# Android e mobile security

relatore: Igor Falcomatà

## client side, server side, privacy

**do android malware writers dream of electric sheep?**

**Linux Day 2012**
**Firenze**

free advertising >

SOME RIGHTS RESERVED
http://creativecommons.org/licenses/by-sa/2.0/it/deed.it

# Chi:

aka “koba”

- **attività professionale:**
  - **analisi delle vulnerabilità e penetration testing (~13 anni)**
  - **security consulting**
  - **formazione**
- **altro:**
  - **sikurezza.org**
  - **(F|Er|bz)lug**

Relatore:

NETWORK enforcer SECURITY

Igor Falcomatà
Chief Technical Officer
ifalcomata@enforcer.it



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 2

SOME RIGHTS RESERVED

# Cosa:

## un po' di crusca del mio sacco..

- **App.. HTML5.. BYOD.. Cloud.. TheNextBuzzword.. come interagiscono queste componenti con la privacy degli utenti, la sicurezza dei dati sui dispositivi e sui server e l'entropia mondiale?**
- **E le buone vecchie vulnerabilità nelle applicazioni web?**
- **Esempi e dettagli su piattaforma Android**
- **Adatto in generale a chiunque sia interessato alla sicurezza delle applicazioni "mobile".**

**..molta farina dai mulini altrui!**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 3

SOME RIGHTS RESERVED

# Perché (device):

## malware/exploit writer's dream platform?

- diffusione e “geopardizzazione” (AUGH!)
- sorgenti (AOSP), docs, SDK, NDK, emulatore, ..
- .apk → decompilazione, reversing, debug
- aggiornamenti OS, app e market alternativi
- permessi delle applicazioni “delegati” agli utenti
- Linux Kernel, ~ Linux userspace e librerie (e bug)
- exploit mitigation techniques (fail) (< 2.3, < 4.0.3)
- OOB “covert” channel (umts/gprs, SMS, ..)
- territori poco explorati: OS/lib custom, hw driver


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 
SOME RIGHTS RESERVED

# Perché (utenti):

## (governi|spioni|stalker|..)'s dream platform?

- **dati personali** (posta, documenti, rubrica, calendario, ..)
- **intercettazioni** (audio, video, messaging, network, ..)
- **geolocalizzazione** (foto, social network, ..)
- **credenziali** (siti, posta, VPN, ..) → **cloud storage**
- **HTML-like client side attacks**
- **EvilApp want to eat your soul.. Install? YES!!!**
- **BY0D (Bring Your 0wned Device)**
- **banking OTP ($$)**
- **NFC ($$)**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 

SOME RIGHTS RESERVED

# Perché (back-ends):

## web application hacker's dream platform?

- **url e web-services “privati”**
- **business logic esposta (client-side)**
- **-> device -> credenziali -> back-end**
- **-> device -> storage -> back-end**
- **credenziali e certificati hard-coded (.apk)**
- **no/lazy input validation**
- **no/broken authentication & session management**
- **the good ole web security vulns**

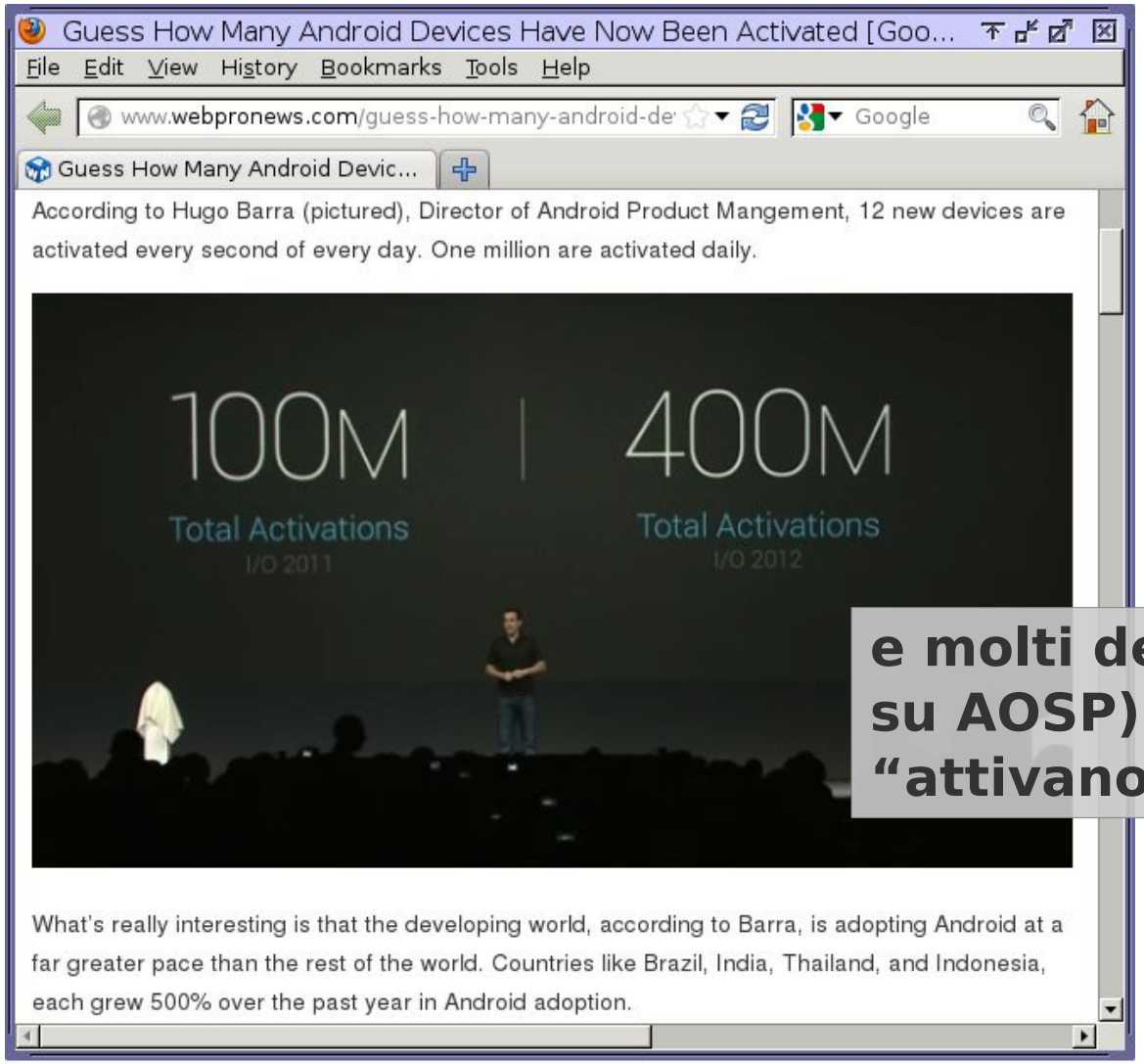


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 

SOME RIGHTS RESERVED

# Diffusione

http://www.webpronews.com/guess-how-many-android-devices-have-now-been-activated-google-io-2012-06




**e molti device (basati su AOSP) che non si "attivano" ..**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 7

SOME RIGHTS RESERVED

# Versioni

**http://developer.android.com/about/dashboards/index.html**

Dashboards | Android Developers - Mozilla Firefox

File Edit View History Bookmarks Tools Help

developer.android.com/about/dashboards/index.html

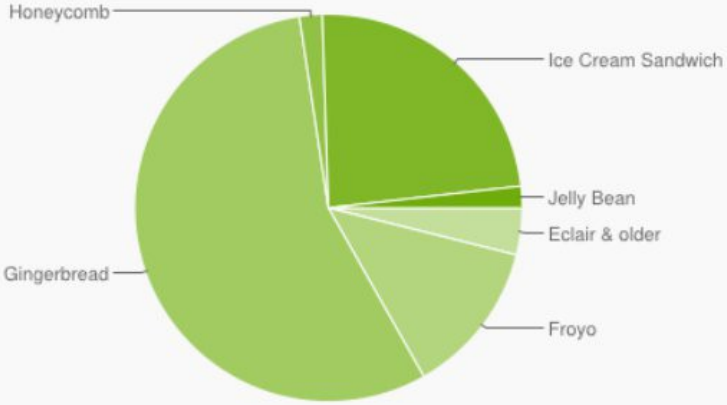
Dashboards | Android Developers

- Welcome
- Jelly Bean
- Ice Cream Sandwich
- Honeycomb
- Gingerbread
- Dashboards

## Current Distribution

The following pie chart and table is based on the number of Android devices that have accessed Google Play within a 14-day period ending on the data collection date noted below.

| Version | Codename | API | Distribution |
|---|---|---|---|
| 1.5 | Cupcake | 3 | 0.1% |
| 1.6 | Donut | 4 | 0.4% |
| 2.1 | Eclair | 7 | 3.4% |
| 2.2 | Froyo | 8 | 12.9% |
| 2.3 - 2.3.2 | Gingerbread | 9 | 0.3% |
| 2.3.3 - 2.3.7 | | 10 | 55.5% |
| 3.1 | Honeycomb | 12 | 0.4% |
| 3.2 | | 13 | 1.5% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 23.7% |
| 4.1 | Jelly Bean | 16 | 1.8% |




*Data collected during a 14-day period ending on October 1, 2012*

## Historical Distribution

**e molti device che usano market alternativi ..**

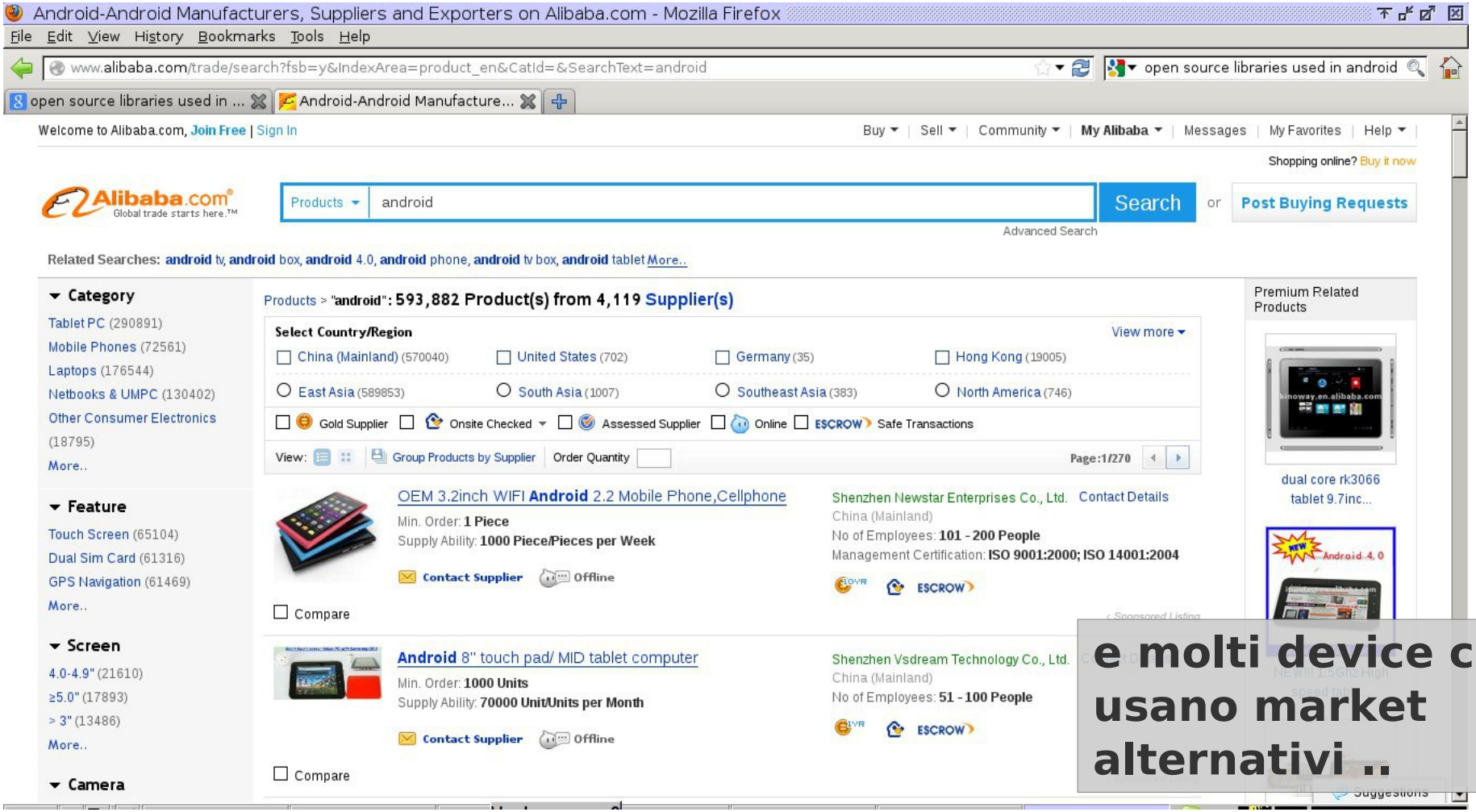


**© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 8**

SOME RIGHTS RESERVED

# (Low Cost) Devices

http://www.alibaba.com/trade/search?fsb=y&IndexArea=product_en&SearchText=android
http://en.wikipedia.org/wiki/Comparison_of_Android_devices




**e molti device che usano market alternativi ..**

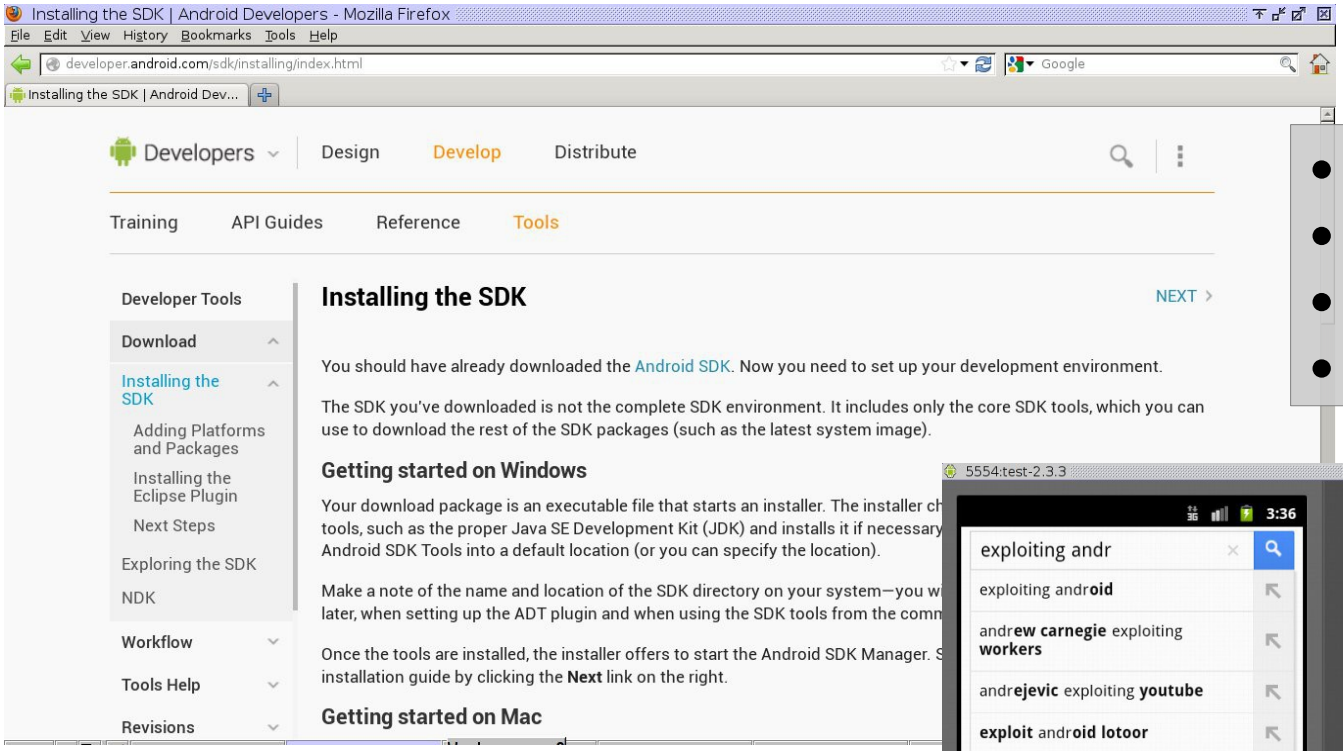


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 9

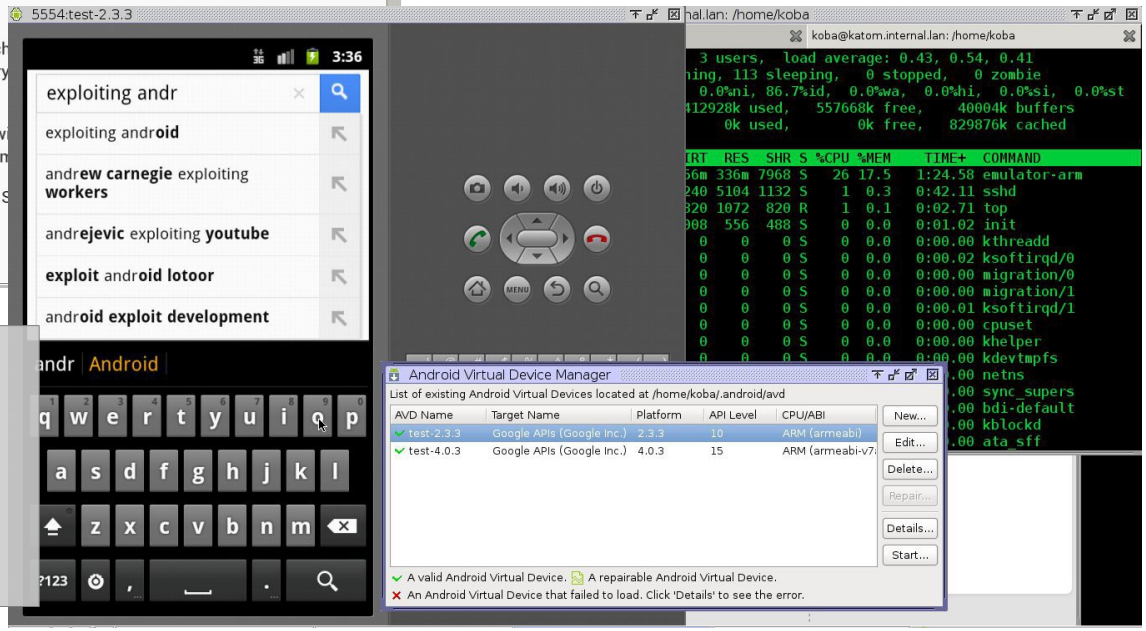
SOME RIGHTS RESERVED

# Docs & Tools

http://developer.android.com/




- API
- Esempi & Howto
- Sorgenti (AOSP)
- ..




- SDK/NDK
- Eclipse plugin (ADT)
- Emulatore (Arm, Intel, ..)
- debug (ADB, ..)



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 10

SOME RIGHTS RESERVED

# Exploiting Android is c00l!

**http://cc.thinkst.com/searchMore/android/**

Con Collector - Mozilla Firefox

File Edit View History Bookmarks Tools Help

cc.thinkst.com/searchMore/android/

ConCollector

thinkst

our blog | more on thinkst

Search: Everything Browse: Speakers Conferences Topics Contributors Analytics Folklore About Login

**Search Results:** 124 hits in 0.943671 seconds searching for android

***Talk Titles and Abstracts: 123 results***

- Team Joch Vs. Android: The Ultimate Showdown — Zach Lanier, Jon Oberheide at ShmooCon 2011
- Reverse Engineering Using The Android Emulator — Scott Dunlop at ShmooCon 2011
- Transparent Botnet Control For Smartphones Over Sms — Georgia Weidman at ShmooCon 2011
- More Bugs In More Places: Secure Development On Moble Platforms — David Kane-parry at Blackhat USA 2010
- & Tim Wyatt — Anthony Lineberry, David Luke Richardson at Blackhat USA 2010
- Exploratory Android Surgery — Jesse Burns at Blackhat USA 2009
- Smart Phones With Dumb Apps — Dan Dan cornell at House Con 2010
- What'S Old Is New Again: An Overview Of Mobile Application Security — Zach Lanier at OWASP BASC 2010
- Android Application Security, The Fun Details — Jesse Burns at hashdays 2010
- Fuzzing The Phone In Your Phone — Collin Mulliner at Chaos Communication Congress 26
- Smart Phones With Dumb Apps: Threat Modeling For Mobile Applications — Dan Dan cornell at OWASP Appsec 2010
- Pulling A John Connor: Defeating Android — Charlie Miller at ShmooCon 2009
- Geschichte Der Automaten, Androiden Und Homunculi — Henriette Fiebig at Chaos Communication Congress 23
- Reconstructing Dalvik Applications (Google Android) — Marc Schoenfeld at SyScan 2009

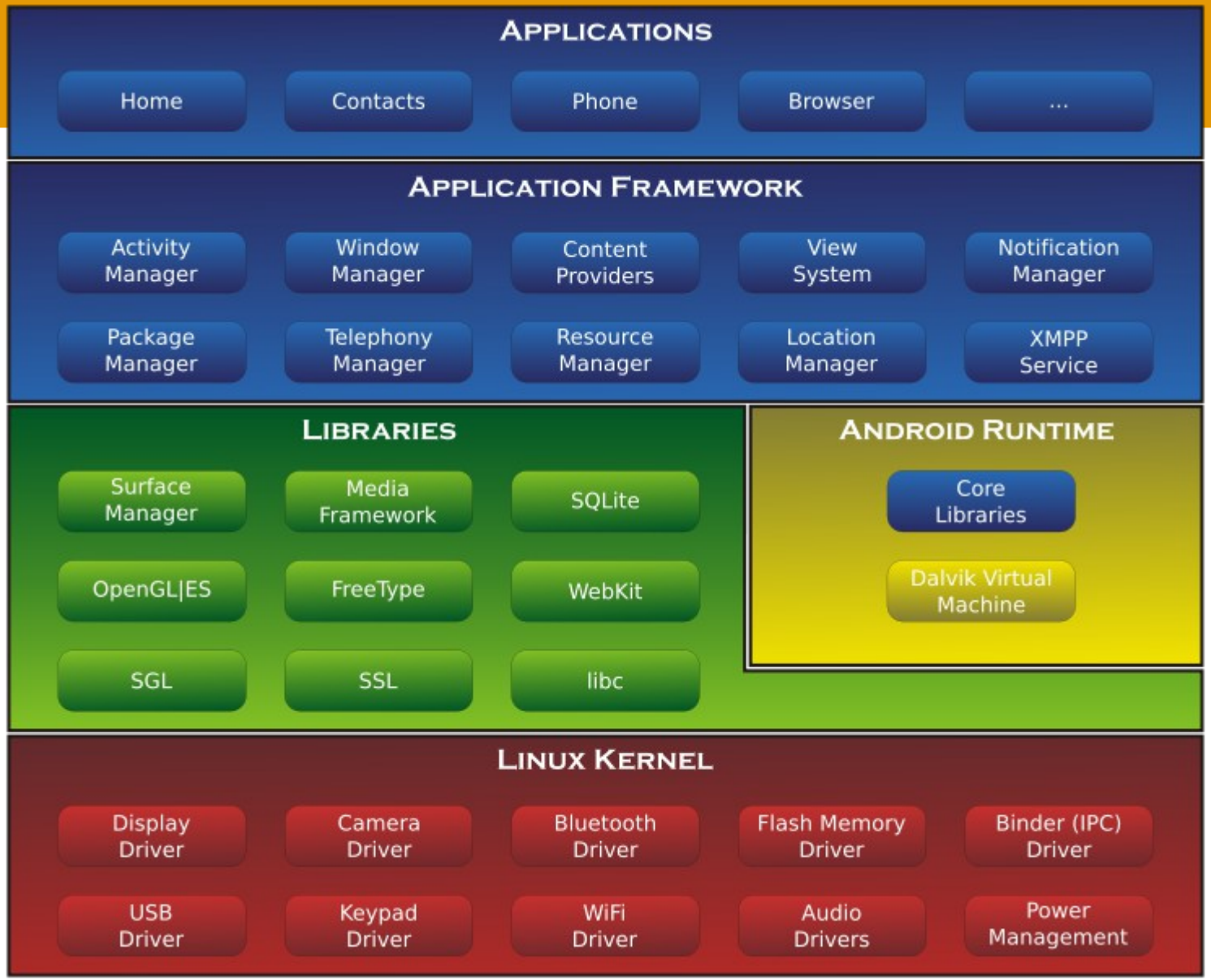
**+ google, slideshare, stackoverflow, ypse, ..**



**© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 11**

SOME RIGHTS RESERVED

# Android software stack




**http://en.wikipedia.org/w/index.php?title=File:Android-System-Architecture.svg**



**© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it** 

SOME RIGHTS RESERVED

# Kernel

http://en.wikipedia.org/wiki/Android_(operating_system)#Linux
http://elinux.org/Android_Kernel_Features#Kernel_features_unique_to_Android

- Architetture: ARM, (MIPS, x86, ..)
- Kernel
  - Kernel Linux 2.6.x (Android 1, 2 e 3.x)
  - Kernel Linux 3.0.x (Android 4.x)
  - componenti e driver standard
  - FS, processi, permessi, processi
  - vulnerabilità standard ;)
- Componenti custom
  - binder, ashmem, pmem, logger, wavelocks, OOM, alarm timers, paranoid network security, gpio, ..
  - android e vendor custom hw driver
  - nuove vulnerabilità da scoprire ;)


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 13
SOME RIGHTS RESERVED

# Librerie + VM

http://source.android.com/tech/security/index.html#the-application-sandbox
http://en.wikipedia.org/wiki/Dalvik_(software)

- **Sandbox (OS level)**
  - sandboxing con uid/gid linux + patch kernel (protected API)
  - 1 processo = 1 applicazione = 1 VM (+ componenti OS)
  - protected API per accesso all'hw: camera, gps, bluetooth, telefonia, SMS/MMS, connessioni di rete)
  - root = root (full access)
- **Librerie**
  - bionic libc (!= gnu libc, !posix)
  - udev, WebKit, OpenGL, SQLite, crypto, .. (& bugs)
- **Dalvik VM (!= JVM)**
  - Java Code -> dex bytecode
  - custom Java libraries
  - può lanciare codice nativo (syscall, ioctls, .. ) -> kernel


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 
SOME RIGHTS RESERVED

# Librerie + VM

http://source.android.com/tech/security/index.html#the-application-sandbox
http://en.wikipedia.org/wiki/Dalvik_(software)

- **Sandbox (OS level)**
  - sandboxin
  - 1 processo
  - protected
    telefonia,
  - root = roo
- **Librerie**
  - bionic libc
  - udev, Web
- **Dalvik VM**
  - Java Code
  - custom Jav
  - può lanciare codice nativo (syscall, ioctls, .. ) -> kernel

> **“Like all security features, the Application Sandbox is not unbreakable. However, to break out of the Application Sandbox in a properly configured device, one must compromise the security of the the Linux kernel.”**


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 15
SOME RIGHTS RESERVED

# Root(ing)

http://source.android.com/tech/security/index.html#rooting-of-devices

- Mozilla Firefox

f-devices

Google

## Rooting of Devices

By default, on Android only the kernel and a small subset of the core applications run with root permissions. Android does not prevent a user or application with root permissions from modifying the operating system, kernel, and any other application. In general, root has full access to all applications and all application data. Users that change the permissions on an Android device to grant root access to applications increase the security exposure to malicious applications and potential application flaws.

The ability to modify an Android device they own is important to developers working with the Android platform. On many Android devices users have the ability to unlock the bootloader in order to allow installation of an alternate operating system. These alternate operating systems may allow an owner to gain root access for purposes of debugging applications and system components or to access features not presented to applications by Android APIs.

On some devices, a person with physical control of a device and a USB cable is able to install a new operating system that provides root privileges to the user. To protect any existing user data from compromise the bootloader unlock mechanism requires that the bootloader erase any existing user data as part of the unlock step. Root access gained via exploiting a kernel bug or security hole can bypass this protection.

Encrypting data with a key stored on-device does not protect the application data from root users. Applications can add a layer of data protection using encryption with a key stored

**meglio sviluppare sull'emulatore o su un device apposito :)**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 16

SOME RIGHTS RESERVED

# Aggiornamenti

https://developer.android.com/guide/faq/security.html#fixes

Android Security FAQ | Android Developers - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://developer.android.com/guide/faq/security.html#fixes

Google

- App Components
- User Interface
- App Resources
- Animation and Graphics
- Computation
- Media and Camera
- Location and Sensors
- Connectivity
- Text and Input
- Data Storage
- Administration
- Web Apps
- Best Practices
- Google Services

## How do Android-powered devices receive security fixes?

The manufacturer of each device is responsible for distributing software upgrades for it, including security fixes. Many devices will update themselves automatically with software downloaded "over the air", while some devices require the user to upgrade them manually.

Google provides software updates for a number of Android devices, including the Nexus series of devices, using an "over the air" (OTA) update. These updates may include security fixes as well as new features.

## Can I get a fix directly from the Android Platform Project?

Android is a mobile platform that is released as open source and available for free use by anybody. This means that there are many Android-based products available to consumers, and most of them are created without the knowledge or participation of the Android Open Source Project. Like the maintainers of other open source projects, we cannot build and release patches for the entire ecosystem of products using Android. Instead, we will work diligently to find and fix flaws as quickly as possible and to distribute those fixes to the manufacturers of the products through the open source project.

If you are making an Android-powered device and would like to know how you can properly support your customers by keeping abreast of software updates, please contact us at info@openhandsetalliance.com.

- **aggiornamenti delegati ai carrier/vendor ...**
- **aftermarket/homebrew (cyanogenmod, ..)**
- **aggiornamento app via market**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 17

SOME RIGHTS RESERVED

# Exploit mitigation techniques

https://developer.android.com/guide/faq/security.html#fixes

https://blog.duosecurity.com/2012/07/exploit-mitigations-in-android-jelly-bean-4-1/

Open Source - Mozilla Firefox

Help

.html#memory-management-security-enhancements

Google

## Memory Management Security Enhancements

Android includes many features that make common security issues harder to exploit. The Android SDK, compilers, and OS use tools to make common memory corruption issues significantly harder to exploit, including:

**Android 1.5+**

- ProPolice to prevent stack buffer overruns (-fstack-protector)
- safe_iop to reduce integer overflows
- Extensions to OpenBSD dlmalloc to prevent double free() vulnerabilities and to prevent chunk consolidation attacks. Chunk consolidation attacks are a common way to exploit heap corruption.
- OpenBSD calloc to prevent integer overflows during memory allocation

**Android 2.3+**

- Format string vulnerability protections (-Wformat-security -Werror=format-security)
- Hardware-based No eXecute (NX) to prevent code execution on the stack and heap
- Linux mmap_min_addr to mitigate null pointer dereference privilege escalation (further enhanced in Android 4.1)

**Android 4.0+**

- Address Space Layout Randomization (ASLR) to randomize key locations in memory

**Android 4.1+**

- PIE (Position Independent Executable) support
- Read-only relocations / immediate binding (-Wl,-z,relro -Wl,-z,now)
- dmesg_restrict enabled (avoid leaking kernel addresses)
- kptr_restrict enabled (avoid leaking kernel addresses)

## Rooting of Devices

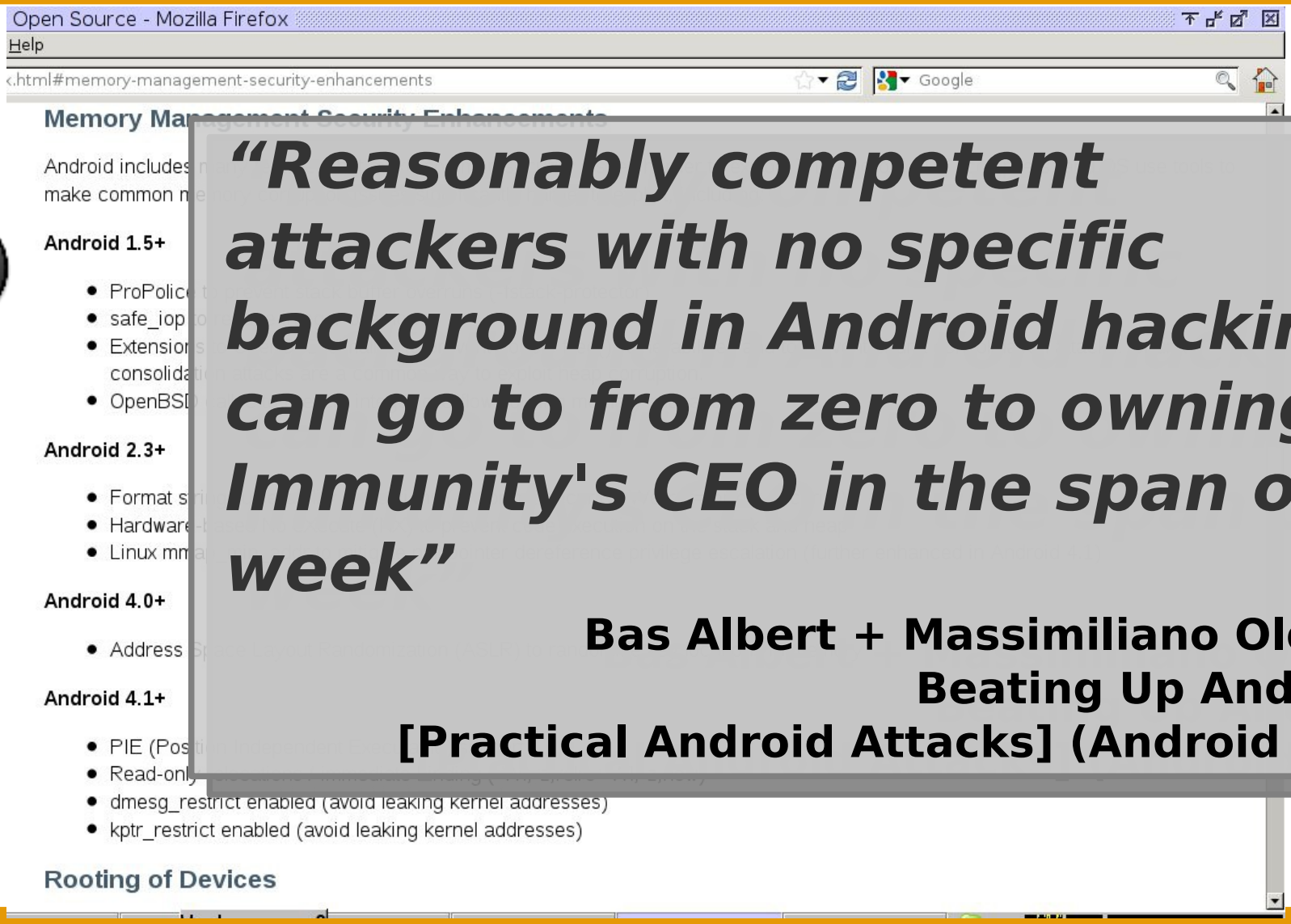


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 18

SOME RIGHTS RESERVED

# (FAIL)

http://www.immunityinc.com/infiltrate/2011/presentations/Android_Attacks.pdf




***"Reasonably competent attackers with no specific background in Android hacking can go to from zero to owning Immunity's CEO in the span of a week"***

**Bas Albert + Massimiliano Oldani**
**Beating Up Android**
**[Practical Android Attacks] (Android 2.1)**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 19

SOME RIGHTS RESERVED

# Known vulnerabilities (scanner)

**http://www.xray.io/#vulnerabilities**

X-Ray for Android - Mozilla Firefox

File Edit View History Bookmarks Tools Help

www.xray.io/#vulnerabilities

X-Ray Intro Download FAQ Vulnerabilities Contact Made by Duo Security

X-Ray is automatically updated with the ability to scan for new vulnerabilities as they are discovered and disclosed. The app currently checks for the following vulnerabilities on your Android device:

### ASHMEM

The ASHMEM custom shared memory allocator written by Google has a nasty bug that allows your device to be easily rooted. Maybe they should have stuck with POSIX SHM, but the bad guys aren't complaining.

### Exploid

Nearly identical to a vulnerability fixed in the Linux udev daemon in 2009, Exploid impacts Android's init daemon by forgetting to check whether Netlink messages are coming from the trusted kernel ... or a malicious app.

### Gingerbreak

A commonly-abused vulnerability in the wild by Android malware, Gingerbreak affects the Android volume manager (vold) via, you guessed it, the same Netlink issue as Exploid. Badness ensues.

### Levitator

Malicious apps will undoubtedly be "levitating" their privileges using this vulnerability that affects the

### Mempodroid

Inherited from the upstream Linux kernel, a vulnerability in the /proc/pid/mem interface allows for writing arbitrary

### Wunderbar

Another crossover from the Linux kernel, this NULL pointer dereference was one of the first privilege escalation



**© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 20**

SOME RIGHTS RESERVED

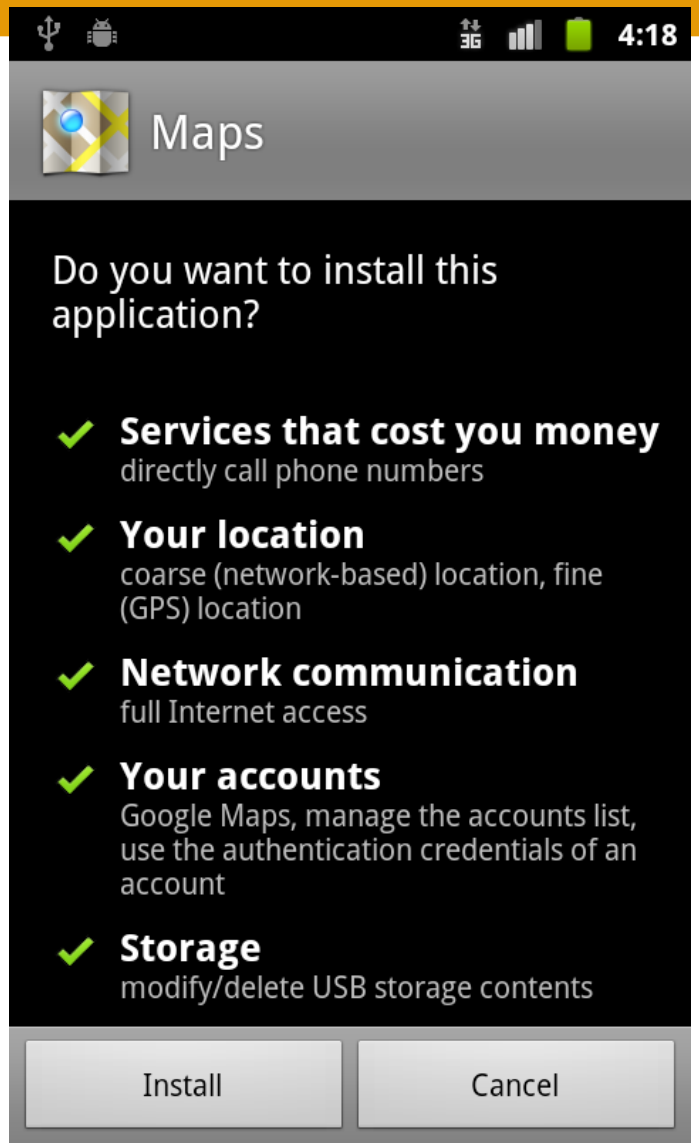
# Altri vettori d'attacco
## (molto più praticabili)

- rogue App
- trojan App
- trojan aftermarket fw (o carrier trojan ... <g>)
- traffico di rete
- client-side ~HTML attacks
- decompilazione / reversing applicazioni
- filesystem / permessi
- setuid
  - praticamente non usati in Android “stock”
  - rooted devices + software di terze parti
  - homebrew (cyanogenmod, ..)


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 
SOME RIGHTS RESERVED

# App Security Permissions

http://source.android.com/tech/security/index.html#how-users-understand-third-party-applications




**permessi definiti nel *Manifest* dell'applicazione che l'utente deve accettare in fase di installazione**

**pacchetti (.apk) firmati digitalmente per OS e Play Store ...**

**"Applications can be signed by a third-party (OEM, operator, alternative market) or self-signed. Android provides code signing using self-signed certificates that developers can generate without external assistance or permission. Applications do not have to be signed by a central authority. Android currently does not perform CA verification for application certificates."**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 22

SOME RIGHTS RESERVED

# Google Bouncer

http://www.h-online.com/security/news/item/Google-s-Bouncer-scans-the-Android-Market-for-Malware-1427814.html

er scans the Android Market for Malware - The H Security: News and Features - Mozilla Firefox

ry Bookmarks Tools Help

com/security/news/item/Google-s-Bouncer-scans-the-Android-Market-for-Malware-1427814.html

Google

THE H SECURITY

Security

You are a guest · Login | Register

In association with heise online

Search The H Security Search

Last 7 days News Archive Features Forums Newsletter RSS Downloads

Price Insight on The H powered by Skinflint

03 February 2012, 11:49

« previous | next »

## Google's Bouncer scans the Android Market for Malware

Google has apparently been scouring the Android Market in search of malicious applications since 2011. Writing on the Google Mobile Blog, Hiroshi Lockheimer, VP of Engineering, Android, explained that an automatic service named "Bouncer" also analysed new apps submitted to the market for trojans and spyware.

Bouncer runs each app in a simulated environment on Google's cloud infrastructure and simulates how the app will be run on an Android device. During this run, it is examined for malicious but hidden behaviour. In 2011, Google acquired the German security vendor Zynamics, but it is unclear whether Zynamics technology is in use in the Google malware scanning. Developer accounts are also analysed in an attempt to block developers who regularly submit malicious applications into the store.

According to Lockheimer, between the first and second half of 2011 there was a 40% drop in the number of "potentially malicious downloads from the Android Market". He notes that this decrease comes at the same time that "companies who market and sell anti-malware and security software have been reporting that malicious applications are on the rise" and suggests that the metric of number of downloads is a more accurate indicator of the impact of malware authors.

Talking with TechCrunch, Lockheimer went further, saying that "There's this impression that Android is a huge target for malware, and I really don't think that's the case". Even AV vendors can't agree what counts as malignant software now. Recently, Symantec warned

Anzeige

itsa 2012

Voucher Shop

Redeem your E-Code here

Redeem E-Code

THE H SECURITY HEADLINES

- miniFlame: the Flame trojan's little brother
- Report: Steam poses security risk
- Stormy October patch day for Oracle
- Gary McKinnon not to be extradited to the US
- Kaspersky developing OS for secure SCADA systems
- Santander's online banking keeps passwords in cookies - Update

THE H SECURITY



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 23

SOME RIGHTS RESERVED

# Google Bouncer (PWNED)

http://jon.oberheide.org/blog/2012/06/21/dissecting-the-android-bouncer/

er scans the Android Market for Malware - The H Security: News and Features - Mozilla Firefox

ry Bookmarks Tools Help

com/security/news/item/Google-s-Bouncer-scans-the-Android-Market-for-Malware-1427814.html

Google

THE H SECURITY

Security

You are a guest · Login | Register

In association with heise online

Search The H Security Search

Last 7 days News Archive Features Forums Newsletter RSS Downloads

Price Insight on The H powered by Skinflint

03 February 2012, 11:49

« previous | next »

## Google's Bouncer scans the Android Market for Malware

Google has apparently been scouring the Android Market in search of malicious applications since 2011. Writing on the Google Mobile Blog, Hiroshi Lockheimer, VP of Engineering, Android, explained that an automatic servi
submitted to the m

Bouncer runs each
cloud infrastructur
Android device. Du
hidden behaviour.
is unclear whether
Developer account
submit malicious a

According to Lockh
in the number of "p
this decrease com
and security softw
suggests that the r
impact of malware

Talking with TechC
that Android is a huge target for malware, and I really don't think that's the case. Even AV vendors can't agree what counts as malignant software now. Recently, Symantec warned

Anzeige

itsa 2012 Voucher Shop

Print Version | Send by email | Permalink: http://h-online.com/-1427814

**Also on The H:**

"Pre-loaded" PC malware leads to domain takeover
AV test lab examines Android security programs
Google tightens up rules for Play store applications
Google's anti-malware Bouncer too tolerant
Report: Android malware doubled in just one month
Google's Bouncer malware scanner for Android pwned

Share this article



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 24

SOME RIGHTS RESERVED

# Rogue App

**http://blogs.mcafee.com/mcafee-labs/android-malware-pairs-man-in-the-middle-with-remote-controlled-banking-trojan**

Android Malware Pairs Man-in-the-Middle With Remote-Controlled Banking Trojan | Blog Central - Mozilla Firefox

File Edit View History Bookmarks Tools Help

blogs.mcafee.com/mcafee-labs/android-malware-pairs-man-in-the-middle-with-remote-controlled-banking-trojan

When the application executes, it shows a WebView component that displays an HTML/JavaScript web page that pretends to be a Token Generator. The web page also appears to be from the targeted bank (same variant of the malware but with different payload):

To get the fake token, the user must enter the first factor of authentication (used to obtain initial access to the banking account). If this action is not performed, the application shows an error. When the user clicks "Generar" (Generate), the malware shows the fake token (which is in fact a random number) and sends the password to a specific cell phone number along with the device identifiers (IMEI and IMSI). The same information is also sent to one of the control servers along with further data such as the phone number of the device. The malware finds the list of control servers from an XML file inside the original APK. This information, along with other parameters of the malware, are loaded and stored in another XML file inside the device:

```
−<settingsSet>
    <catchSmsList class="java.util.ArrayList"/>
    <deleteSmsList class="java.util.ArrayList"/>
    <number>7[redacted]7</number>
    <version>1.0</version>
```



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 25

SOME RIGHTS RESERVED

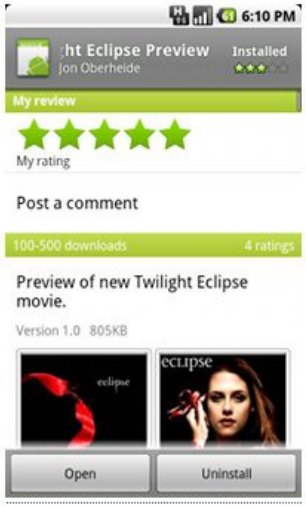
# Trojan App

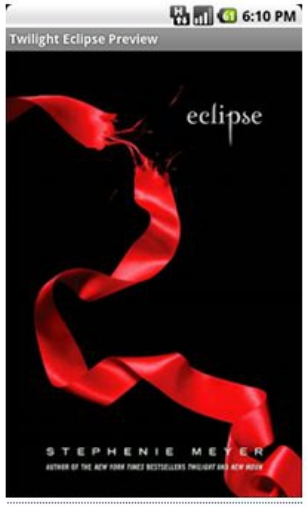
http://jon.oberheide.org/blog/2010/06/25/remote-kill-and-install-on-google-android/
http://jon.oberheide.org/files/summercon10-androidhax-jonoberheide.pdf

log - remote kill and install on google android - Mozilla Firefox
ookmarks Tools Help
og/2010/06/25/remote-kill-and-install-on-google-android/

## RootStrap Background

So if you didn't check out my slides from SummerCon last week in NYC, I talked a bit about a program called RootStrap in the second half of my talk. RootStrap is intended as an example of an application that could be used to bootstrap a rootkit (hence the name). Summed up as briefly as possible, RootStrap phones home periodically to fetch remote native ARM code and executes it outside the Dalvik VM. An attacker could use such an approach to gain a large install base for a seemingly innocent application and then push down a local privilege escalation exploit as soon as a new vulnerability is discovered in the Linux kernel and root the device. Since carriers are fairly conservative in pushing out OTA patches for their devices, an attacker could easily push out their malicious payload before the devices were patched.




In addition to the sample RootStrap application, I also posted an innocent looking app called "Twilight Eclipse Preview" that claimed to be a preview of the upcoming Twilight Eclipse movie to the Android Market. The Twilight app was actually just RootStrap in disguise, displaying a Twilight image while

- applicazione "innocente"
- pubblicata sul market
- "call home"
- scarica malicious payload
- lo esegue run-time



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 26
SOME RIGHTS RESERVED

# Trojan aftermarket firmware

(non ci sono casi pubblicamente conosciuti, AFAIK)

The Security Implications of Custom Android ROMs « Neohapsis Labs - Mozilla Firefox

File Edit View History Bookmarks Tools Help

labs.neohapsis.com/2011/12/21/the-security-implications-of-custom-android-roms/

The problem of software authenticity has been encountered before by the linux community, and over the last decade, that community has developed distribution methods such as centralized Debian APT repositories that provide some degree of certainty over what the end user is actually installing on their computer. Additionally, many linux users still download the source code for projects and compile them locally on their computer. Within the Android modding community, neither of these options have been implemented with the same level of maturity that linux has seen. The more popular aftermarket firmware, or "ROMs", such as CyanogenMod, are distributed by more accountable means, providing MD5 checksums for the files and a clear distribution network. However, often, Android customization software is provided through links to anonymous file-sharing sites such as Mediafire and Megaupload. This creates the opportunity to trick a user into installing malicious files.

**A Ripe Target**

There has already been at least one documented case of malware targeting custom Android ROMs, a trojan affected devices in the Chinese market. With the popularity of the Galaxy Nexus, and the continued interest in Android customization community, this could become an attack vector that is more and more appealing to malware authors. And the fact that majority of customized Android software is distributed without the usage of the Android Market, users do not have the additional means of protection that the Market provides, namely the "kill-switch" for apps that Google has flagged as malicious that were installed via the Android

bit.ly/NqzeaH--- 2 months ago

Follow @neohapsis

FACEBOOK

Neohapsis on Facebook

Like 15

Follow

**http://labs.neohapsis.com/2011/12/21/the-security-implications-of-custom-android-roms/**



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 27

SOME RIGHTS RESERVED

# Traffico di rete

**http://phys.org/news/2011-05-android-devices-susceptible-eavesdropping.html**

Researchers show Android devices susceptible to eavesdropping - Mozilla Firefox

File Edit View History Bookmarks Tools Help

phys.org/news/2011-05-android-devices-susceptible-eavesdropping.html

on Google applications.

The German research team, after seeing what Wallach had found, decided to look a little deeper; they found that because Android phones use tokens, called authTokens, that allow legitimate users to remain logged into certain Google applications for up to two weeks, that are unencrypted; nefarious characters listening in could capture those tokens and then use them for their own illegitimate purposes, such as scraping calendar information, contact email addresses or to view private images in Picasa.

In some respects, many might not see such a breach as all that big of a deal; it's not like Google is passing around bank account codes willy-nilly, but, that's beside the point. What's important is that Google, a huge company with vast resources and staffed with some of the best in the business, clearly knew and understood what it was doing when it chose to ... authTokens; a move th... community; something ... discovery that it has b...

*Use of HTTPS in Android Google Apps:*

| Android version | Calendar Sync | Contacts Sync | Picasa Sync (Gallery) |
|---|---|---|---|
| 3.0 | yes | yes | ? |
| 2.3.4 | yes | yes | no |
| 2.3.3 | no | no | no |
| 2.2.1 | no | no | n/a |
| 2.2 | no | no | n/a |
| 2.1 | no | no | n/a |

Enlarge

And while this issue will eventually go away as users upgrade the software on their phones, something else is rather important here, and that is the means by which this news has come to the fore, i.e. through a grad student taking an undergraduate course, basically just fooling around with sniffing software. This quite naturally begs the question of, what else is at risk? If there is no organization or agency testing the products that are sold by huge companies to users, how can we know that the things we do are safe from those who might wish to steal our data, impersonate us, or worse use the things they find against us, such as disseminating embarrassing pictures we thought were safely tucked away under password protection on Picasa?

© 2010 PhysOrg.com

Google adds voice and video to Google Talk on Android smartphones

Feb 11, 2011 0

Google account users get extra security

android, eavesdropping, picasa, google

- no HTTPS (ahi ahi ahi)
- MiTM
- Hot Spot
- Rogue APs



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 28

SOME RIGHTS RESERVED

# Decompilazione / reversing

## Batteries (almost) included, no assembly required

**http://code.google.com/p/apk-extractor/**

"is capable of parsing Android Manifest, XML layouts etc. and converting DEX/ODEX to CLASS, which can be opened by any de-compiler. "

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

Tools to work with android .dex and java .class files (read, convert, modify, deobfuscate, ..)

**http://code.google.com/p/smali/**

An assembler/disassembler for Android's dex format

**http://code.google.com/p/android-apktool/**

It is a tool for reverse engineering 3rd party, closed, binary Android apps. It can decode resources to nearly original form and rebuild them [..]

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

Yet another fast Java decompiler



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 29

SOME RIGHTS RESERVED

# .apk tools demo

## Batteries (almost) included, no assembly required

demo



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 

SOME RIGHTS RESERVED

# reversing, injections, ..

## (some) assembly required

**http://mulliner.org/android/feed/binaryinstrumentationandroid_mulliner_summercon12.pdf**

Binary Instrumentation on Android, Collin Mulliner

**http://www.slideshare.net/jserv/practice-of-android-reverse-engineering**

Practice of Android Reverse Engineering, Jim Huang

**http://code.google.com/p/androguard/**

Reverse engineering, Malware and goodware analysis of Android applications ...
and more (ninja !)

**https://redmine.honeynet.org/projects/are**

Virtual Machine for Android Reverse Engineering

**http://radare.org**

radare, the reverse engineering framework



© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it - Pagina 31

SOME RIGHTS RESERVED

# OWASP Top 10 Mobile Risks (RC1)

https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Top_Ten_Mobile_Risks
http://www.slideshare.net/JackMannino/owasp-top-10-mobile-risks

## Top 10 Mobile Risks, Release Candidate v1.0

1. Insecure Data Storage
2. Weak Server Side Controls
3. Insufficient Transport Layer Protection
4. Client Side Injection
5. Poor Authorization and Authentication
6. Improper Session Handling
7. Security Decisions Via Untrusted Inputs
8. Side Channel Data Leakage
9. Broken Cryptography
10. Sensitive Information Disclosure


© Igor Falcomatà <ifalcomata@enforcer.it>, alcuni diritti riservati: http://creativecommons.org/licenses/by-sa/2.0/it/deed.it 
SOME RIGHTS RESERVED

# (Domande?)

# do android malware writers dream of electric sheep?

**Linux Day 2012**
**Firenze**

free advertising >

SOME RIGHTS RESERVED
http://creativecommons.org/licenses/by-sa/2.0/it/deed.it