Android Malware Pattern Recognition for Fraud and Attribution
Index

- Introducing Mobile App Singularities
- Case Study 1: Who is behind FOBUS aka “PODEC”?
- Case Study 2: SHUABANG. How we discovered one of the most ingenious ways of fooling Google Play.
- Conclusions
Some thoughts...

“83 % of the top 400 Mobile android Apps come from few unique developers”

The same applies for malware

Figure 3. The scheme of actors involved in cybercrime. Source: INTERPOL

Android App Developers, Leaders Matrix
Some thoughts...

The entire ecosystem is potentially insecure even by using "trusted" markets.

Malware slips through, it is impossible to review in all the detail required all the apps.

Cyber criminals find ways to trick the controls (i.e. Gremlin apps).

“TRUST” is not a security control.
New Research Finds Mobile Malware Infections Overhyped in US

Research Conducted on 50% of US Mobile Traffic Finds You are 1.3 Times More Likely To Get Struck By Lightning Than Have Mobile Malware Communicating on Your Device

Investigación sobre el 50% de los móviles estadounidenses conlleva que tienes 1.3 más posibilidades de ser alcanzado por un rayo que de tener malware en tu dispositivo de comunicación

Malware, All Malware: How Free Software Advocate Richard Stallman Sees Windows, Android And iOS

By Sreenu Ponnuru, Tech Times | May 27, 7:39 AM

Richard Stallman, a computer programmer and free software activist, says that OS, Android and Windows are systems. Stallman believes the

In an opinion piece in The Guardian, Stallman suggests that nearly all operating systems, whether desktop operating system or mobile operating system, can be considered malware. Stallman argues that any software that is not distributed free of cost is malware.

Stallman, who founded the Free Software Foundation, also made it clear that he is not talking about any kind of...

Google guru: Android doesn’t have malware, it has Potentially Harmful Applications™ instead

And who installs five AV apps on their mobes?
Android and Google Play, they do indeed have great security measures and are getting better.

But what if... malware (ok, PHA) do not need to break them?

• Adware
• RATs
• Trojans
• mRansomware
• ...

...all with a significant install rate. At least enough to keep production... why would attackers invest time in creating them otherwise?
Big Data, intelligence & malware

- With almost **1.5M apps** in each Google Play and App Store, with 2-3k new apps every day, traditional approaches are condemned to always be one step behind the attackers.

- Modern Cybersecurity is about leveraging Big Data Analytics & Intelligence to become **proactive**.

- Can we apply similar techniques in order to **change the pace in mobile app security**?
Creating malware **professionally is very demanding.**

Attackers repeat patterns and make mistakes like any other SW developers, thus introducing **Singularities.**

Having tools to easily crawl for such Singularities at the **moment of submission and at scale** greatly enhances **detection and attribution** capabilities.

Apps with more votes than downloads
Attribution: Who is behind FOBUS aka "PODEC"?

CASE Study 1
Fobus (aka “podec”) is a very hard to uninstall malware. It spies on the phone and is able to make calls, steal data, etc. And it is incredibly hard to remove.

We started the analysis with a sample, shown in blog of Avast.

Fobus, the sneaky little thief that could

One small Android application shows lots of determination and persistence. Too bad it's evil.
Narrowing the scope - reusing files

Files inside the original APK are reused among malicious apps: x.app & notification.png seems to be very uncommon.
Narrowing the scope - Analyzing permissions

Comparing both apps, they have a lot of permissions (basically full control of the device) and exactly the same.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>com.android.launcher.permission.INSTALL_SHORTCUT</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission SEND_SMS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission PROCESS_OUTGOING_CALLS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission WRITE_EXTERNAL_STORAGE</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission WRITE_CALL_LOG</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission WRITE_SMS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission ACCESS_WIFI_STATE</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission ACCESS_COARSE_LOCATION</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission RECEIVE_SMS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission CALL_PHONE</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission READ_CONTACTS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission WRITE_CONTACTS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission READ_PHONE_STATE</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission READ_SMS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission RECEIVE_BOOT_COMPLETED</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission INTERNET</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission WRITE_SETTINGS</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>android.permission ACCESS_FINE_LOCATION</td>
<td>☑</td>
<td>☑</td>
</tr>
</tbody>
</table>
Narrowing the scope - Dates & digital certificate

- Same update, uploaded and signing dates ➔ Singularity
- Different developers and digital certificates
- We already have a name ‘Yan Datsuk’ and a company ‘Fuzzle Pun’
- Both apps are in Russian
- A deeper analysis confirms that both apps are Fobus.
HUMINT – Profiling the bad guy

- Location
- Company profile (VK)
- Personal profile (VK, Facebook, Google+)
- Professional profile (LinkedIn)
- HackerOne profile
- Email
- Relationship between company & developer
SHUABANG: How we discovered one of the most ingenious ways of fooling Google Play

CASE 2
Black ASO {App Store Optimization}

- Black App Store Optimization
  - Positioning apps in markets
    - Spam
    - Download rate
    - Reviews

- Goal
  - Build up an automated infrastructure
  - Infrastructure for sale to third parties
  - To extend the botnet
What is Shuabang?

- Shuabang is a technique, quite common in China. It is the BlackHat App Store Optimization.

- There are companies that charge you to “rise up” your app in stores, voting them, or adding fake downloads.

- To get that in Google Play, they need registered users in Google Play, that means, basically, Gmail accounts (users and passwords) associated with a telephone (a real deviceID). How to get them?
  - You can buy them.
  - You can create them.
  - You can steal them.
  - Or you can create your own botnet...
1. The attacker recovers a Google Security Token and links it to a fake Google user account and a specific device.

2. From the malware app the Google check-in process is done using that token.

3. If the token is valid, the attacker provides a user name and a password to the malware app.

4. From the malware app on the device, credentials and device config are uploaded.

5. If the credentials are valid, Google Play returns a new Security token as a response.

6. With this new token, the malware app is able to execute several tasks and actions, ordered by the attacker in the background.
We found something that looked very suspicious...

- We were looking for **Wallpapers that connected to PHP sites (title:*wallpaper* links:*\.php permissionName:*ACCOUNT* permissionName:*BOOT*)**. That simple. And we found this...

```
<table>
<thead>
<tr>
<th>ID</th>
<th>Username</th>
<th>Password</th>
<th>Created Time</th>
<th>Device OS</th>
<th>Region</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>10661</td>
<td><a href="mailto:stcloudby29@gmail.com">stcloudby29@gmail.com</a></td>
<td>--------</td>
<td>2014-11-06,23:32:10</td>
<td></td>
<td>巴西</td>
<td></td>
</tr>
<tr>
<td>10662</td>
<td><a href="mailto:amandacq92@gmail.com">amandacq92@gmail.com</a></td>
<td>--------</td>
<td>2014-10-29,18:26:43</td>
<td></td>
<td>巴西</td>
<td></td>
</tr>
<tr>
<td>10663</td>
<td><a href="mailto:besscwb6@gmail.com">besscwb6@gmail.com</a></td>
<td>--------</td>
<td>2014-11-06,23:37:14</td>
<td></td>
<td>巴西</td>
<td></td>
</tr>
<tr>
<td>10664</td>
<td><a href="mailto:isaacpd70@gmail.com">isaacpd70@gmail.com</a></td>
<td>--------</td>
<td>2014-10-29,18:34:31</td>
<td></td>
<td>巴西</td>
<td></td>
</tr>
<tr>
<td>10665</td>
<td><a href="mailto:milstdh@gmail.com">milstdh@gmail.com</a></td>
<td>--------</td>
<td>2014-11-06,23:42:50</td>
<td></td>
<td>巴西</td>
<td></td>
</tr>
</tbody>
</table>
```

![User Accounts](image)
<table>
<thead>
<tr>
<th></th>
<th>Program</th>
<th>URL/Host</th>
<th>Login</th>
<th>Password</th>
<th>Computer</th>
<th>Date</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>361</td>
<td>Firefox</td>
<td><a href="https://login.facebook.com">https://login.facebook.com</a></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>362</td>
<td>Firefox</td>
<td><a href="https://www.google.com">https://www.google.com</a></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>363</td>
<td>Firefox</td>
<td><a href="https://addons.mozilla.org">https://addons.mozilla.org</a></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>364</td>
<td></td>
<td><a href="mailto:confusedmine@gmail.com">confusedmine@gmail.com</a></td>
<td>pleasefuckoff</td>
<td>ZARDOZ</td>
<td>2011-10-16 12:51:54</td>
<td>71.35.156.85</td>
<td></td>
</tr>
<tr>
<td>365</td>
<td></td>
<td><a href="mailto:confusedmine@gmail.com">confusedmine@gmail.com</a></td>
<td>pleasefuckoff</td>
<td>ZARDOZ</td>
<td>2011-10-16 12:51:55</td>
<td>71.35.156.85</td>
<td></td>
</tr>
<tr>
<td>366</td>
<td></td>
<td><a href="mailto:confusedmine@gmail.com">confusedmine@gmail.com</a></td>
<td>pleasefuckoff</td>
<td>ZARDOZ</td>
<td>2011-10-16 12:51:55</td>
<td>71.35.156.85</td>
<td></td>
</tr>
</tbody>
</table>
### Shuaban Botnet: Control Panel

#### Phone

<table>
<thead>
<tr>
<th>国家</th>
<th>账号个数</th>
</tr>
</thead>
<tbody>
<tr>
<td>巴西</td>
<td>6964</td>
</tr>
<tr>
<td>HK</td>
<td>2</td>
</tr>
<tr>
<td>印度</td>
<td>1701</td>
</tr>
<tr>
<td>俄罗斯</td>
<td>2000</td>
</tr>
</tbody>
</table>

#### Appid Appversion Appot Country

<table>
<thead>
<tr>
<th>appid</th>
<th>appversion</th>
<th>appot</th>
<th>country</th>
<th>空闲账号</th>
<th>限制次数</th>
<th>下载量</th>
<th>失败次数</th>
<th>等待个数</th>
<th>等待未请求任务</th>
<th>失效账号</th>
<th>操作</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>巴西</td>
<td>1</td>
<td>1</td>
<td>4111</td>
<td>6123</td>
<td>4395</td>
<td>4379</td>
<td>0</td>
<td>97</td>
<td>910</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>印度</td>
<td>1</td>
<td>1</td>
<td>187</td>
<td>610</td>
<td>1055</td>
<td>1072</td>
<td>1</td>
<td>78</td>
<td>137</td>
</tr>
</tbody>
</table>
And this is what it was...

SHA256: 34e9927358bcb56c3ce0ef09fd71bddd48cfc22b949f565de77df0e8b7b93c99
Nombre: com.businessprisonice.eletircscreen.apk
Detecciones: 0 / 48
Fecha de análisis: 2014-10-30 11:00:35 UTC (hace 4 meses, 1 semana)

<table>
<thead>
<tr>
<th>Antivirus</th>
<th>Resultado</th>
<th>Actualización</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>✔</td>
<td>20141030</td>
</tr>
<tr>
<td>Ad-Aware</td>
<td>✔</td>
<td>20141030</td>
</tr>
<tr>
<td>AegisLab</td>
<td>✔</td>
<td>20141030</td>
</tr>
</tbody>
</table>
These applications had several points in common that could be used as the developer's “fingerprint“.

- The apps were created with Java version 1.8.0_05 (Oracle Corporation).
- The certificates, although different, were valid for 271 years.
- `certificateSubjectCommonName` is normally formed by a combination of several words.
- Images were also useful to find similar applications. For example, if a search was made for similar images, a different developer may be found: **yu jinhui**.

And finally we found this in common:

http://apptools.myappblog.net/selfpush/selfpush/gameframe/www/test/getcontent.php (links:*myappblog*)
The number of different developers took a huge bump.

**shengzls, feng wenjie, tong ronghai, shui hongli, yan dongba, tang xiaocan, wan lichun, jie libao, wen xiaojian, yuan junrong, wen xiaojian...**
Finally it got baptized...

<table>
<thead>
<tr>
<th>SHA256</th>
<th>Nombre</th>
<th>Detecciones</th>
<th>Fecha de análisis</th>
</tr>
</thead>
<tbody>
<tr>
<td>227bd4004a2bb5b9431645e5204c3f5c4d5f35fb863c359f00c94bb0f74f8900</td>
<td>227bd4004a2bb5b9431645e5204c3f5c4d5f35fb863c359f00c94bb0f74f8900</td>
<td>3 / 56</td>
<td>2014-12-11 10:39:45 UTC (hace 2 meses, 3 semanas)</td>
</tr>
</tbody>
</table>

**Antivirus** | **Resultado** | **Actualizado**
--- | --- | ---
ESET-NOD32 | a variant of Android/Glocken.A | 20141211
Ikarus | Trojan AndroidOS ShuaBing | 20141211
Kaspersky | not-a-virus HEUR Risk Tool AndroidOS Bauta | 20141211
Shuabang Botnet: Facts

- Stolen 12,567 Google accounts
- Expert understanding of internals (unpublished) of the device registration system with.
- 100 malicious apps available in Google Play
- Permissions apparently harmless
- Complex tasking system managed effectively
- Fraud-per-click managed in a smart way
- Resources usage without using Google original account in the infected device
- Building up a development framework
Conclusions
How high were those groups in the food chain?

Figure 3. The scheme of actors involved in cybercrime. Source: INTERPOL
What should we do about all this?

Mobile apps are pervasive in our digital world. They represent a huge opportunity for criminal activity and they are already being exploited by criminal organizations.

Official ecosystems to distribute apps have helped to increase security but have been demonstrated to not be totally failsafe.

Don’t use unofficial stores unless you know what you are doing.

Do not implicitly trust mobile apps. Depending on your security requirements the risk by not be acceptable.

Apply adequate policies through MDM solutions.

Consider anti-malware solutions.

Monitor your own apps.

Educate your employees.

Employ both traditional signature-based techniques and complement with Big-Data Analytics

Leverage singularities to hunt down malware as soon as they appear on the app store. We don’t have to wait for them to be installed and do harm

Combine this with OSINT source to hunt-down the authors, contact the appropriate LEA
Thank you