破解之一：Android Java混淆(ProGuard)http://bbs.pediy.com/showthread.php?t=137112

【原创】APK反破解之二：Android APK 签名比对

【转帖】APK反破解之三：NDK编译.so动态库