Hacking your Droid

ANDROID MALWARES

Aditya Gupta

Facebook[dot]com/aditya1391

@adi1391

./whoami

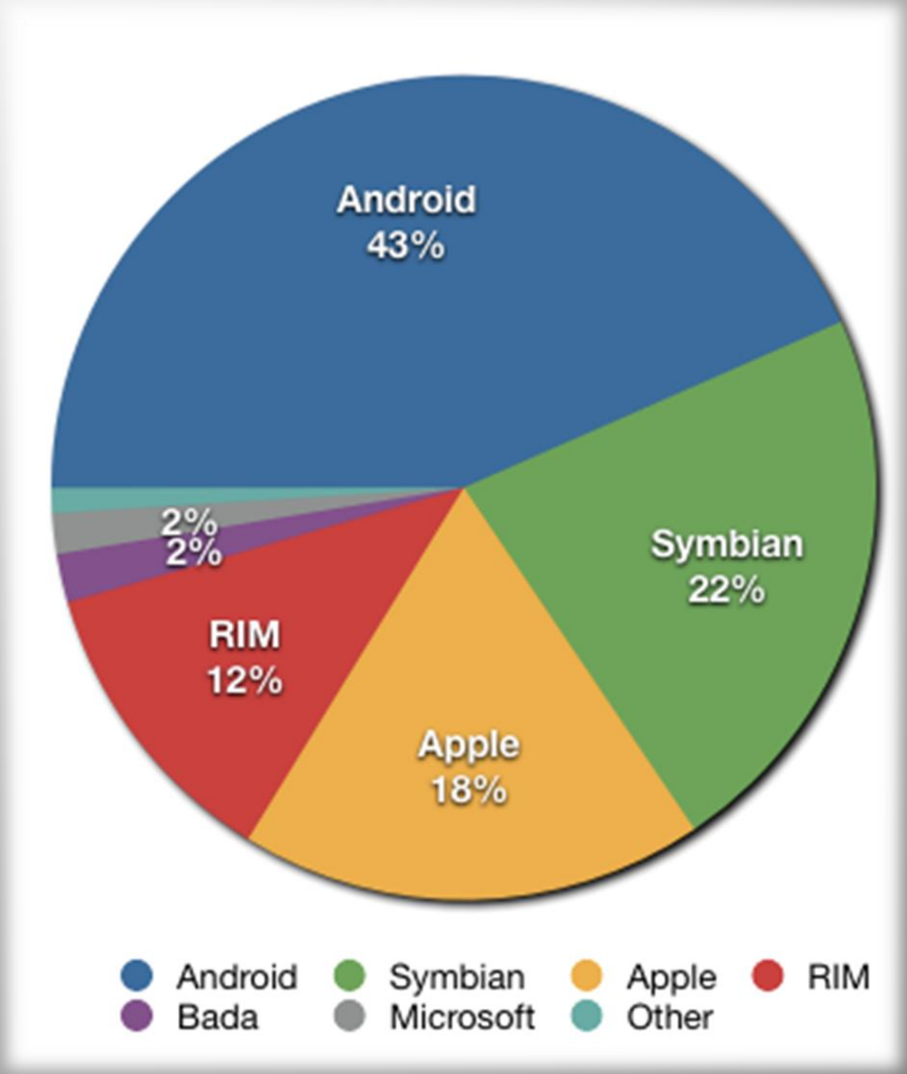
- College Student
- Security Researcher
- NOT an expert
- Grey Hat

Agenda

- Android OS Basics
- Inside the APK
- Android Security Model
- Reversing the codes
- Some case studies
- Making our own malware
- Malware = Money
- Mobile App Pentesting

What is Android

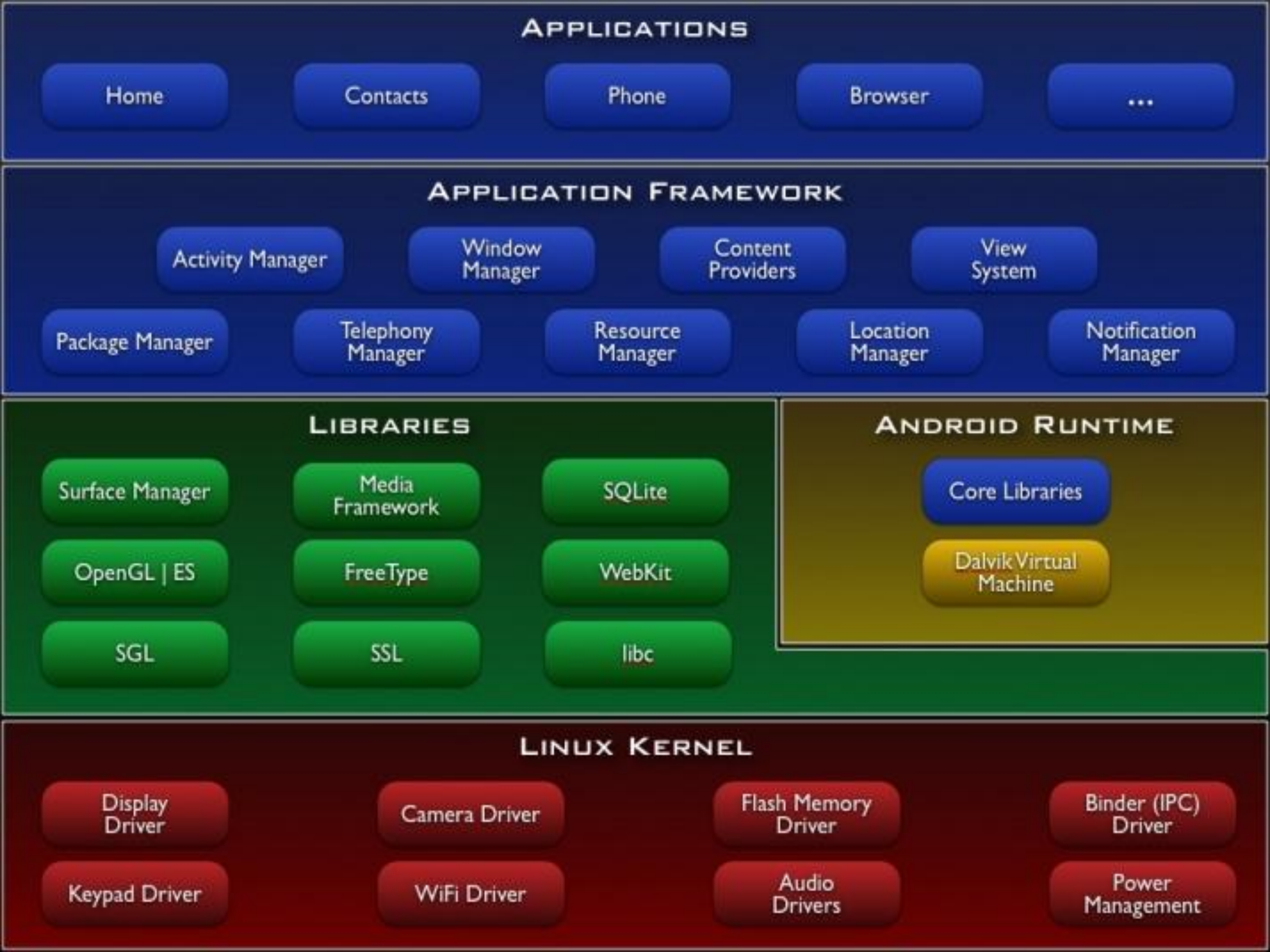
- Software Stack including OS, middleware and applications
- Developed by Google and OHA(Open Handset Alliance)
- Largest Market Share, more than Symbian and IOS.



Why Android

- Everywhere! (TV, phones, tablets)
- Easy to expl0it + Open Source
- Runs on Linux 2.6.x kernel
- Uses SQLite database
- Huge community base
- Official market containing over 4,00,000 apps

ANDROID ARCHITECTURE



Applications

- Home
- Contacts
- Phone
- Browser
- ...

Application Framework

- Activity Manager
- Window Manager
- Content Providers
- View System
- Package Manager
- Telephony Manager
- Resource Manager
- Location Manager
- Notification Manager

Libraries

- Surface Manager
- Media Framework
- SQLite
- OpenGL | ES
- FreeType
- WebKit
- SGL
- SSL
- libc

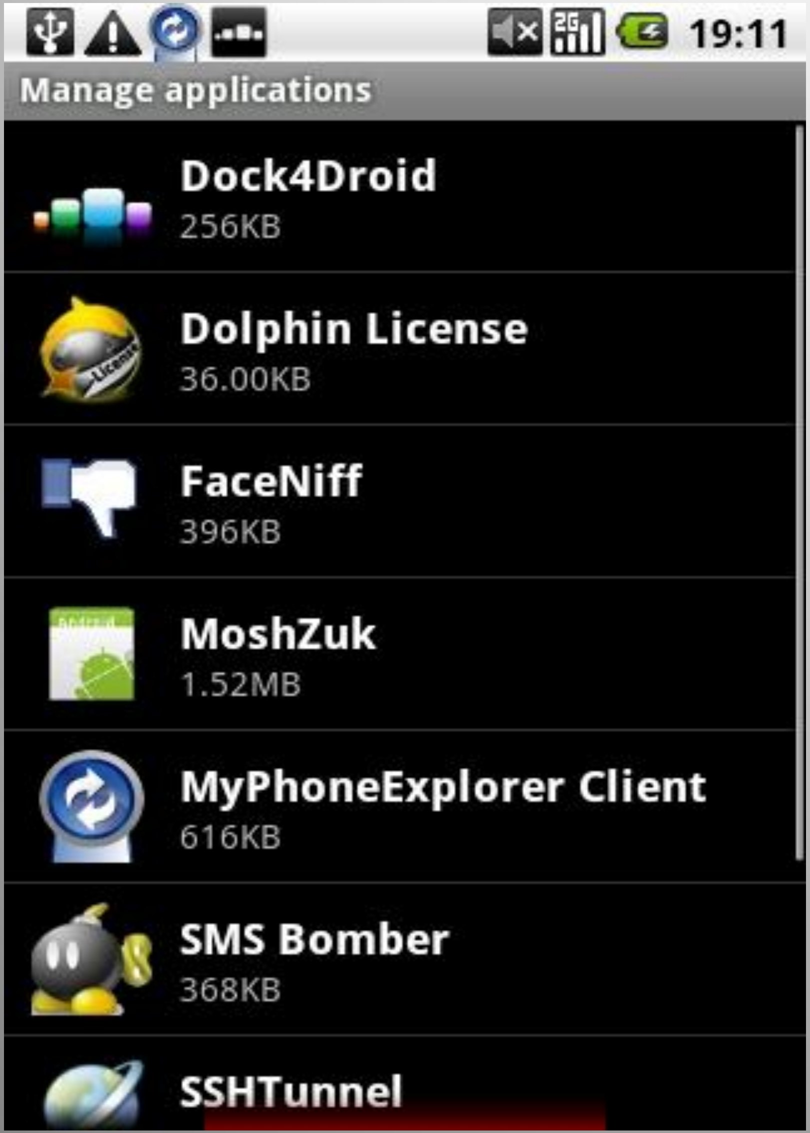
Android Runtime

- Core Libraries
- Dalvik Virtual Machine

Linux Kernel

- Display Driver
- Camera Driver
- Flash Memory Driver
- Binder (IPC) Driver
- Keypad Driver
- WiFi Driver
- Audio Drivers
- Power Management

Android Applications



Manage applications

Dock4Droid
256KB

Dolphin License
36.00KB

FaceNiff
396KB

MoshZuk
1.52MB

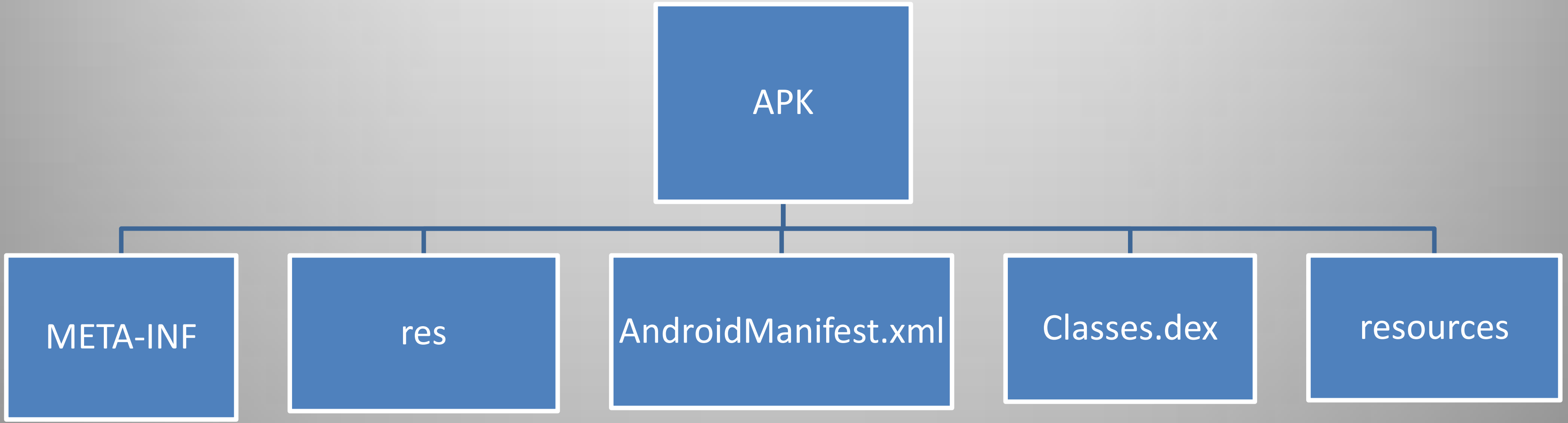
MyPhoneExplorer Client
616KB

SMS Bomber
368KB

SSHTunnel

- .apk (Android Package) format
- Nothing more than a zip file.
- Written exclusively in Java, with native libraries in C/C++.
- Composed of components such as Activities, Services, Broadcast Recievers, etc.

Android Applications



APK

META-INF

res

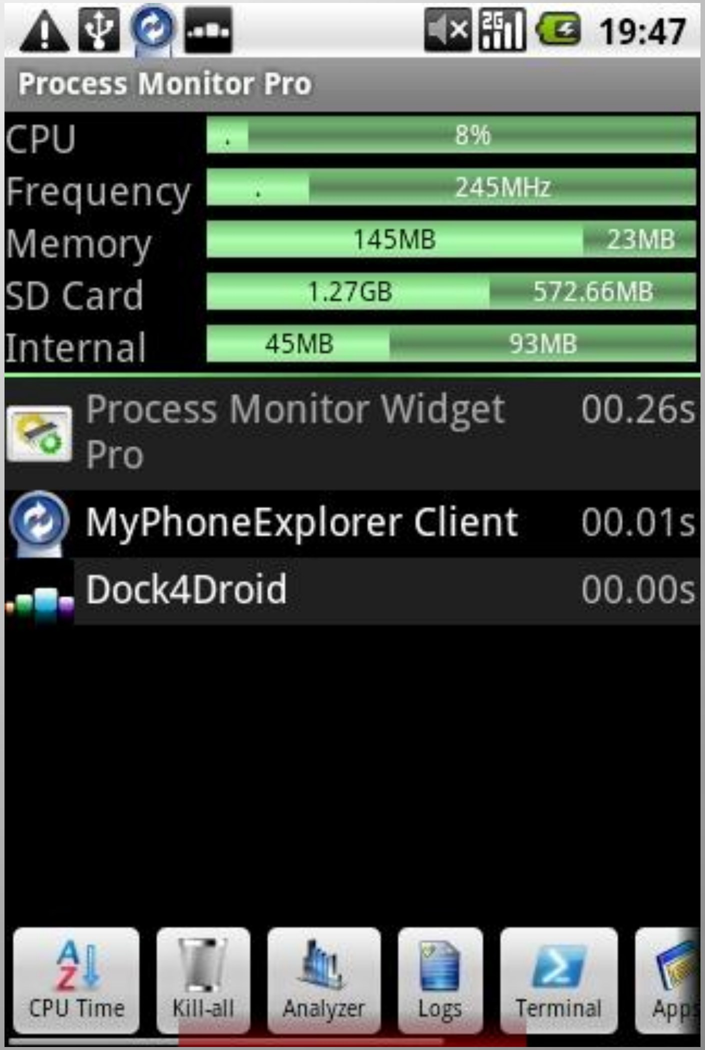
AndroidManifest.xml

Classes.dex

resources

ACTIVITY

- Screen to let users interact
- Consists of views (Buttons, TextView, ImageView, Table view, List view etc)
- “main” activity presented on start
- Lifecycle is “LIFO”



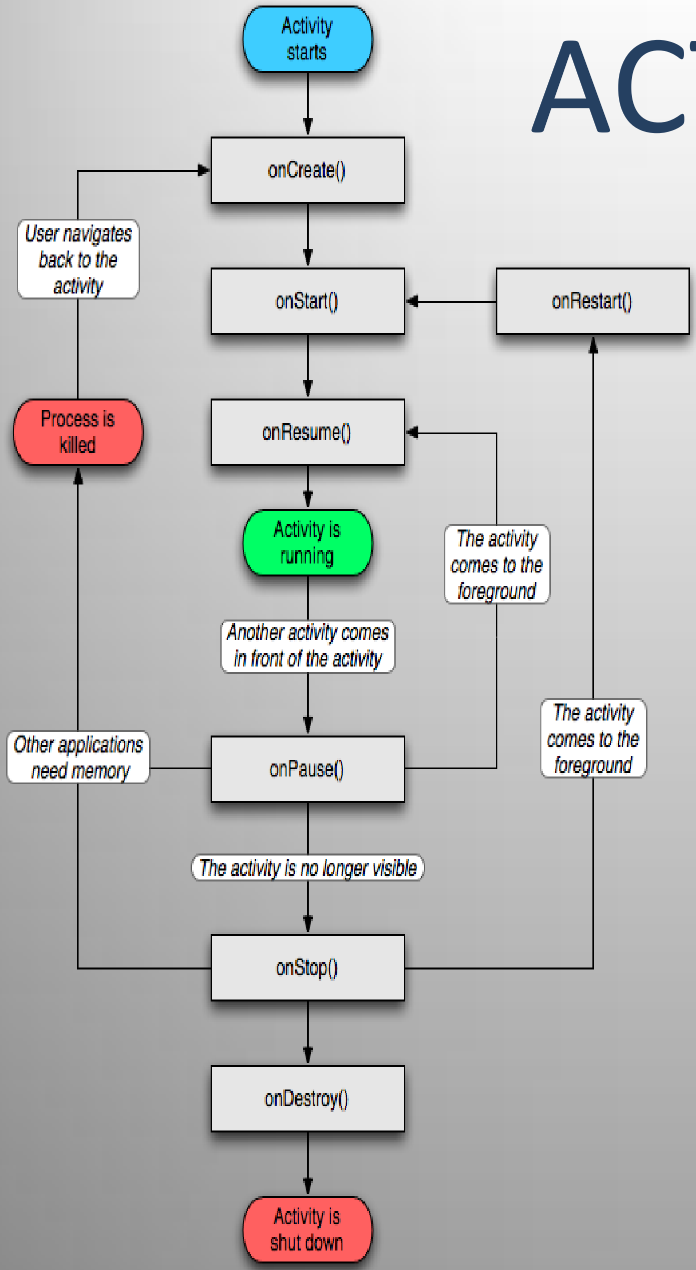
19:47

Process Monitor Pro

CPU	8%	
Frequency	245MHz	
Memory	145MB	23MB
SD Card	1.27GB	572.66MB
Internal	45MB	93MB

Process Monitor Widget Pro	00.26s
MyPhoneExplorer Client	00.01s
Dock4Droid	00.00s

ACTIVITY


Activity starts
onCreate()
User navigates back to the activity
onStart()
onRestart()
Process is killed
onResume()
Activity is running
The activity comes to the foreground
Another activity comes in front of the activity
Other applications need memory
onPause()
The activity comes to the foreground
The activity is no longer visible
onStop()
onDestroy()
Activity is shut down

- Follows the Activity Lifecycle.
- Activity of one application can be accessed by other application*.
- Permission has to be granted

SERVICE

- Performs the work in the background
- Doesn’t comes with a UI
- Can be either stated or bound(or both)
- Example – playing music in the bg, network activities, file i/o operations etc.

Other Components

- Broadcast Reciever

  receives and responds to broadcast announcements

  Incoming SMS , Screen Off etc.

- Intents

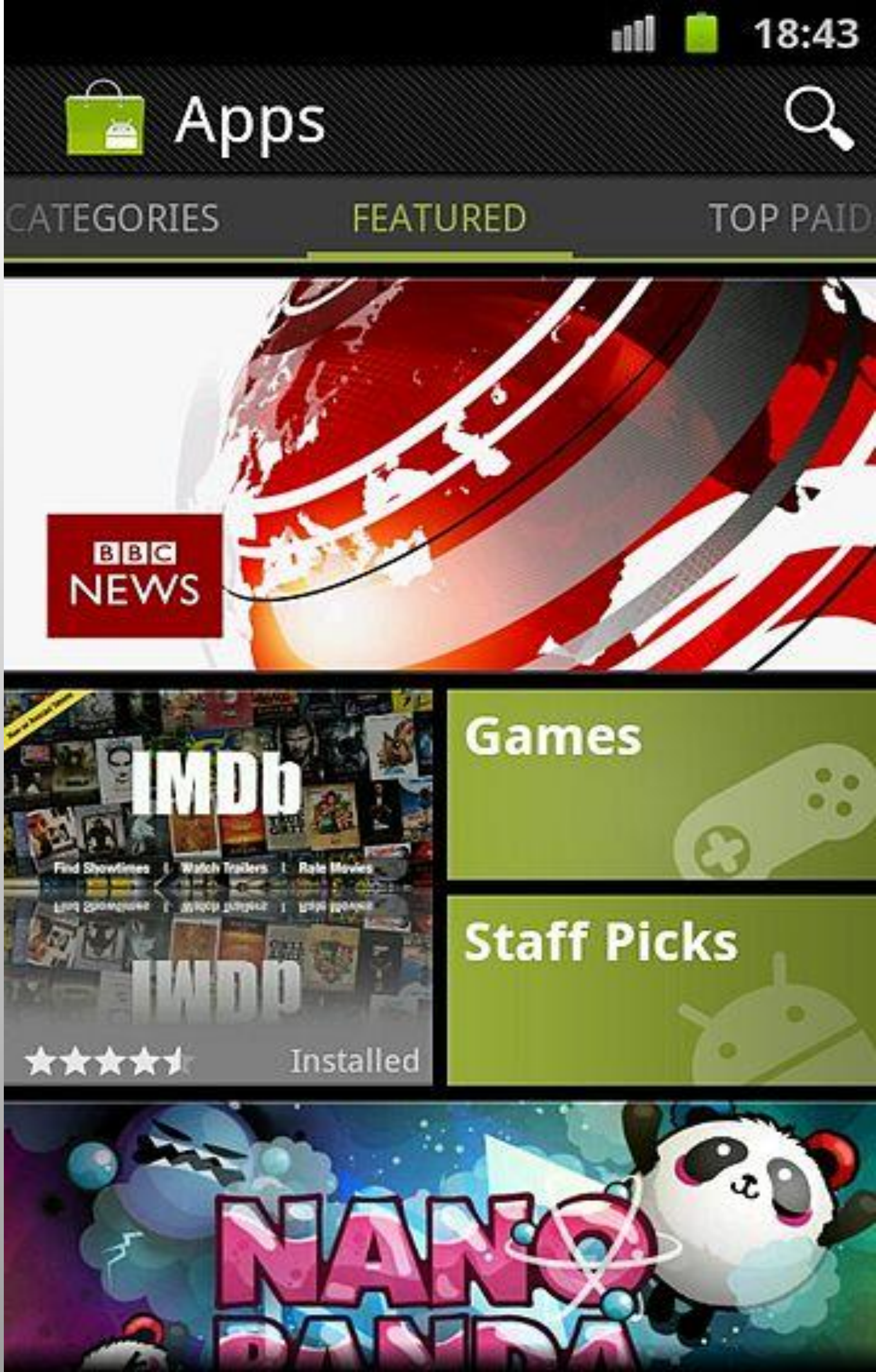
  Binds individual components at runtime

- Content Providers

  Stores and retrieves the application data

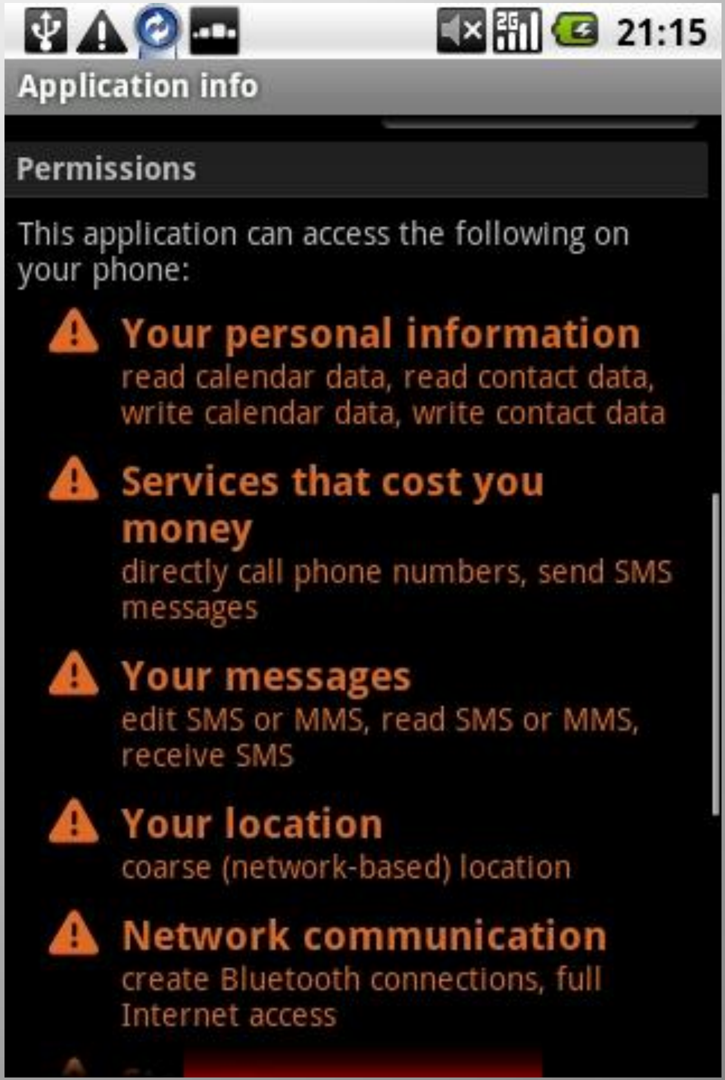
  Data stored in an SQLite database

- Preinstalled on all Android devices
- Contains over 4.5 billion apps
- Anyone can publish his/her app



Permissions.. WTF?

- Declared in AndroidManifest.xml
- XML file containing all the components and permissions
- Can only use the declared permissions



Permissions

This application can access the following on your phone:

Your personal information
read calendar data, read contact data, write calendar data, write contact data

Services that cost you money
directly call phone numbers, send SMS messages

Your messages
edit SMS or MMS, read SMS or MMS, receive SMS

Your location
coarse (network-based) location

Network communication
create Bluetooth connections, full Internet access

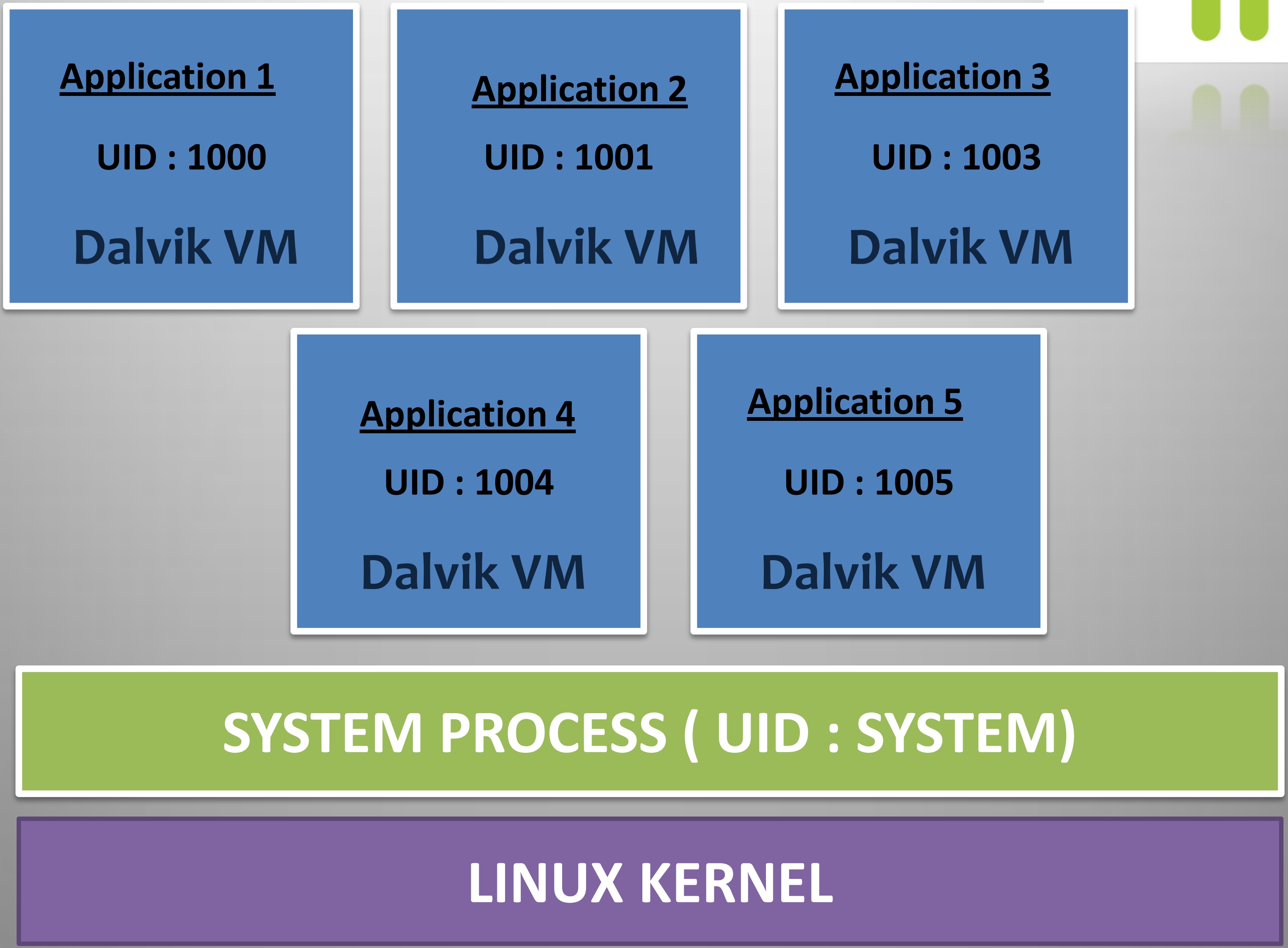
Permissions.. WTF?

- ACCESS_COARSE_LOCATION
- ACCESS_FINE_LOCATION
- BRICK
- CALL_PHONE
- INTERNET
- GET_ACCOUNTS
- PROCESS_OUTGOING_CALLS
- READ_OWNER_DATA
- READ_SMS
- RECEIVE_SMS
- SEND_SMS
- USE_CREDENTIALS
- WRITE_OWNER_DATA
- RECORD_AUDIO

Android Security Model

- Each application is run within a Dalvik Virtual Machine
- With unique UID:GID
- By default no permission is granted
- Permissions required by an application have to be approved by the user.
- Apk files must be signed with a certificate.

Android Security Model



Application 1

UID : 1000

Dalvik VM

Application 2

UID : 1001

Dalvik VM

Application 3

UID : 1003

Dalvik VM

Application 4

UID : 1004

Dalvik VM

Application 5

UID : 1005

Dalvik VM

SYSTEM PROCESS (UID : SYSTEM)

LINUX KERNEL

DALVIK VIRTUAL MACHINE(DVM)

Created by Dan Bornstein

DVM vs JVM

Virtual System to run the android apps

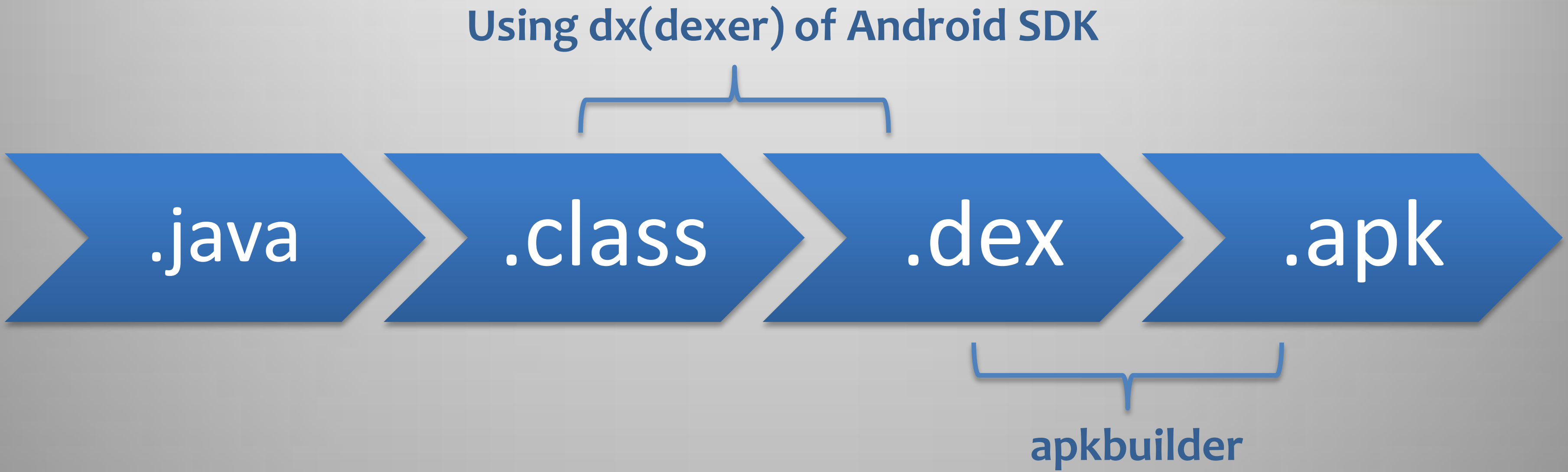
Register based instead of stack based

Runs the dex(Dalvik Executable) files

REVERSE ENGINEERING

BREAKING THE CODES

Making of the APK



Using dx(dexer) of Android SDK

.java

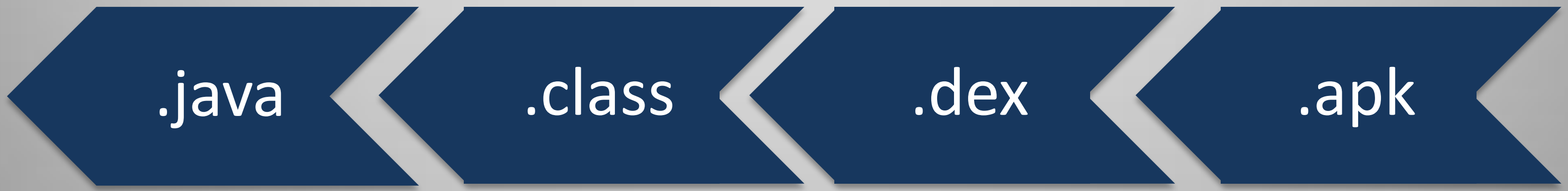
.class

.dex

.apk

apkbuilder

REVERSING THE APK



.java

.class

.dex

.apk

REVERSING THE APK

Tools of the trade

Dedexer

Baksmali

Undx

JD-GUI

Dex2JAR

DexDump

APKTool

GETTING OUR HANDS DIRTY

DEMO TIME

ANDROID MALWARES

Special thanks to **Mila** for his awesome website

http://contagiodump.blogspot.com

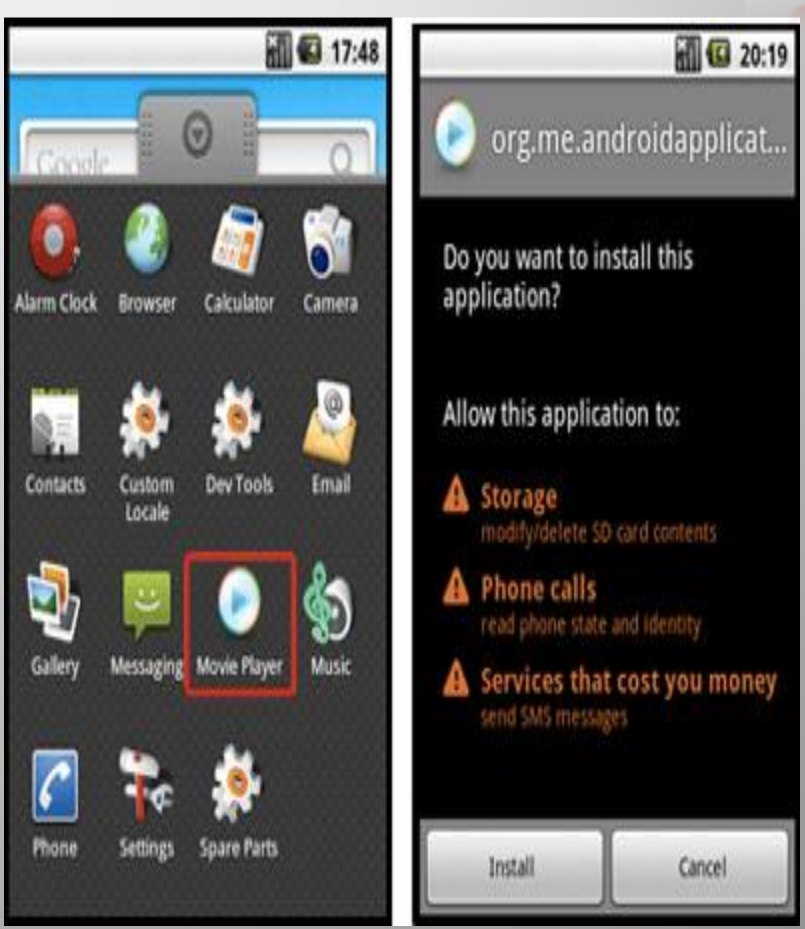
Memories of the Past

Some famous Android Malwares

- Trojan-SMS.AndroidOS.FakePlayer.a
- Geinimi
- Snake
- DreamDroid
- GGTracker

Trojan-SMS.FakePlayer.a

- Simplest malware till date.
- Sends SMS to a premium rated number
- $6-10/sms
- Mainly distributed through porn/media apps
- Stop watching porn? :O



org.me.androidapplicat...

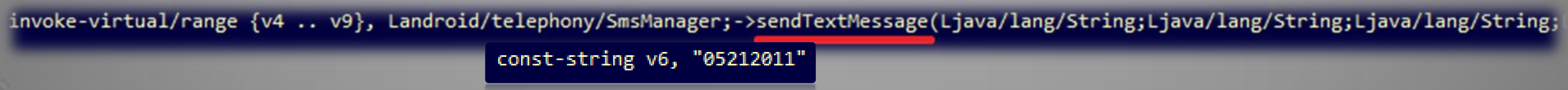
Do you want to install this application?

Allow this application to:

Storage
modify/delete SD card contents

Phone calls
read phone state and identity

Services that cost you money
send SMS messages



```
invoke-virtual/range {v4 .. v9}, Landroid/telephony/SmsManager;->sendTextMessage(Ljava/lang/String;Ljava/lang/String;Ljava/lang/String;
const-string v6, "05212011"
```

GEINIMI : THE HOTTEST MALWARE

THE LOOKOUT BLOG

DECEMBER 29, 2010

Security Alert: Geinimi, Sophisticated New Android Trojan Found in Wild

Geinimi Android malware has 'botnet-like' capabilities

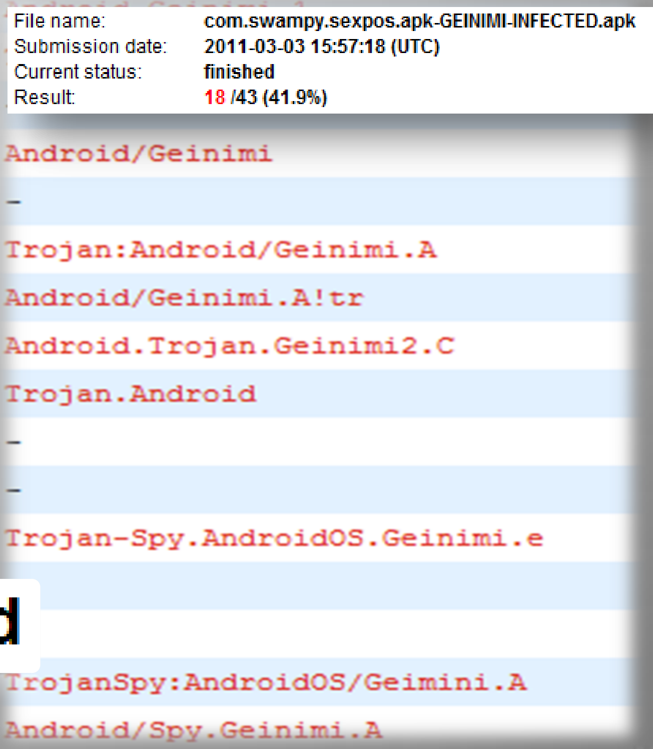
Android Trojan dubbed 'Geinimi' found in legitimate applications

December 30, 2010 by John Messina • report

SOPHOS

Troj/Geinimi-A

Category:	Viruses and Spyware
Type:	Trojan



File name:	com.swampy.sexpos.apk-GEINIMI-INFECTED.apk
Submission date:	2011-03-03 15:57:18 (UTC)
Current status:	finished
Result:	18 /43 (41.9%)

```
Android/Geinimi
-
Trojan:Android/Geinimi.A
Android/Geinimi.A!tr
Android.Trojan.Geinimi2.C
Trojan.Android
-
-
Trojan-Spy.AndroidOS.Geinimi.e


TrojanSpy:AndroidOS/Geimini.A
Android/Spy.Geinimi.A
```

nakedsecurity

IT Security Blog of the Year

Geinimi Android Trojan horse discovered

GEINIMI

- Most sophisticated malware till date.
- Botnet like capabilities
- Multiple variants created on the same device
- Obfuscated code
- Strings decrypted at runtime
- All network data encrypted (DES with a key - 012345678)

GEINIMI

- Three ways of starting (Using service or Broadcast Receivers
- Makes a connection with C&C server
- Identifies each device with unique IMEI & IMSI
- Can be in 5 states (Start, download, parse, transact, idle)
- Info Stealer
- Infected legitimate apps (Sex Positions, MonkeyJump2 etc.)

(Another reason for not watching porn on mobile!)

GEINIMI(continued)

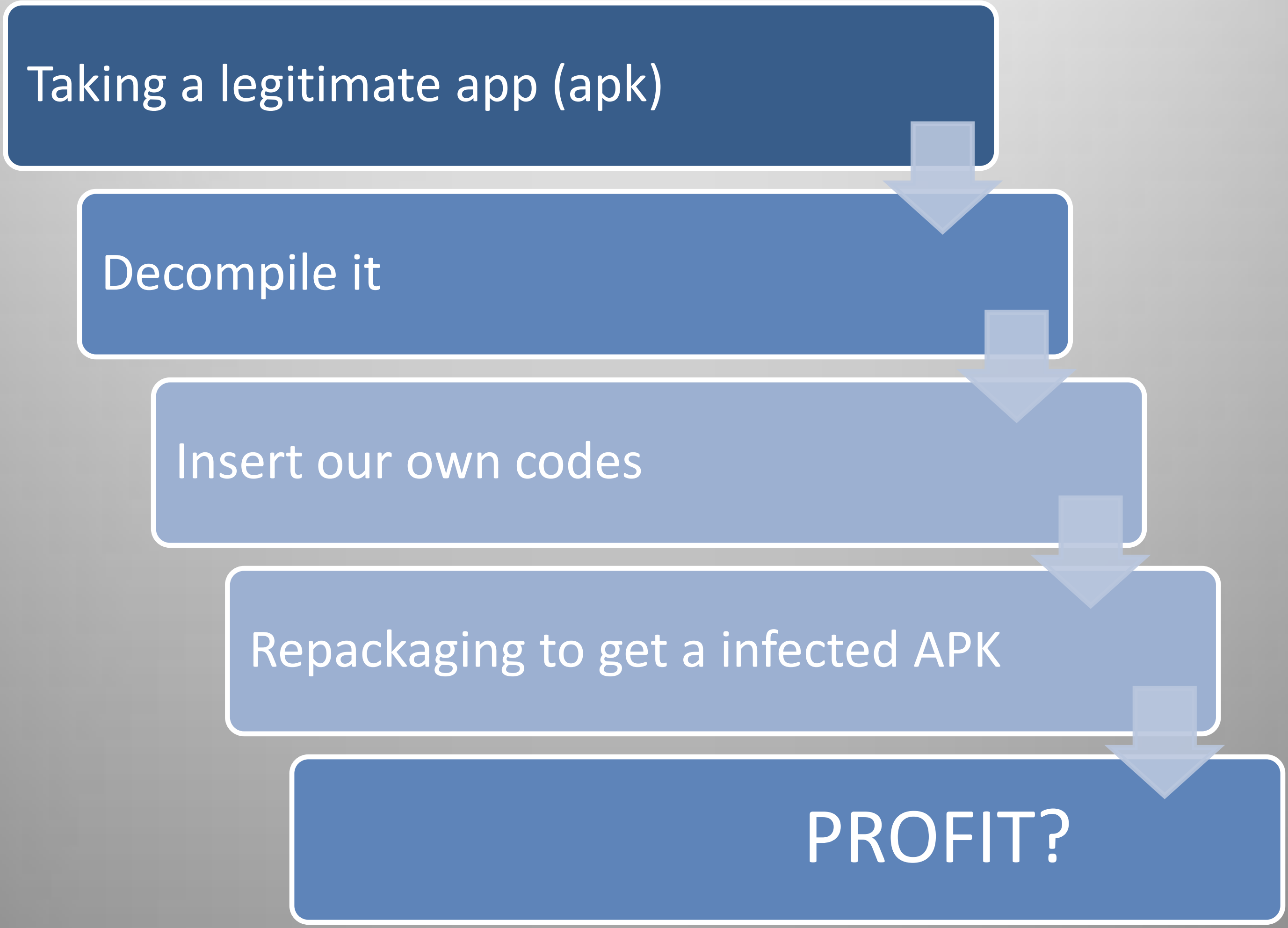
- **Botnet Command Capabilities :**
  - call – Call a number
  - Email – Send a email
  - Smsrecord – Sends all the sms’es to the server
  - Install – install an app
  - Shell – get a shell
  - Contactlist - get the contact list of the victim
  - Wallpaper – change the wallpaper etc.

DREAMDROID

- Infected legitimate software
- Hosted at “Android Market”
- Came with exploits namely Exploid (CVE-2009-1185) and rageagainstthecage(CVE-2010-EASY)
- Multi Staged Payload
- XOR Encrypted data
- Another malware with Botnet capabilities

Creating our own Android Malware

Agenda



Taking a legitimate app (apk)

Decompile it

Insert our own codes

Repackaging to get a infected APK

PROFIT?

CREATING A MALWARE

Expected Time to be taken < 5 mins

Vulnerable Applications

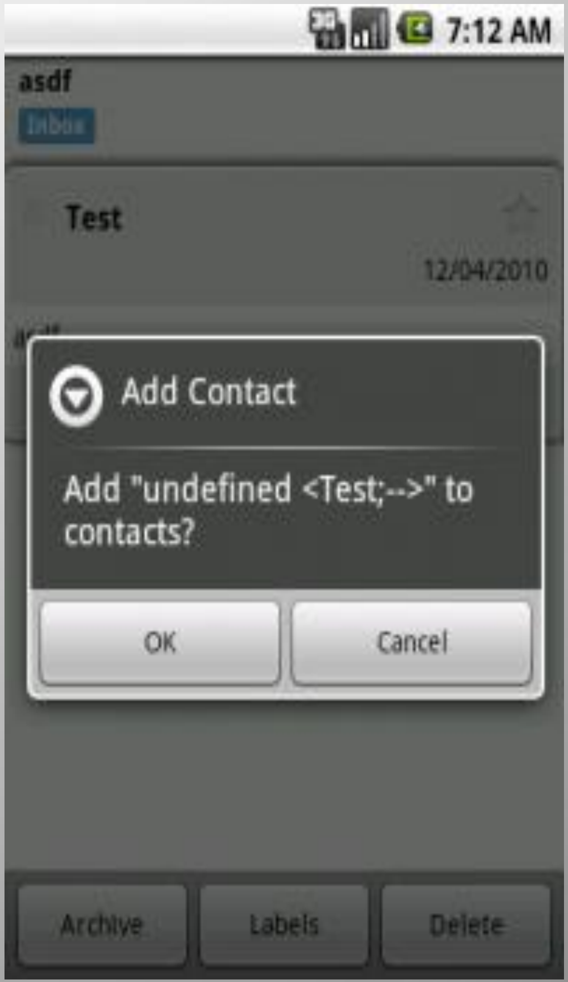
- GMail App(in <Android v2.1 vuln to XSS :O

  From field: “ onload=window.location=‘http://google.com’ “@yahoo.com”

  (Found by supernothing of spareclockcycles.org)

- Use this to launch more exploits such as the Data Stealing Bug or Free Webkit Exploit

- Steal Emails & SD Card Files



Add Contact

Add "undefined <Test;-->" to contacts?

Stored Passwords

- Browser passwords stored in database called **webview.db**
- Got r00t?

```
#adb pull /data/data/com.android.browser/databases/webview.db
#sqlite webview.db
 > SELECT * FROM password;
```

Insecure Data Storage

```
# cd /data/data/com.evernote
# ls
cache
databases
shared_prefs
lib
# cd shared_prefs
# ls
com.evernote_preferences.xml
# cat com.evernote_preferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
<string name="serviceHost"><string
name="username">myusername</string>
<boolean name="ACCOUNT_CHECKED" value="true" />
<string name="password">youcanthackme</string>
<int name="servicePort" value="0" />
<boolean name="NotifyUploadStatus" value="true" />
</map>
#
```

Is that all?

Webkit and platform vulnerabilities

Android 2.0 ,2.1, 2.1.1 WebKit Use-After-Free Exploit

Android 2.0/2.1 Use-After-Free Remote Code Execution on Webkit
Vulnerabilities in Apps, SQLi, XSS, etc.

Use platform vulns to get root & shell

SD card information leakage

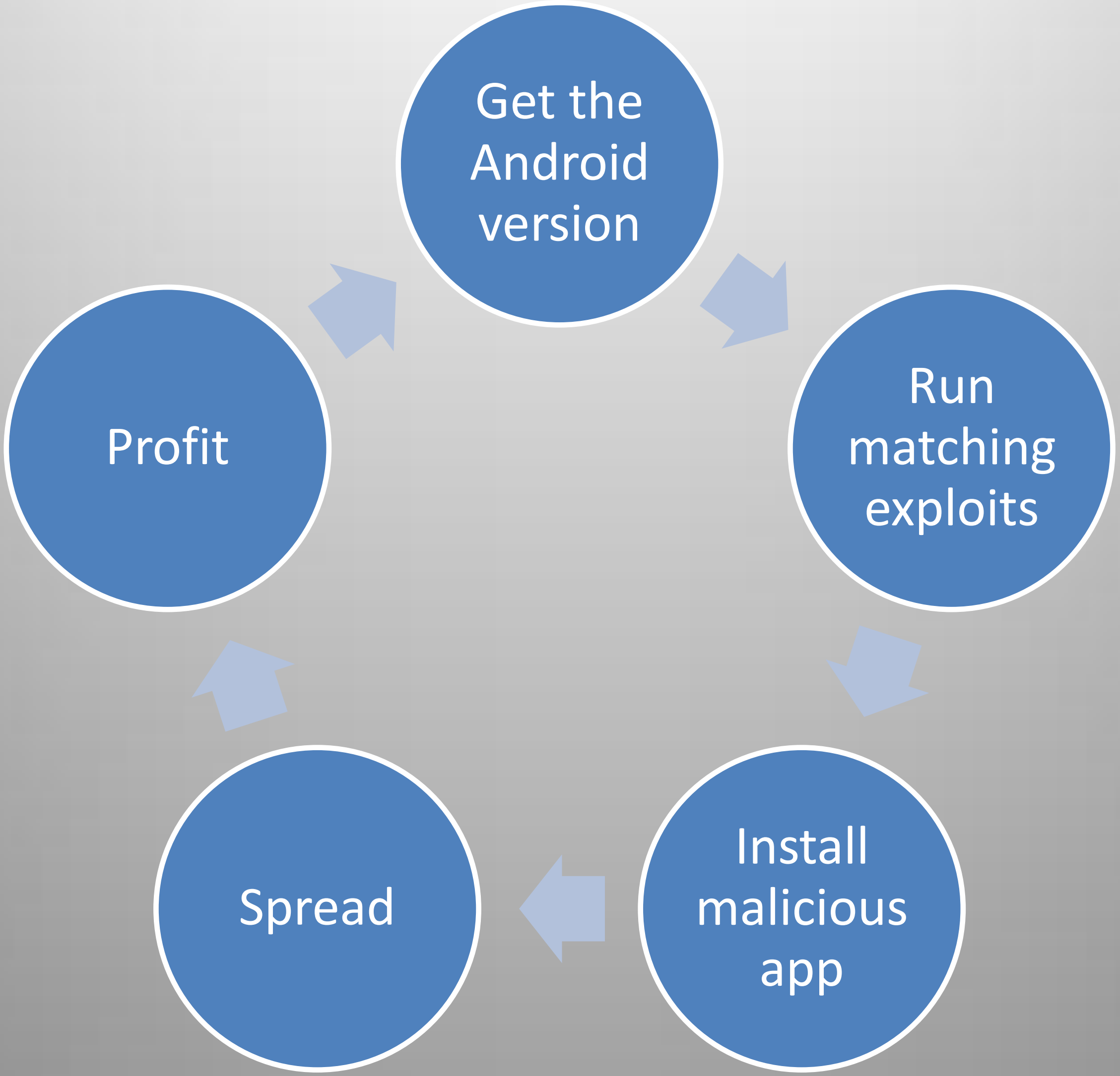
XSSF Framework

ROOTSTRAP

Sniffing the network :)

Try MoshZuk & ANTI

Is that all?



Get the Android version

Run matching exploits

Install malicious app

Spread

Profit

[$]Where is the money?[$]

[$$$]100% Illegal Ways to get rich! [$$$]

- Mobile App moolah by Jimmy Shah

Your phone has been hacked!
Transfer $1000 to my account
Or else…….
Acc No : xxxxxxxxxxxxxxxxxxxx

- Premium Rates SMSes
- Make malwares for sale
- Click Fraud, BlackHat SEO, Traffic generation, PPC Ads
- Steal Accounts/CCs and sell them
- Get personal information and blackmail the owner
- Sign up to many services with your referral id
- Make a bank phishing app

[$$]Spread Yourself![$$]

- Forums
- P2P
- Send SMS’es/chat with your download link from the infected user’s phone
- Make a blog of cracked full version of famous android apps!
- Social Network viral scripts
- Android Market
- Amazon App Store

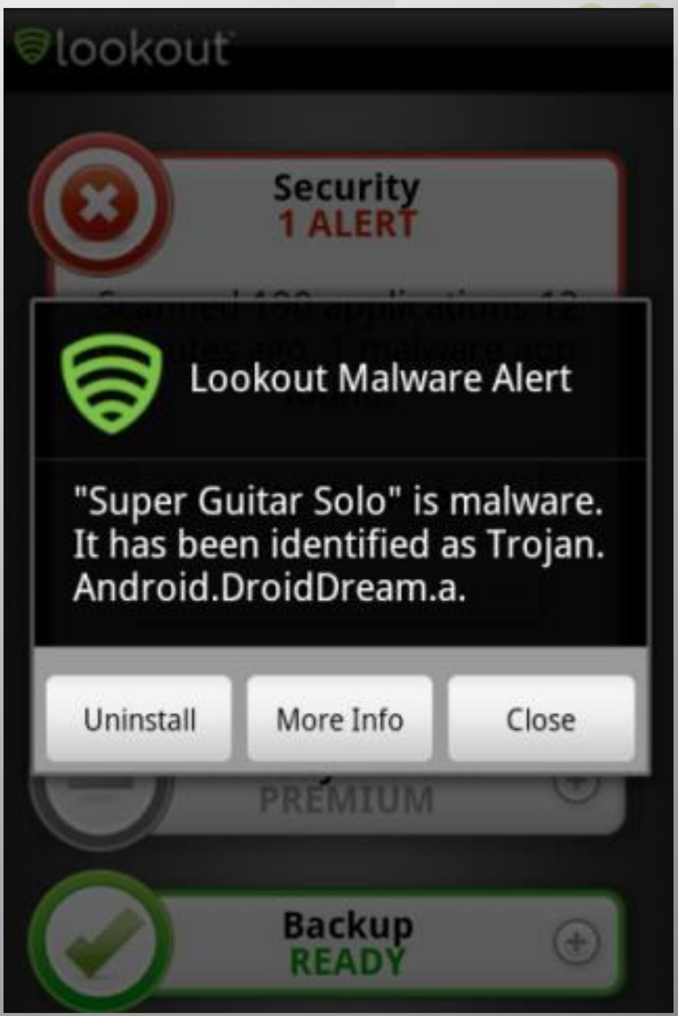
Outlaws vs Angels

The game is over!

- Malware scanners developed for this platform.

- Lookout(one of the best security solutions), AVG, Quick Heal, Kaspersky have come up with their security solutions.

- Can detect most of the malwares of this platform.



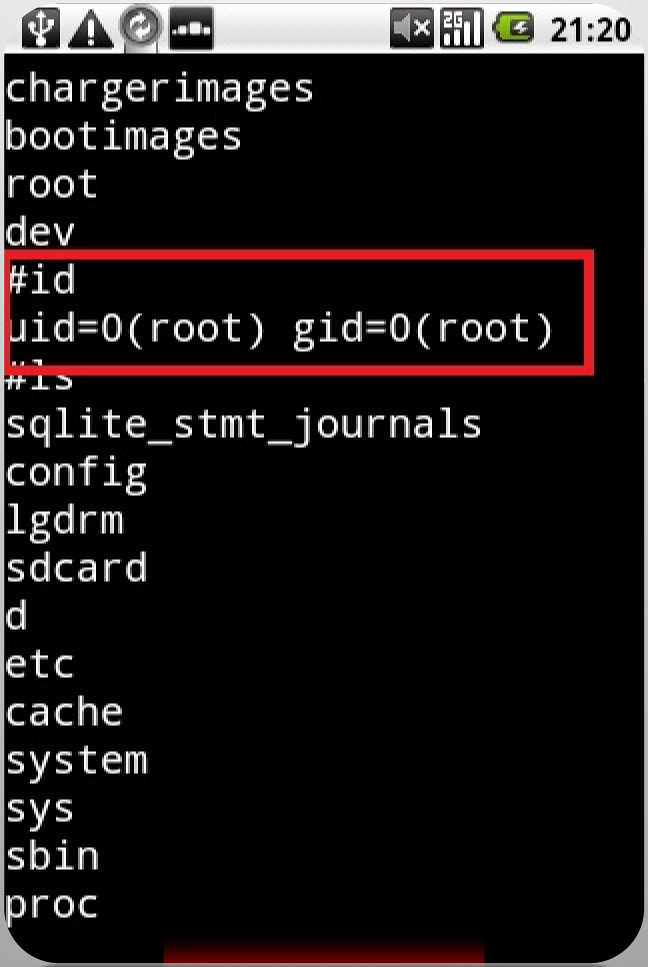
Lookout Malware Alert

"Super Guitar Solo" is malware. It has been identified as Trojan. Android.DroidDream.a.

~~The game is over!~~
The game is not over yet!

- Can create a malware not detected by the scanners
- Most of them signature based, so, can easily be bypassed.
- Obfuscating code can bypass most of them.
- Disable the AV
- Encryption for network data.
- Use your own “blackhat” creativity!

MobileApp Pentesting FTW!



```
chargerimages
bootimages
root
dev
#id
uid=0(root) gid=0(root)
#ls
sqlite_stmt_journals
config
lgdrm
sdcard
d
etc
cache
system
sys
sbin
proc
```

MobileApp Pentesting FTW!

- Decompile the apk after pulling it from the phone.

```
 adb pull /data/app(or app-private)/hello.apk
 unzip hello.apk
 dex2jar classes.dex
 jdgui classes2jar.jar
```

or convert to smali and then analyse the code

```
 adb pull /data/app/hello.apk
 unzip hello.apk
java –jar baksmali.jar –o C:\pentest\app classes.dex

            OR

 apktool d hello.apk
```

MobileApp Pentesting FTW!

- Start Emulator with Proxy

```
Emulator -avd MYAVD -http-proxy http://127.0.0.1:5001
```

- Install the app in the emulator

```
avd install apptotest.apk
```

- Use Wireshark, Fiddler & Burp Suite to monitor traffic
- Run the app and check logcat
- WhisperMonitor – Android App to monitor outgoing traffic

MobileApp Pentesting FTW!

Check the security mechanism and encryption used in a banking or payment app for network data

Manifest Explorer

Strace for debugging system calls and signals

Check the location where the app stores the login credentials.

THANK YOU!