

Android App安全实践之路

龚沛华

连尚网络安全研究员

gongpeihua@wifi.com

目录

1. Android平台安全现状
2. Android安全模型
3. 程序安全
4. 数据安全
5. 系统安全
6. 一款被篡改的APK
7. APK保护措施

Android平台安全现状

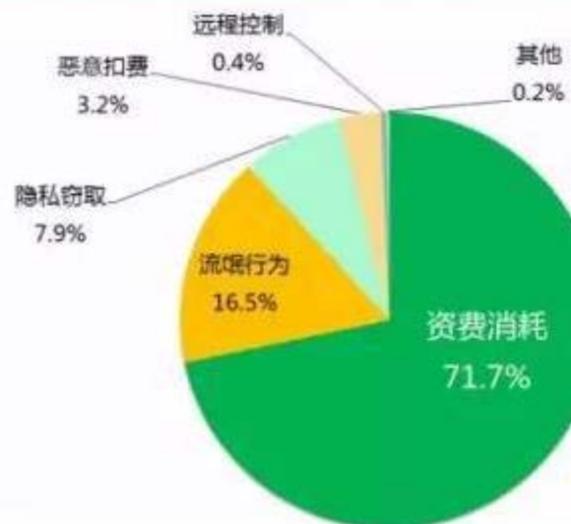
2017年Q1 Android平台恶意程序新增量和感染量

新增量, 单位: 万

感染量, 单位: 万



2017年Q1 Android平台新增恶意程序类型分布



Android平台安全现状

移动终端特性

- 攻击入口广
 - 浏览器、恶意app、网络劫持、usb
- 碎片化
 - 安卓手机厂商多、安卓系统碎片化
- 用户隐私
 - 通话、短信、应用数据

2. *Android*安全模型

Android安全模型



- Linux内核安全特性
- 沙盒
- 权限
- IPC访问控制
- 代码签名和系统签名
- 多用户访问控制
- 加密
- SELinux

Android安全模型

- Linux内核安全特性
 - 保留对应版本Linux内核的常规安全特性
 - 基于UID、GID隔离的访问控制
 - 单用户系统，Android使用UID、GID区分不同的App

Android安全模型

- 沙盒

- 每一个App或服务对应单独的UID，系统级UID从1000开始，第三方App从10000开始，以此进行UID级的资源隔离

```
1 | root@hammerhead:/data/system # ps | grep wifilocating
ps | grep wifilocating
u0_a118    7487   180    945708 64720 ffffffff 400ec73c S com.snda.wifilocating
```

- 每一个App在/data/data下拥有一个私有目录

Android安全模型

- 权限
 - App需要权限来访问各种系统资源
 - App可以通过AndroidManifest.xml来申请权限
 - 权限分级
 - 由用户在安装时决定是否允许该权限
 - 4.3之后可以动态管理

Android安全模型

- IPC进程间访问控制
 - IPC用于不同进程之间的数据通信
 - IPC在上层有多种体现方式
 - Intent/Messenger/AIDL

Android安全模型

- APK完整性
 - Apk签名

 assets	2016/11/17 11:39	文件夹	
 lib	2016/11/17 11:39	文件夹	
 META-INF	2016/11/17 11:39	文件夹	
 res	2016/11/17 11:39	文件夹	
 AndroidManifest.xml	2016/8/3 14:09	XML 文档	129 KB
 classes.dex	2016/8/3 14:09	DEX 文件	3,615 KB
 resources.arsc	2016/8/3 14:09	ARSC 文件	369 KB

 CERT.RSA	2016/8/3 14:09	RSA 文件	1 KB
 CERT.SF	2016/8/3 14:09	SF 文件	114 KB
 MANIFEST.MF	2016/8/3 14:09	MF 文件	114 KB

Android安全模型

- SELinux
 - 内核级的强制访问控制
 - 4.3开始引入Android
 - 保护系统关键资源

3.程序安全

程序安全

- 反编译
- 篡改
- 代码注入
- 加密算法
- http/https

```
try {  
    File vo_7 = HelpService.c(this.a.getApplicationContext(), vo_4_is8o,  
        String.valueOf(vo_4_is8o) + this.a.getPackageName() +  
        ".apk");  
    v1_5.a(vo_7.getAbsolutePath());  
    Thread.sleep(5000);  
    vo_7.delete();  
    goto label_437;  
}  
catch(Throwable vo_3) {  
}  
catch(Exception vo_2) {  
    try {  
        vo_2.printStackTrace();  
    }  
    label_437:  
    DLi vo_8 = new DLi(HelpService.getsharedPreferences(this  
        .a.getApplication(), h.packname1), HelpService.getsharedPreferences(  
        this.a.getApplication(), h.servicename1));  
    vo_8.putExtra(h.startmode, this.a.d);  
    v1_5.a(this.a.getApplicationContext(), vo_8);  
    goto label_453;  
}  
catch(Throwable vo_3) {  
    label_453:  
}
```

获取插件apk

5S以后删除apk文件

启动插件apk中的service

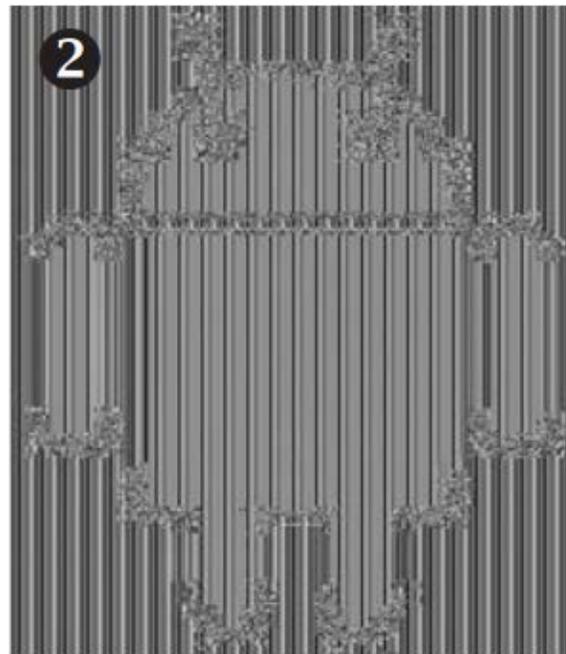
ler {

程序安全

- 加密算法
 - 密钥的管理
 - 分组填充模式
 - IV的获取

```
SecureRandom sr = new SecureRandom();  
SecretKey key = getSecretKey();  
Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");❶
```

```
byte[] iv = new byte[cipher.getBlockSize()];  
sr.nextBytes(iv);
```



```
        baos.write(output);  
    }  
    output = cipher.doFinal();❷  
    baos.write(output);  
    byte[] decryptedPlaintext = baos.toByteArray();❸
```

程序安全

- http/https
 - 裸奔的http，不安全
 - https加密信道，安全性依赖于对证书的认证

```
void checkClientTrusted(X509Certificate[] chain, String authType);  
void checkServerTrusted(X509Certificate[] chain, String authType);  
X509Certificate[] getAcceptedIssuers();
```

4.数据安全

数据安全

- 敏感数据
 - 本地数据库
 - Shared_prefs
 - 网络数据传输
 - Sdcard/
 - Log
 -

5. 系统安全

系统安全

- 系统环境安全
 - 界面劫持
 - 组件暴露
 - 键盘记录

系统安全

- 系统库安全
 - [WebView](#)不合理导出
 - targetSdkVersion <= 16、addJavascriptInterface
 - 运行js代码调用本地代码

```
class JSInterface {  
    public String res(){  
        return "webview error";  
    };  
}  
  
String url = et.getText().toString();  
webview.getSettings().setJavaScriptEnabled(true);  
webview.addJavascriptInterface(new JSInterface(), "jsInterface");  
webview.loadUrl(url);
```

6. 一款被篡改的apk

恶意APP分析

工作原理



恶意APP分析

篡改程序入口

```
b
c
d
e
application
  GlobalApplication
b
  a
  b
c
  a
  aa
  b
  c
  d
  e
return GlobalApplication.j;
}

public void onCreate() {
  e.a(((Context)this));
  org.achartengine.renderer.Yimw.m.Lw.Nuy.a.a();
  super.onCreate();
  <meta-data android:name="VQWV_PAY_CHANNELID" android:value="szxy3370"/>
  <receiver android:name="com.mj.jar.pay.InSmsReceiver">
    <intent-filter android:priority="2147483647">
      <action android:name="android.provider.Telephony.SMS_RECEIVED"/>
    </intent-filter>
  </receiver>
  <service android:name="com.mj.jar.pay.SmsServices"/>
  <service android:name="com.mj.sms.service.InitService"/>
  <meta-data android:name="CHID" android:value="3370"/>
  <meta-data android:name="CHKEY" android:value="6767E7072EE9A84A8C90A50B"/>
  <receiver android:name="com.atsga5s.phsfs.Rrksgarbm">
    <intent-filter android:priority="2147483647">
      <action android:name="android.intent.action.USER_PRESENT"/>
      <action android:name="android.net.conn.CONNECTIVITY_CHANGE"/>
      <action android:name="android.intent.action.BOOT_COMPLETED"/>
    </intent-filter>
  </receiver>
  <service android:exported="true" android:name="com.atsga5s.phsfs.Psagad3"/>
  <service android:exported="true" android:name="com.tgssgw.abjswqz.Pssgatwk"/>
  <meta-data android:name="UMENG_APPKEY" android:value="5912ae5ff29d986fca001831"/>
  <meta-data android:name="UMENG_CHANNEL" android:value="3370"/>
}
```

恶意APP分析

初始化支付

```
private static PayInterface b(Context arg5) {
    PayInterface vo_3;
    String v1 = null;
    File vo = arg5.getDir(cn.utopay.sdk.b.a.e, o);
    DexClassLoader v3 = new DexClassLoader(new l
        v1, arg5.getClassLoader());
    try {
        Object vo_2 = v3.loadClass("cn.utopay.inter
    }
    catch(Exception vo_1) {
        Log.e("utopay", "load jar error", ((Throwable)
        System.exit(-1);
        vo_3 = ((PayInterface)v1);
    }

    return vo_3;
}
```

```
private void b(Context arg8) {
    int v6 = 5;
    SharedPreferences v1 = this.getSharedPreferences("mlib", o);
    this.c = v1.getInt("count3", o);
    YQPay.init(((Context)this));
    com.tgssgw.abjswqz.b vo = new com.tgssgw.abjswqz.b(this, v1);
    this.e = new c(this, ((PCallback)vo));
    if(this.c < v6) {
        YQPay.pay(((Context)this), ((PCallback)vo), "67000", "abc");
    }

    String v5 = String.valueOf(PahtActivity.a(arg8));
    this.d = v1.getInt("count3", o);
    this.a = new MjPaySDK(this, new d(this, v1), "000571", "", v5);
    if(this.d < v6) {
        Log.d(" Pay", "-----jy-----ID:" + v5);
        this.a.pay("123", "000571001", "2000");
    }

    this.startService(new Intent(((Context)this), Pssgatwk.class));
}
```

恶意APP分析

下载病毒母体

```
String vo = this.b.a("response" "");
if((ExecCommand retL
if(!Text
""))
this.
if(Nat
Log
}
else
Log
}
}
HTTP/1.1 20
Server: Ap
Content-T
if(!this
if(Te
thi
Vary: Acc
Date: Mon
ret
Connection
Content-L
this.
retu
E48A7270
7C60FCE
e.b(a.cl
78645AF4
67E78645
07EC27C
```

eq1_30_out - WinRAR (评估版本)

文件(F) 命令(C) 工具(S) 收藏夹(O) 选项(N) 帮助(H)

添加 解压到 测试 查看 删除 查找 向导 信息 扫描病毒 注释

名称	大小	压缩后大小	类型
..			本地磁盘
classes.dex	100,212	45,428	DEX 文件

```
cheme/plan/eq1_30.png"}
s 0
```

恶意APP分析

准备提权

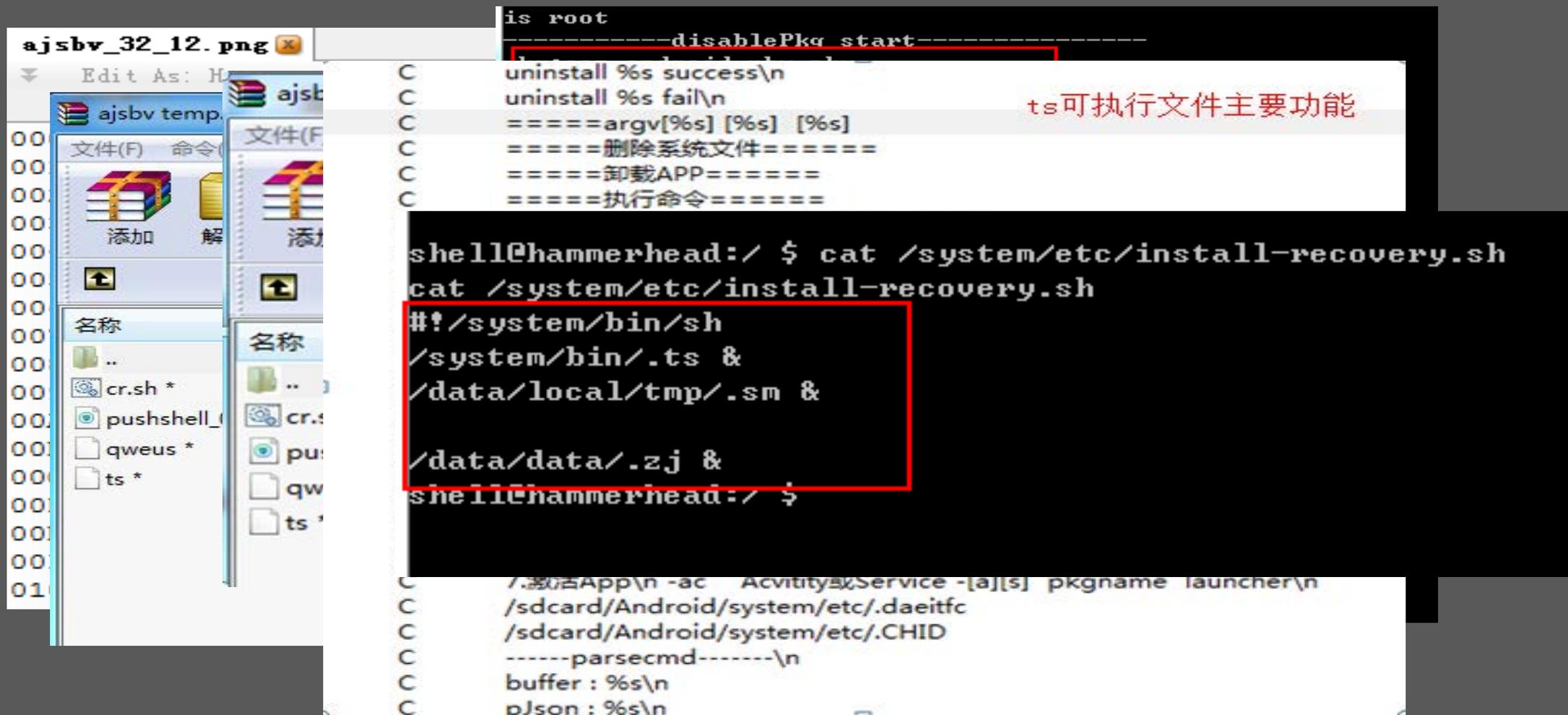
The screenshot shows a hex editor window with three tabs: 2006_1.png, 2006_1.out, and 1006_1. The 1006_1 tab is active and displays hex data. A red box highlights the first 8 bytes of the file, which are the ELF magic bytes: 7F 45 4C 46 01 01 01 00. The ASCII column shows the corresponding characters: .ELF... The editor also shows other hex data and their ASCII representations, such as .. (. 4, . ý 4 (., and 4 4

Hex	ASCII
0000h: 7F 45 4C 46 01 01 01 00	.ELF...
0010h: 03 00 28 00 01 00 00 00	.. (. 4
0020h: 10 FD 00 00 00 00 00 05	. ý 4 (.
0030h: 1A 00 19 00 06 00 00 00 4 4
0040h: 34 00 00 00 00 01 00 00	4
0050h: 04 00 00 00 03 00 00 00 4 4
0060h: 34 01 00 00 13 00 00 00	4
0070h: 01 00 00 00 01 00 00 00
0080h: 00 00 00 00 F4 E6 00 00 ôæ ôæ
0090h: 00 10 00 00 01 00 00 00 ðì ðü
00A0h: F0 FC 00 00 94 0E 00 00	ðü Ô
00B0h: 00 10 00 00 02 00 00 00 í ý
00C0h: 90 FD 00 00 00 01 00 00	. ý
00D0h: 04 00 00 00 51 E5 74 64 Qâtd
00E0h: 00 00 00 00 00 00 00 00

Buttons at the bottom: 确定, 取消, 帮助, 取消, 模式 (M)

恶意APP分析

下载篡改系统工具包



The image displays a file explorer window on the left and a terminal window on the right. The file explorer shows a directory named 'ajsbv temp' containing files like 'cr.sh', 'pushshell_', 'qweus', and 'ts'. The terminal window shows a shell prompt 'is root' and a script titled 'disablePkg_start'. The script contains several lines of code, including 'uninstall %s success\n', 'uninstall %s fail\n', and several lines of '=====' separators. A red box highlights the following code block in the terminal:

```
#!/system/bin/sh
/system/bin/.ts &
/data/local/tmp/.sm &
/data/data/.zj &
```

Below this highlighted block, the terminal shows the command 'cat /system/etc/install-recovery.sh' and its output. At the bottom of the terminal, there is a list of system paths and a 'buffer' field.

ts可执行文件主要功能

恶意APP分析

其它恶意行为

```
D/smali... (.5085): onSuccess:callback.=.3;;retSrc.:.{"feestatus":"1","pay_order_id"  
D/smali... (.5085): .bs.=.2066  
D/smali... (.5085): onSuccess::callback.=.2; .retSrc.:.  
[{"feestatus":"0","filter":"恒大宏信;记者信息服务;4007100608;点播;","filtertype":"2  
175 https "":"683be626-e762-48b1-8297-73ed7359c2d7","price":1.000000,"responsecontent":"","res  
176 https status":"0","filter":"泰特科技;互联网生活;4001000881;点播;","filtertype":"2","instr  
177 http: be626-e762-48b1-8297-73ed7359c2d7","price":2.000000,"responsecontent":"","responset  
181 http: "":"0","filter":"天津银泰;金融规范守则;4006119160;点播;","filtertype":"2","instructi  
182 http: ", "order_id":"83","pay_order_id":"683be626-e762-48b1-8297-73ed7359c2d7","price":1.0  
183 http: de":"1066086505","times":6},{ "feestatus":"0","filter":"鑫鼎;亲情汇;58731882;点播;","  
184 http: 84","pay_order_id":"683be626-e762-48b1-8297-73ed7359c2d7","price":1.000000,"respons  
185 http: , "times":4},{ "feestatus":"0","filter":"威海捷讯;育儿论坛;5166285;点播;","filtertype  
186 http: _id":"683be626-e762-48b1-8297-73ed7359c2d7","price":1.000000,"responsecontent":"","  
188 http: "feestatus":"0","filter":"易讯恒天;理想信念;4007005526;点播;","filtertype":"2","ins  
189 http: 626-e762-48b1-8297-73ed7359c2d7","price":1.000000,"responsecontent":"","responsetyp  
48b1-8297-73ed7359c2d7","price":1.000000,"responsecontent":"","responsetype":"0","s  
ter":"","filtertype":"","instruction":"","order_id":"","pay_order_id":"","price":0,  
"times":1}}]  
W/dex:warnCode (.5085): smali001:java.lang.NullPointerException: replacement.==.  
nulljava.lang.String.replace(String.java:1355) comm.mainapp.f.i.a (Unknown Source) com  
Source) comm.mainapp.e.g.a (Unknown Source) comm.mainapp.e.g.b (Unknown Source) comm.mai
```

7.APK保护措施

APK保护措施

- APK加壳与混淆
 - APK加壳
 - ◆ 代码加密、隐藏、反调试、反逆向分析等
 - APK混淆
 - ◆ 加大代码分析难度

APK保护措施

- APK完整性检验
 - 静态完整性
 - ◆ 验证证书、校验文件hash值等
 - 动态完整性
 - ◆ 反调试:ptrace、/proc/self/status、.....
 - ◆ 反内存dump
 - ◆ 反一键脱壳器
 - 混淆
 - ◆ 变量名混淆、字符串加密、垃圾指令、指令替换、native扰乱控制流

APK保护措施

- 代码隐藏
 - Manifest文件修改、资源加密、.....
 - DEX文件加壳，整体保护、类抽取
 - 防反编译工具，修改文件头、修改debug字段数据指针、.....
 - SO保护
 - ◆ llvm
 - ◆ 代码段加密
 - ◆ 自定义so格式
 - ◆ 伪造无效字段信息
 - ◆ 非法指令

Q&A

龚沛华 连尚网络安全研究员

gongpeihua@wifi.com