SandDroid User's Manual
SandDroid

SandDroid is an automatic Android application analysis system which combines static and dynamic analysis techniques. The home page shows as below.

Figure 1 SandDroid home page

Overview

Click the Overview link in the navigation bar and you will see the over-view information of the analyzed Android applications.
Search

You can look for the Android applications in our database by using the search functionality.

Figure 2 Search Form

How to search:

- File MD5: the MD5 value of the Android application;
- Signature: the SHA-1 value of the Android application’s signature;
- Package Name: the package name of the Android application;
- Malware Name: the malware family name of the Android application.

Apk Information Table

The searched results will show in the table which displays the brief information about the analyzed Android applications. Click the “detail icon” to see the detail analysis report.

Figure 3 Apk Information Table

Explanation of the apk Information table’s columns:
- Date: the date when starts to analyze the apk;
- MD5: the MD5 value of the apk;
- Package Name: the package name of the apk;
- Malware: the malware name of the apk;
- Risk: the risk level of the apk.

**IP Distribution**

The ip distribution map shows geographical distribution of ip addresses extracted from all the analyzed Android applications.

![IP Distribution Map](image)

**Top 20 Used Permissions**

The following chart shows top 20 used permissions in all the analyzed Android applications.
Top 20 Malware Families

The following chart shows top 20 malware families in all the analyzed Android applications.

Top 20 Advertisement Module Used

The following chart shows top 20 advertisement modules used in all the analyzed Android applications.
Upload Page

You can upload an apk or a zip file on the Upload page and the file size limits to 50MB.

Detail Report

As shown in Figure 3, you can click the detail icon to see the detail analysis report.

General Information

The icon on the right is the Android application's icon. You can download the pcap file and log file captured during dynamic analysis.
Figure 8 General Information

Risk Score

The risk score, between 0 and 100, represents the risk level. The higher the score is, the riskier the Android application is.

Risk Score

\[ \text{Score} = 43 \]

Figure 9 Risk Score

Risky Behaviors

As shown in Figure 10, Risky Behaviors table displays the suspicious embedded in this Android application.

Risky Behaviors

- Encrypt or Decrypt data
- Executes a Internet request
- Exist unused permissions
- Gets geographic location
- Gets the alphabetic name of current registered operator

Figure 10 Risky Behaviors

Malware Detected by VirusTotal

The malware detection results are based on the VirusTotal.

Malware Detected by VirusTotal

<table>
<thead>
<tr>
<th>Scanner</th>
<th>Threat Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>Android.Adware.Youmil.A</td>
</tr>
<tr>
<td>Ad-Aware</td>
<td>Adware/ANDR:Youmil.A.Gen</td>
</tr>
<tr>
<td>AntiVir</td>
<td></td>
</tr>
<tr>
<td>Baidu-International</td>
<td></td>
</tr>
</tbody>
</table>

Figure 11 Malware Detected by VirusTotal
Certificate

The certification content is extracted from the META-INF/*.RSA of the Android application.

Classification

Classify this Android application based on permissions using ID3, NaiveBayes, Decision Table, J48 algorithms respectively and calculate the probability that which category the Android application belongs to.
Code Features

As shown in Figure 14, Code Features table shows whether the Android application utilizes techniques including Native Code, Dynamic Loader, Java Reflection and Crypto.

<table>
<thead>
<tr>
<th>Code Feature</th>
<th>Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native Code</td>
<td></td>
</tr>
<tr>
<td>Dynamic Loader</td>
<td></td>
</tr>
<tr>
<td>Java Reflection</td>
<td></td>
</tr>
<tr>
<td>Crypto</td>
<td></td>
</tr>
</tbody>
</table>

Figure 14 Code Features

Permissions

Display the permissions declared in the AndroidManifest.xml.

<table>
<thead>
<tr>
<th>Permission Name</th>
<th>Protection Level</th>
<th>Threat Level</th>
<th>Customized</th>
<th>Duplicated</th>
<th>Used</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>android.permission.ACCESS_COARSE_LOCATION</td>
<td>dangerous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows an app to access approximate location derived from network location sources such as cell towers and Wi-Fi.</td>
</tr>
<tr>
<td>android.permission.ACCESS_FINE_LOCATION</td>
<td>dangerous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows applications to access fine location data. The accuracy of the location data is not specified.</td>
</tr>
<tr>
<td>android.permission.ACCESS_NETWORK_STATE</td>
<td>normal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Used for permissions that provide access to networking services. The app can then connect to the internet and other related network operations.</td>
</tr>
</tbody>
</table>

Figure 15 Permissions

Components

Display the components declared in the Android application.
Activities

<table>
<thead>
<tr>
<th>Name</th>
<th>Main Activity</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>com.kai.trivialpuzzle.FruitPuzzleActivity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>net.youmi.android.AdActivity</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Services

N/A

Broadcast Receivers

<table>
<thead>
<tr>
<th>Name</th>
<th>Dynamically Registered</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>net.youmi.android.em</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Content Providers

N/A

Figure 16 Components

Features

Display the features declared in the AndroidManifest.xml.

Features

- android.hardware.sensor.accelerometer
- android.hardware.touchscreen

Figure 17 Features

Libraries

Display the libraries declared in the AndroidManifest.xml.

Libraries

N/A

Figure 18 Libraries

Advertisement Modules

Display the advertisement modules used in this Android application.
Advertisement Modules

- AdColony
- AdMob
- Chartboost
- Domob
- Flurry
- HeyZap
- InMobi
- JumpTap
- MillennialMedia
- Mobclick

IP Distribution

The geographical distribution of the ip addresses extracted from the Android application.

IP Distribution

![IP Distribution Map](image)

Urls

Display urls embedded in this Android application and corresponding country name and ip address.

<table>
<thead>
<tr>
<th>Country</th>
<th>Url</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azerbaijan</td>
<td><a href="https://www.facebook.com/dialog/feed?id=101821551567328&amp;link=">https://www.facebook.com/dialog/feed?id=101821551567328&amp;link=</a></td>
<td>37.61.54.156</td>
</tr>
<tr>
<td>Azerbaijan</td>
<td><a href="http://twitter.com/home?status=">http://twitter.com/home?status=</a></td>
<td>37.61.54.156</td>
</tr>
<tr>
<td>China</td>
<td><a href="http://re.admob.com/impr?ad_loc=r@gw_adlcrk@Apptana@gw_qdpata@AaD_network_idr@gwi_adminid1&amp;src=gw_sdkver&amp;ssesion_idr=gwi_sessid&amp;seq_num=gwi_seqnum&amp;knr=gwi_admerfresh&amp;aad=r@gw_addj&amp;nea=gw_ace@">http://re.admob.com/impr?ad_loc=r@gw_adlcrk@Apptana@gw_qdpata@AaD_network_idr@gwi_adminid1&amp;src=gw_sdkver&amp;ssesion_idr=gwi_sessid&amp;seq_num=gwi_seqnum&amp;knr=gwi_admerfresh&amp;aad=r@gw_addj&amp;nea=gw_ace@</a></td>
<td>203.208.37.13</td>
</tr>
</tbody>
</table>
Sensitive Files

Display the sensitive files in this Android application.

<table>
<thead>
<tr>
<th>File Path</th>
<th>File Type</th>
<th>APK</th>
</tr>
</thead>
<tbody>
<tr>
<td>assets/na.so</td>
<td>zip archive data, at least v2.0 to extract</td>
<td></td>
</tr>
<tr>
<td>assets/com.so</td>
<td>zip archive data, at least v2.0 to extract</td>
<td></td>
</tr>
</tbody>
</table>

Figure 22 Sensitive Files

Native Codes

Display the information about calling native codes, such as library name, caller code, etc.

<table>
<thead>
<tr>
<th>Lib Name</th>
<th>Caller Code</th>
<th>Path index</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td>L.com/unity3d/Player/UnityPlayer-&gt;loadLibrary(String,string)</td>
<td>4</td>
</tr>
<tr>
<td>mono</td>
<td>L.com/unity3d/Player/UnityPlayer-&gt;x(IV)</td>
<td>4</td>
</tr>
<tr>
<td>unity</td>
<td>L.com/unity3d/Player/UnityPlayer-&gt;x(IV)</td>
<td>64</td>
</tr>
</tbody>
</table>

Figure 23 Native Codes

Dynamic Loaders

Display the information about dynamic loaders in this Android application.

<table>
<thead>
<tr>
<th>Dex Path</th>
<th>Lib Path</th>
<th>Caller Code</th>
<th>Path Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>/data/app/com.cps_multiplier.games.swingrace-rifiers-1.apk</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>L.com/android/tbtc:-&gt;b(Landroid.content.Context) LoanWikiSystem/Dec1ClassLoader;</td>
<td>156</td>
</tr>
</tbody>
</table>

Figure 24 Dynamic Loaders

Crypto Operation

Display the information about crypto behavior in this Android application including algorithm used, plain text, cipher text, etc.
Crypto Operation

<table>
<thead>
<tr>
<th>Operation</th>
<th>Algorithm</th>
<th>Key Encoded</th>
<th>Key Format</th>
<th>Plain Text</th>
<th>Cipher Text</th>
</tr>
</thead>
</table>
| encryption  | AES/CBC             | 36302351323130264696375646696162 | R&M        | 35724043337511...7473F6E 826920A914038A4802DA0D | 9ad6f8naxia7saz8986 <39ae9ed8obx98f9fed | ```

Figure 25 Crypto Operation

Socket Connections

Display the socket connection information including remote address and remote port.

<table>
<thead>
<tr>
<th>Remote Address</th>
<th>Remote Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>cisco.kicks-ass.net</td>
<td>9999</td>
</tr>
<tr>
<td>cisco.kicks-ass.net</td>
<td>2641</td>
</tr>
<tr>
<td>cisco.kicks-ass.net</td>
<td>9999</td>
</tr>
<tr>
<td>cisco.kicks-ass.net</td>
<td>2965</td>
</tr>
</tbody>
</table>

Figure 26 Socket Connections

File Operations

Display the information about file operations during dynamic analysis.

<table>
<thead>
<tr>
<th>Operation</th>
<th>File Path</th>
<th>Data</th>
</tr>
</thead>
</table>
| write     | /mnt/data/installer.com/socket/gts/shared_prefs | < unarmed version=1.0 encoding=utf-8 standalone=yes ? |>

Figure 27 File Operations

DNS Query

Display DNS query information during dynamic analysis, such as QName, QType and ip addresses.

<table>
<thead>
<tr>
<th>QName</th>
<th>QType</th>
<th>IPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>ads.net2share.com</td>
<td>A</td>
<td>216.158.77.154, 104.237.51.202</td>
</tr>
<tr>
<td>s.net2share.com</td>
<td>A</td>
<td>78.46.100.240</td>
</tr>
</tbody>
</table>

DNS Query
HTTP Data

Display information about HTTP Data generated during dynamic analysis.

<table>
<thead>
<tr>
<th>Method</th>
<th>Host</th>
<th>Path</th>
<th>Status Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>ads.net2share.com</td>
<td>/config/1+bad.jsr?app=ciw</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td>GET</td>
<td>ads.net2share.com</td>
<td>/temporary_api/1.9/ad;jsr?type=app&amp;inc=true</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td>POST</td>
<td>s.net2share.com:8080</td>
<td>/beacon</td>
<td>HTTP/1.1 200 OK</td>
</tr>
</tbody>
</table>

Files Recovered From Http

Display information about files recovered from http traffic produced during dynamic analysis.

<table>
<thead>
<tr>
<th>Domain</th>
<th>File Name</th>
<th>File Type</th>
<th>Is APK</th>
</tr>
</thead>
<tbody>
<tr>
<td>jiapk.apkshare.com</td>
<td>_api_1.php_get_040303index.html</td>
<td>ASCII text, with very long lines, with no line terminators</td>
<td>○</td>
</tr>
</tbody>
</table>

Execute Shells

Display the shell commands executed by this Android application.

```
/system/bin/getprop
/system/bin/getprop
/system/bin/getprop
```

Started Services

Display the information of started services.

```
 intent { action=com.google.app.DebugService ACTION_START cmp=installer.com.rockstar.gtas:j com.adc.re.qsapp.DebugService (has extras) }
 intent { action=com.google.app.DebugService ACTION_START cmp=Installer.com.rockstar.gtas:j com.adc.re.qsapp.DebugService (has extras) }
```
Figure 32 Started Services

May Send SMS

Extract SMS sending information through static analysis.

Figure 33 May Send SMS

Send SMS

Display SMS sent information based on dynamic analysis.

Figure 34 Send SMS

Block SMS

Display the information of the sms blocking behaviors.

Figure 35 Block SMS
Data Leakage

Display information about data leakage during dynamic analysis.

<table>
<thead>
<tr>
<th>Leak Type</th>
<th>Leak Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Write</td>
<td>TAIN_Tمهند</td>
</tr>
<tr>
<td>File Write</td>
<td>TAIN_Tمهند</td>
</tr>
</tbody>
</table>

Figure 36 Data Leakage

Sensitive APIs

Display information about sensitive APIs.

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
<th>Caller Code</th>
<th>Threat Level</th>
<th>Path Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android/Telephony</td>
<td>GetDeviceId</td>
<td><code>Lcom/us/telephony/device/TelephonyManager; - &gt; getDeviceId()</code></td>
<td>``Ljava/lang/</td>
<td>62</td>
</tr>
<tr>
<td>Android/Telephony</td>
<td>GetDeviceId</td>
<td><code>Landroid/content/Context; - &gt; getDeviceId()</code></td>
<td>``Ljava/lang/String;</td>
<td>54</td>
</tr>
</tbody>
</table>

Figure 37 Sensitive APIs

Permission Usage

In Android, if you want to call some special functions, you have to declare corresponding permissions in AndroidManifest.xml. As shown in Figure 38, it displays information about permission used in this apk, such as callee code correspond to permission.
Permission Usage

- Permission Name: android.permission.ACCESS_NETWORK_STATE
- Caller Code: android/support/v4/net/ConnectivityManagerCompatGingerbread/ConnectivityManagerCompatImpl.<isActiveNetworkMetered
  (android/net/ConnectivityManager);Z
- Caller Code: android/support/v4/net/ConnectivityManagerCompatGingerbread.<isActiveNetworkMetered();android/net/ConnectivityManager;Z
- Path Index: 0

Figure 38 Permission Usage

Log Message

Display the logging messages during dynamic analysis.

Figure 39 Log Message

May Log Message

Display the information about log message extracted via static analysis.

Figure 40 May Log Message

ScreenShots

The screenshots of the Android application’s running on the Android
device.

**Screenshot**

Figure 41 ScreenShots