ILLINOIS SECURITY LAB

Android OS Exploits

by Soteris Demetriou & Nikhil Tayal

Outline

ILLINOIS SECURITY LAB

- Motivation
- Background
- Malware & Vulnerabilities
- Root Exploits
- GingerBreak
- Approach
- Design - Implementation
- Demonstration
- References
- Conclusions

ILLINOIS SECURITY LAB

Motivation

My smartphone and me

ILLINOIS SECURITY LAB

Daily News

Smartphones rule half the U.S. market

CHRISTINE ROBERTS

Wednesday, August 15, 2012

Regular cell phones may be on the way out.

For the first time since they hit the market, smartphones are now in the pockets of more than half of the Americans who use mobile phones, according to a report published by the consulting firm, Chetan Sharma Consulting.

A Nielsen study in May boasted similar results, finding that 50.4 percent of U. smartphones - a c

However, Sharma' they come from o reported. Nielsen'

Join SCMA
Member Benefits
Sonoma Medicine
Physician Directory
SCMA News Briefs
Physician Resources

Forbes · New Posts · Most Popular · Lists

52 Share 228 Tweet 126 Share 0 Submit 7

Gene Marks, Contributor

9 Smartphone Apps Every Salesperson Must Have

If you are a salesperson then you will have a smartphone. Not a Blackberry because no one in this country uses Blackberrys any more. You will have an iPhone or a phone that uses the Android operating system. This may not be the case in just a few years when (in my opinion) tablets, like the iPad, Nexus

LOCAL FRONTIERS

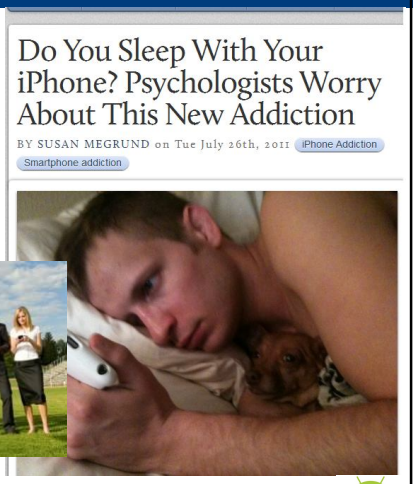
How Smartphones Are Transforming the Practice of Medicine

Rachel Friedman, MD

Ease, efficiency, efficacy—do these words describe your daily clinical practice? As more physicians feel the burden of information overload and struggle to avoid burnout in an increasingly frustrating healthcare system, the technology has finally arrived to make (almost) everything better: smartphones. After years of unsuccessful attempts to create and universaliz the nation's electronic health records (EHRs), smartphones and other mobile devices are now rapidly changing the way physicians and patients experience healthcare.

My smartphone and me

ILLINOIS SECURITY LAB

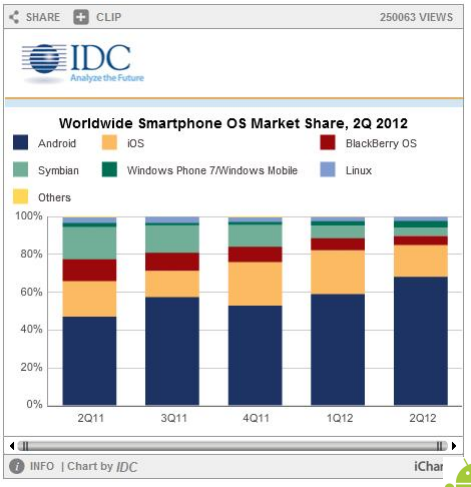


My smartphone and me

ILLINOIS SECURITY LAB

Why Android

ILLINOIS SECURITY LAB



FRAMINGHAM, Mass. August 8, 2012

Why Android

ILLINOIS SECURITY LAB

Top Smartphone Operating Systems, Shipments, and Market Share, Q2 2012 (Units in Millions)

Operating System	Q2 2012 Shipments	Q2 2012 Market Share	Q2 2011 Shipments	Q2 2011 Market Share	Year-over-year Change
Android	104.8	68.1%	50.8	46.9%	106.5%
iOS	26.0	16.9%	20.4	18.8%	27.5%
BlackBerry OS	7.4	4.8%	12.5	11.5%	-40.9%
Symbian	6.8	4.4%	18.3	16.9%	-62.9%
Windows Phone 7 / Windows Mobile	5.4	3.5%	2.5	2.3%	115.3%
Linux	3.5	2.3%	3.3	3.0%	6.3%
Others	0.1	0.1%	0.6	0.5%	-80.0%
Grand Total	154.0	100.0%	108.3	100.0%	42.2%

Source: IDC Worldwide Mobile Phone Tracker, August 8, 2012

FRAMINGHAM, Mass. August 8, 2012

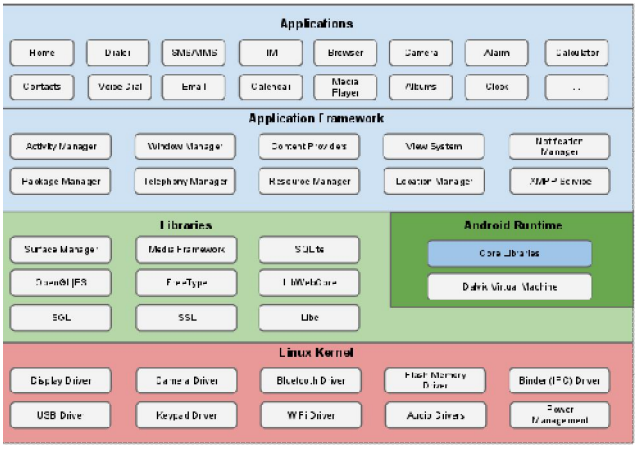
ILLINOIS SECURITY LAB

Background

Is it safe?

ILLINOIS SECURITY LAB

- Android Platform Security Architecture



Is it safe?

ILLINOIS SECURITY LAB

- System and kernel Level Security
  - Linux based
- Application Security
  - Permissions

Is it safe?

ILLINOIS SECURITY LAB

- Kernel
  - User ID per application

- Application Level
  - Permissions

Is it safe?

ILLINOIS SECURITY LAB

- Kernel
  - Linux Security
  - App Sandbox
  - System Partition & Safe mode
  - Filesystem Permissions
  - Filesystem Encryption
  - Password Protection
  - Device Administration
  - Memory Management Security Enhancements
  - Rooting of Devices

Is it safe?

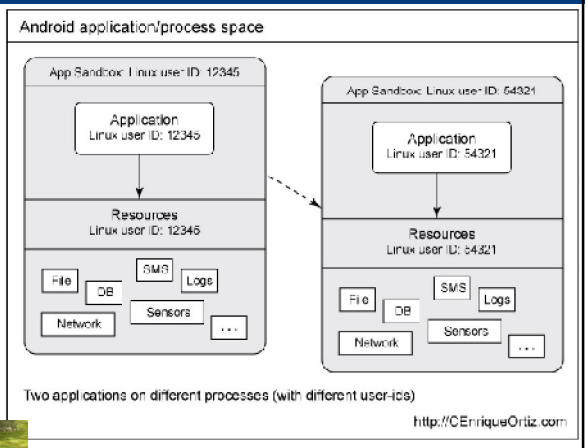
ILLINOIS SECURITY LAB

- Kernel (1/9)
  - Linux Security
    - User-based permission model
    - Process Isolation
    - Ability to modify the kernel

Is it safe?

ILLINOIS SECURITY LAB

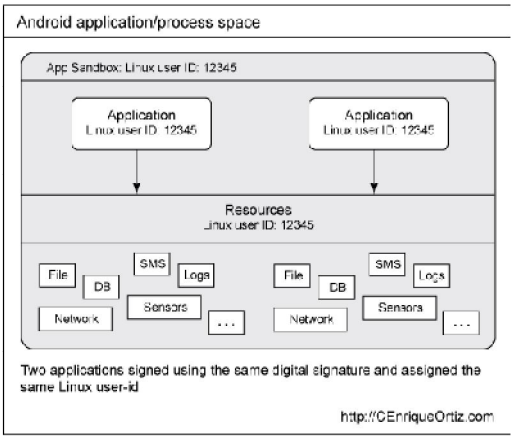
- Kernel (2/9)
  - Application Sandbox
    - Unique **UID** and **GID** per app on install
    - a Linux Process per app



Is it safe?

ILLINOIS SECURITY LAB

- Kernel (2/9) cnt'd
  - Application Sandbox cnt'd
    - (`android:sharedUserId`)



Is it safe?

ILLINOIS SECURITY LAB

- Kernel (3/9)
  - System Partition
    - Android Kernel
    - OS libraries
    - Application runtime
    - Application Framework
    - Applications
  - Safe mode
    - only core applications

Is it safe?

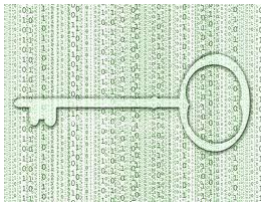
ILLINOIS SECURITY LAB

- Kernel (4/9)
  - Filesystem Permissions
    - Ensure that User A cannot alter or read User's B files
    - Application = User

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (5/9)
  - Filesystem Encryption
    - >= 3.0
    - Data encrypted in the Kernel



Is it safe?

ILLINOIS SECURITY LAB

- Kernel (6/9)
  - Password Protection
    - User defined
    - Prevents unauthorized access to the device
    - Protects the cryptographic key for full filesystem encryption

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (7/9)
  - Device Administration
    - >= 2.2
    - API for security-aware enterprise applications
    - Password policies enforcement
    - Remotely wipe handsets

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (8/9)
  - Memory Management Security Enhancements
    - Memory Corruption Mitigation
      - ASLR
      - DEP

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (9/9)
  - Rooting of Devices
    - Who is Root?
      - Kernel
      - small subset of core applications
    - Root can modify:
      - the OS
      - the Kernel
      - other applications

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (9/9) cont'd
  - Why Root?
    - Developers
      - Debugging
      - Access features not present in the API
  - User Data?
    - Bootloader erases any existing user data as part of the unlock step
    - **Rooting through kernel exploits, bypasses this protection**

Is it safe?

ILLINOIS SECURITY LAB

- Kernel (9/9) cont'd
  - Data Encryption?
    - key stored on device?
    - key stored off device?
      - password, stored on a server
      - AT SOME POINT THE KEY MUST BE PROVIDED TO THE APPLICATION
  - More robust approach
    - Hardware solutions by OEMs
  - Lost/Stolen Device
    - Edev.pass(encryption key) used to Ee.k(filesystem)

Is it safe?

ILLINOIS SECURITY LAB

- Application Security
  - Application Elements
  - Permission Model
  - Interprocess Communication
  - Cost Sensitive APIs
  - SIM card access
  - Personal Information
  - Sensitive Data Input Devices
  - Device Metadata
  - Application Signing

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (1/9)
  - Application Elements
    - AndroidManifest.xml
    - Activities
      - One per screen (typically)
    - Services
      - Background processes
    - BroadcastReceiver
      - It's your mailbox!
    - Content Provider
      - Store & Share Data

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (2/9)
  - Permission Model
    - "*no application by default, has permission to perform any operations that would adversely impact other applications, the operating system, or the user.*"
    - Sensitive APIs are protected through Permissions
    - Some capabilities are protected by an intentional lack of APIs
    - 138 available Permissions (TODO : confirm number)

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (2/9)
  - Permission Model cnt'd
    - Resources are only accessible through the OS
      - Do you want to use a capability? Ask for it
        - How? AndroidManifest.xml

```
<uses-permission
    android:name="android.permission.ACCESS_COARSE_LOCATION"></uses-permission>
<uses-permission
    android:name="android.permission.ACCESS_FINE_LOCATION"></uses-permission>
<uses-permission
    android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission
    android:name="android.permission.AUTHENTICATE_ACCOUNTS" />
<uses-permission
    android:name="android.permission.CAMERA" />
<uses-permission
    android:name="android.permission.GET_ACCOUNTS" />
```

      - User must approve all your requests to install your app
        - No turning back! Once granted cannot be ungranted. hmm.. are you sure? (/data/system/packages.xml)

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (2/9)
  - Permission Model cnt'd
    - Try to use capabilities without permission
      - permission failure (printed on system log)
      - SecurityException
        - not always (sendBroadcast(Intent))
    - Permission Enforcement
      - call
      - activity start
      - send/rcv broadcasts
      - accessing content providers
      - binding to or starting a service

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (2/9)
  - Permission Model cnt'd
    - You can enforce your own permissions as well!
      - control who can launch your Activity
      - In General, "protect" your application's resources
      - Define it
        - android:protectionLevel
          - normal
          - dangerous
      - Declare it
        - android:permission

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (3/9)
  - Interprocess Communication
    - filesystems, local sockets, signals (with respect to Linux Permissions)
    - Android introduces :
      - Binder
      - Intent
      - Service
      - ContentProvider

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (4/9)
  - Cost Sensitive APIs
    - any API that might be costly to the user or the Network
      - Telephony
      - SMS/MMS
      - Network/Data
      - In-App Billing
      - NFC access
    - Protected by the OS
    - User must grant explicit permission

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (5/9)
  - SIM card access
    - 3rd party apps don't have low level access to SIM (OS handles all communication with it)
    - no access to AT commands (Radio Interface Layer manages them and it doesn't provide any API for accessing them)

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (6/9)
  - Personal Information
    - Access through protected APIs
    - Data collected by 3rd party apps?
      - Developers should address this with permissions

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (7/9)
  - Sensitive Data Input Devices
    - Camera
    - GPS
    - Microphone
  - Security Mechanism is (drum roll..):
    - Permissions!! (applause)

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (8/9)
  - Device Metadata
    - data that can leak user information indirectly
    - SystemLogs
    - Browser History (Memento paper)
    - phone number
    - hardware / network identification
    - Permission protected!

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (9/9)
  - Application Signing
    - Aim: Identify the author
    - If you don't sign it?
      - Can't publish to GooglepLay
      - package installer will refuse to install it
    - Signed App Certificate defines the UID on the device
    - share UID only if the Pkey in the certificate matches the Pkey of any other installed app

Is it safe?

ILLINOIS SECURITY LAB

- Application Security (9/9)
  - Application Signing cnt'd
    - Certificates
      - Self-signed
      - No CA verification

Some beg to differ!

ILLINOIS SECURITY LAB

- Jana, S. and Shmatikov, V. 2012. **Memento: Learning secrets from process footprints**. In Security and Privacy (SP),2012 IEEE Symposium on. 143 {157.Namestnikov, Y. 2012. It threat evolution: Q2 2012. http://www.securelist.com/en/analysis/204792239/IT_Threat_Evolution_Q2_2012. [Online; accessed 23-October-2012].
- Schlegel, R., Zhang, K., yong Zhou, X., Intwala, M., Kapadia, A., and Wang, X. 2011. **Soundcomber: A stealthy and context-aware sound trojan for smartphones**. In NDSS.
- Cai, L. and Chen, H. 2011. **Touchlogger: inferring keystrokes on touch screen from smartphone motion.** In Proceedings of the 6th USENIX conference on Hot topics in security. HotSec'11. USENIX Association, Berkeley, CA, USA, 9{9.
- Felt, A. P., Chin, E., Hanna, S., Song, D., and Wagner, D. 2011. **Android permissions demystied**. In Proceedings of the 18th ACM conference on Computer and communications security. CCS '11. ACM, New York, NY, USA, 627{638.
- (TODO: Fix this reference) Privilege Escalation Attacks on Android, Ldavi, A.Dmitrienko, A-R.Sadeghi, M.Winandy

ILLINOIS SECURITY LAB

Malware & Vulnerabilities

Android malware

Android activation will reach 1 billion by Nov. 2013

~85% of all mobile malware attacks since 2011 on Android

Android malware families quadruple from 2011 to 2012

Google Play w/ no central virus scanner

# of Android malware from Q3 2011 to Q2 2012

Glossary

ILLINOIS SECURITY LAB

- **Vulnerability**: A flaw on the kernel, OS or application
- **Exploit**: A utilization of that vulnerability to achieve a goal
- **Malware**: It can use an exploit for a malevolent purpose

Android Vulnerabilities

ILLINOIS SECURITY LAB

Google » Android : Security Vulnerabilities

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9

Sort Results By : Cve Number Descending Cve Number Ascending CVSS Score Descending Number Of Exploits Descending

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Confidentiality	Integrity	Availability
1	CVE-2012-3979			Exec Code	2012-08-29	2012-08-29	6.8	None	Remote	Medium	Not required	Partial	Partial	Partial
Mozilla Firefox before 15.0 on Android does not properly implement unspecified callers of the __android_log_print function, which allows remote attackers to execute arbitrary code via a crafted web page that calls the JavaScript dump function.														
2	CVE-2011-4276	200		+Info	2012-01-25	2012-01-26	4.3	None	Remote	Medium	Not required	Partial	None	None
The Bluetooth service (com/android/phone/BluetoothHeadsetService.java) in Android 2.3 before 2.3.6 allows remote attackers within Bluetooth range to obtain contact data via an AT phonebook transfer.														
3	CVE-2011-3975	200		+Info	2011-10-03	2011-10-20	2.6	None	Remote	High	Not required	Partial	None	None
A certain HTC update for Android 2.3.4 build GRJ22, when the Sense interface is used on the HTC EVO 3D, EVO 4G, ThunderBolt, and unspecified other devices, provides the HtcLoggers.apk application, which allows user-assisted remote attackers to obtain a list of telephone numbers from a log, and other sensitive information, by leveraging the android.permission.INTERNET application permission and establishing TCP sessions to 127.0.0.1 on port 65511 and a second port.														
4	CVE-2011-3918	399		DoS	2012-10-07	2012-10-08	7.8	None	Remote	Low	Not required	None	None	Complete
The Zygote process in Android 4.0.3 and earlier accepts fork requests from processes with arbitrary UIDs, which allows remote attackers to cause a denial of service (reboot loop) via a crafted application.														
5	CVE-2011-3874	119		Exec Code Overflow	2012-01-27	2012-02-06	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Stack-based buffer overflow in libsysutils in Android 2.2.x through 2.2.2 and 2.3.x through 2.3.6 allows user-assisted remote attackers to execute arbitrary code via an application that calls the FrameworkListener::dispatchCommand method with the wrong number of arguments, as demonstrated by zergRush to trigger a use-after-free error.														

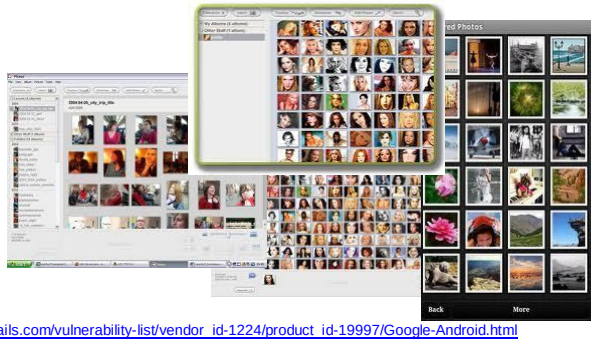
http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html

Picasa: Photos Anyone?

ILLINOIS SECURITY LAB

7 CVE-2011-2344 310	+Priv	2011-07-08	2011-07-08	10.0	None	Remote	Low	Not required	Complete	Complete	Complete

Android Picasa in Android 3.0 and 2.x through 2.3.4 uses a cleartext HTTP session when transmitting the authToken obtained from ClientLogin, which allows remote attackers to gain privileges and access private pictures and web albums by sniffing the token from connections with picasaweb.google.com.



http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html

Picasa: Photos Anyone?

ILLINOIS SECURITY LAB

Vulnerability Details : CVE-2011-2344

Android Picasa in Android 3.0 and 2.x through 2.3.4 uses a cleartext HTTP session when transmitting the authToken obtained from ClientLogin, which allows remote attackers to gai
access private pictures and web albums by sniffing the token from connections with picasaweb.google.com.
Publish Date : 2011-07-08 Last Update Date : 2011-07-08

Collapse All Expand All Select Select&Copy Scroll To Comments External Links
Related Tweets Even more tweets Search Twitter Search YouTube Search Google

– CVSS Scores & Vulnerability Types

Cvss Score	10.0
Confidentiality Impact	Complete (There is total information disclosure, resulting in all system files being revealed.)
Integrity Impact	Complete (There is a total compromise of system integrity. There is a complete loss of system protection, resulting in the entire system being compromised.)
Availability Impact	Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Gain privileges
CWE ID	310

– Products Affected By CVE-2011-2344

#	Product Type	Vendor	Product	Version	Update	Edition	Language	
1	OS	Google	Android	2.1				Details Vulnerabilities
2	OS	Google	Android	2.2	Rev1			Details Vulnerabilities
3	OS	Google	Android	2.2				Details Vulnerabilities
4	OS	Google	Android	2.2.1				Details Vulnerabilities
5	OS	Google	Android	2.2.2				Details Vulnerabilities
6	OS	Google	Android	2.3	Rev1			Details Vulnerabilities
7	OS	Google	Android	2.3.3				Details Vulnerabilities
8	OS	Google	Android	2.3.4				Details Vulnerabilities
9	OS	Google	Android	3.0				Details Vulnerabilities

http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html

Picasa: Photos Anyone?

ILLINOIS SECURITY LAB

Catching AuthTokens in the Wild
The Insecurity of Google's ClientLogin Protocol

by Bastian Könings, Jens Nickels, and Florian Schaub

UPDATE, June 15, 2011

Google has released patches for securing the Picasa synchronization as well. The patches are available in the Android open source code repository as part of the Gallery3D application for Android 2.1 (Eclair), 2.2 (Froyo), and 2.3 (Gingerbread). However, as the app became the default pre-installed gallery app in Android 2.3, it is not clear whether and how the patched app is going to be pushed on 2.3 devices.

UPDATE, May 20, 2011

Google announced that they are going to fix the issue also for devices with older Android versions. The fix does not require an update of the Android OS and will be transparent to the user. So, as far as we know, users will not get any feedback when the update will be available on their devices. The fix is based on a changed configuration file for Google services on the device. The update mechanism might be similar to the application removal or Android Cloud to Device Messaging (C2DM) features. The update will only ensure encrypted synchronization of Calendar and Contacts. The Picasa synchronization, which was integrated in Android 2.3, will remain unencrypted.

http://www.uni-ulm.de

Skype Privacy Leak

ILLINOIS SECURITY LAB


```
# ls -l /data/data/com.skype.merlin_mecha/files/shared.xml
-rw-rw-rw- app_152  app_152     56136 2011-04-13 00:07 shared.xml

# grep Default /data/data/com.skype.merlin_mecha/files/shared.xml
    <Default>jcaseap</Default>
```

```
# ls -l /data/data/com.skype.merlin_mecha/files/jcaseap
-rw-rw-rw- app_152  app_152    331776 2011-04-13 00:08 main.db
-rw-rw-rw- app_152  app_152    119528 2011-04-13 00:08 main.db-journal
-rw-rw-rw- app_152  app_152     40960 2011-04-11 14:05 keyval.db
-rw-rw-rw- app_152  app_152      3522 2011-04-12 23:39 config.xml
drwxrwxrwx app_152  app_152           2011-04-11 14:05 voicemail
-rw-rw-rw- app_152  app_152         0 2011-04-11 14:05 config.lck
-rw-rw-rw- app_152  app_152     61440 2011-04-13 00:08 bistats.db
drwxrwxrwx app_152  app_152           2011-04-12 21:49 chatsync
-rw-rw-rw- app_152  app_152     12824 2011-04-11 14:05 keyval.db-journal
-rw-rw-rw- app_152  app_152     33344 2011-04-13 00:08 bistats.db-journal
```

http://www.androidpolice.com

Skype Privacy Leak

ILLINOIS SECURITY LAB

15th April 2011

[Fixed] Privacy vulnerability in Skype for Android

Adrian Asher

20 April 2011: This vulnerability has been fixed. Please update Skype on your Android device.

It has been brought to our attention that, were you to install a malicious third-party application onto your Android device, then it could access the locally stored Skype for Android files.

These files include cached profile information and instant messages. We take your privacy very seriously and are working quickly to protect you from this vulnerability, including securing the file permissions on the Skype for Android application.

To protect your personal information, we advise users to take care in selecting which applications to download and install onto their device.

Posted to: Privacy

http://blogs.skype.com/security/2011/04/privacy_vulnerability_in_skype.html

Skype Privacy Leak

ILLINOIS SECURITY LAB

20th April 2011

Privacy vulnerability in Skype for Android fixed

Adrian Asher

After a period of developing and testing we have released a new version of the Skype for Android application onto the Android Market, containing a fix to the vulnerability reported to us. Please update to this version as soon as possible in order to help protect your information.

We have had no reported examples of any 3rd party malicious application misusing information from the Skype directory on Android devices and will continue to monitor closely. Please rest assured that we do take your privacy and security very seriously and we sincerely apologise for any concern this issue may have caused.

Please ensure that you download Skype only from skype.com, or from the Android Market links on skype.com.

Posted to: Privacy

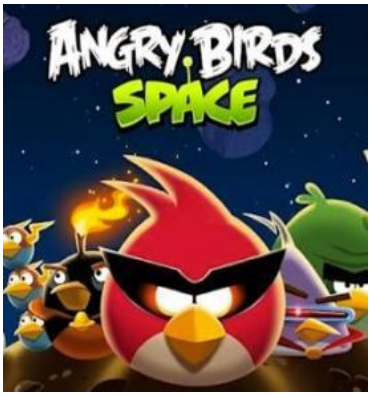
http://blogs.skype.com/security/2011/04/privacy_vulnerability_in_skype.html

Free Games: Really?

ILLINOIS SECURITY LAB

Not just private information but your money

- Angry Birds, Assassin's Creed, Cut the Rope
- Repackaged version on unofficial application stores
- permissions - "sending messages at premium rates"
- send messages every time app is started
- intercept and consume incoming credit card transaction alerts



Free Games: Really?

ILLINOIS SECURITY LAB

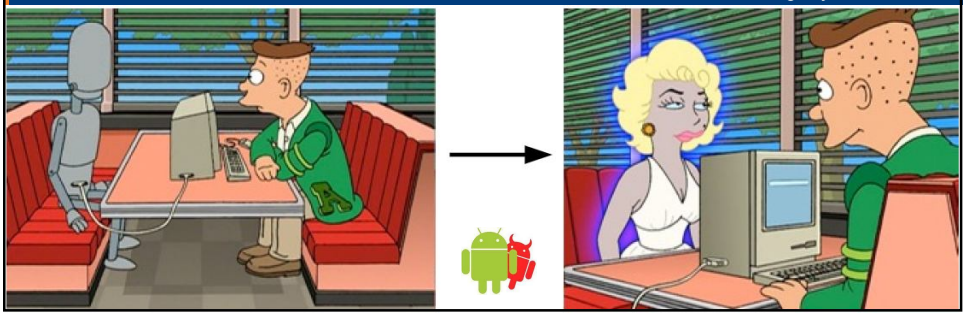
GingerBreak!

- root exploit
- install malicious code
- communicate to remote websites to compromise phone information
- download and install further malware
- your device - a part of botnet under the control of hackers

ILLINOIS SECURITY LAB

Root Exploits

Image by TEAM JOCH



ILLINOIS SECURITY LAB

Root Exploits

- Who wants to be root?
  - Developers
    - Debugging
    - Access hidden capabilities
  - Owner
    - Customization
  - Malware
    - Circumvent security
- It's like "jailbreaking"
  - usually are based on a Kernel exploit

Root Exploits

ILLINOIS SECURITY LAB

- User Data?
  - Bootloader erases any existing user data as part of the unlock step
  - **Rooting through kernel exploits, bypasses this protection**

Root Exploits

ILLINOIS SECURITY LAB

- RageAgainstTheCage - ADB Exhaustion attack
  - Linux - RLIMIT_NPROC
    - defines the max number of simultaneous processes allowed for a userId
  - **Android Debug Bridge**
    - adb daemon is root by default
      - on emulator
      - on a device configured in debug mode
    - Otherwise…
      - tries to downgrade its privileges to shell user (**AID_SHELL**) by calling setuid
    - Hmmm. **FORK BOMB!!**

Root Exploits

ILLINOIS SECURITY LAB

- RageAgainstTheCage - ADB Exhaustion attack cnt'd
  - **FORK BOMB**
    - fork processes out of the adb daemon which is now running as AID_SHELL user
    - once we exceed RLIMIT_NPROC adb dies
    - adb tries to restart
      - it starts as root (default)
      - it tries to call setuid
        - Ooops.. no more processes are allowed for AID_SHELL
        - adb continues as AID_ROOT (there is no check the setuid return value)

ILLINOIS SECURITY LAB

GingerBreak

GingerBreak

ILLINOIS SECURITY LAB

Impact

- Android 2.2(Froyo) and Android 2.3(GingerBread)
- Repackaged app on unofficial app stores

8	CVE-2011-1823	189	Exec Code +Priv	Mem. Corr.	Bypass	2011-06-09	2012-04-25	7.2	Admin	Local	Low	Not required	Complete	Complete	Complete

The vold volume manager daemon on Android 3.0 and 2.x before 2.3.4 trusts messages that are received from a PF_NETLINK socket, which allows local users to execute arbitrary code and gain root privileges via a negative index that bypasses a maximum-only signed integer check in the DirectVolume::handlePartitionAdded method, which triggers memory corruption, as demonstrated by Gingerbreak.

http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html

GingerBreak

ILLINOIS SECURITY LAB

Vulnerability - VOLD(Volume Manager Daemon)

- mPartMinors and part_num - signed integers
- assumes Netlink messages used for IPC b/w kernel and user to be initiated by kernel only
- DirectVolume::handlePartitionAdded method
  - part_num and minor variables initialized from NetLink message configurations
- part_num checked for MAX value(but not for negative)!
  - **mPartMinors[part_num -1] = minor**

GingerBreak

ILLINOIS SECURITY LAB

Exploit

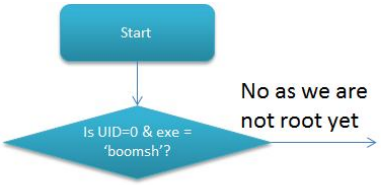
- places the rooting binary at a valid address inside the memory space of vold by exploiting the vulnerability
- modifies the .got table address of one of the shared functions (like atoi() or strcmp()) for vold and points it to the rooting binary address

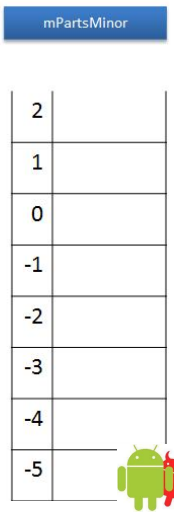
Result

- vold executes the rooting binary whenever it tries to call the overwritten shared function
- the device is rooted as vold was running as system process and hence the rooting binary was executed by root itself

GingerBreak

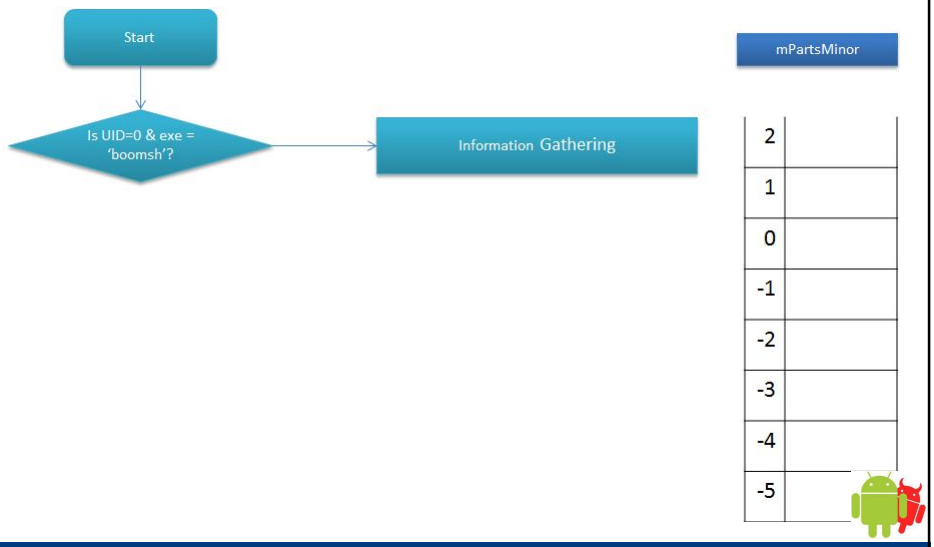
ILLINOIS SECURITY LAB





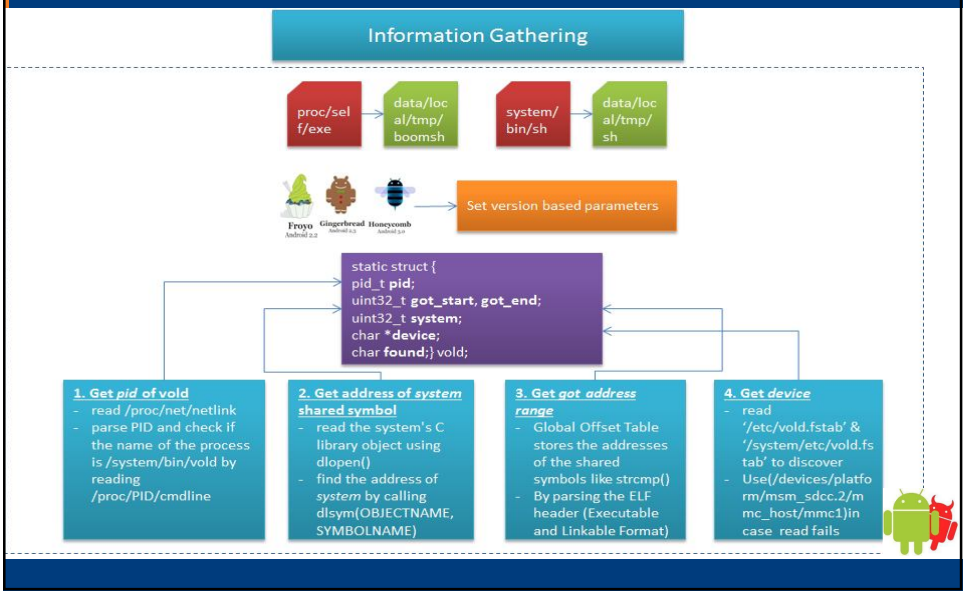
GingerBreak

ILLINOIS SECURITY LAB



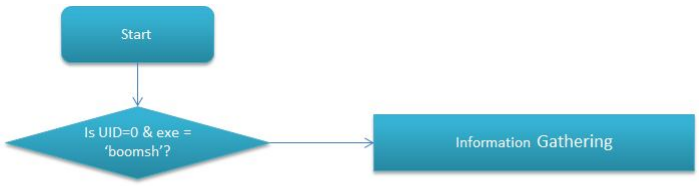
GingerBreak

ILLINOIS SECURITY LAB



GingerBreak

ILLINOIS SECURITY LAB

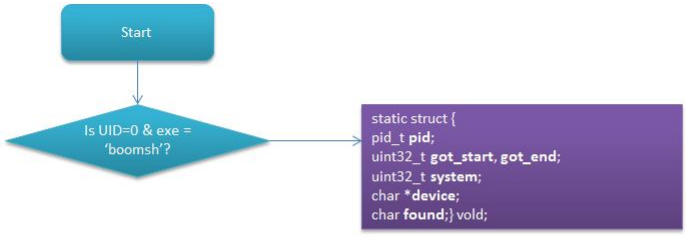


mPartsMinor

2	
1	
0	
-1	
-2	
-3	
-4	
-5	

GingerBreak

ILLINOIS SECURITY LAB

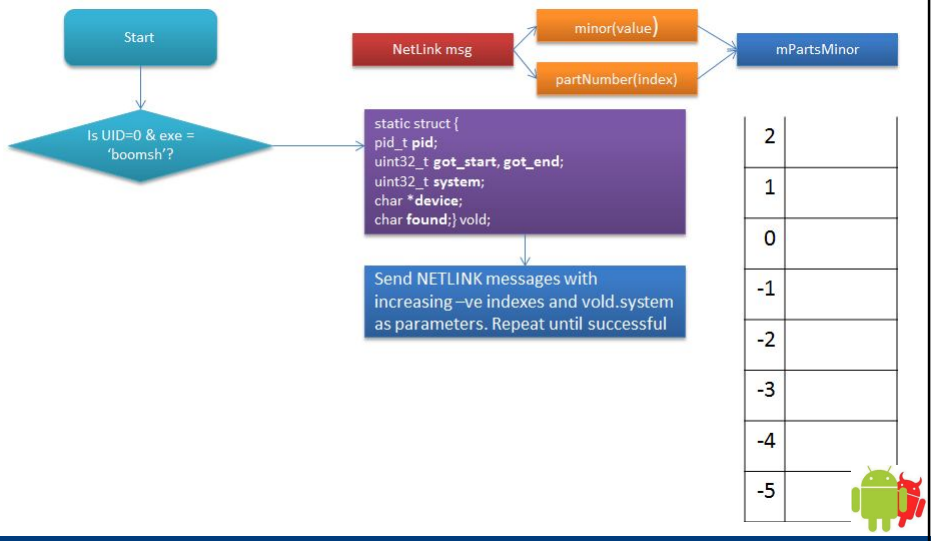


mPartsMinor

2	
1	
0	
-1	
-2	
-3	
-4	
-5	

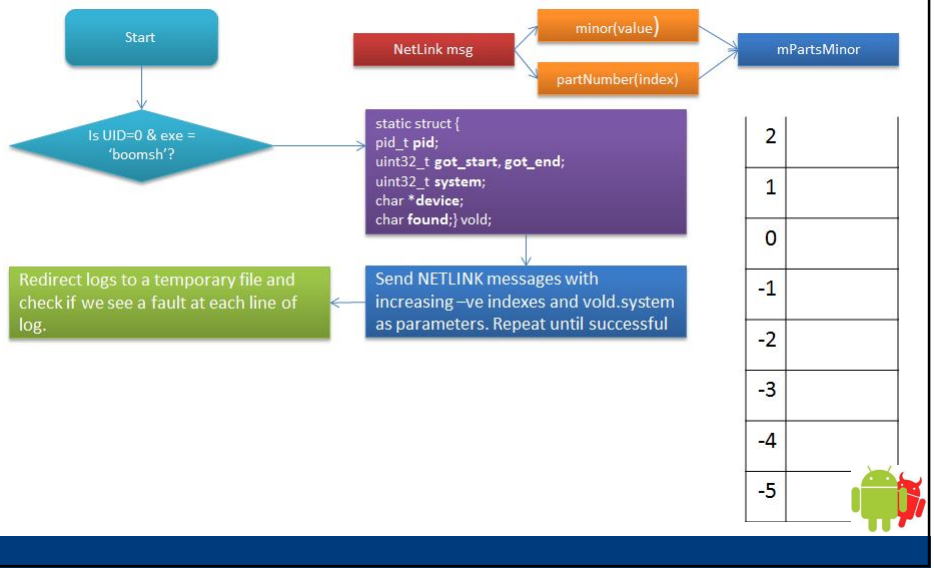
GingerBreak

ILLINOIS SECURITY LAB



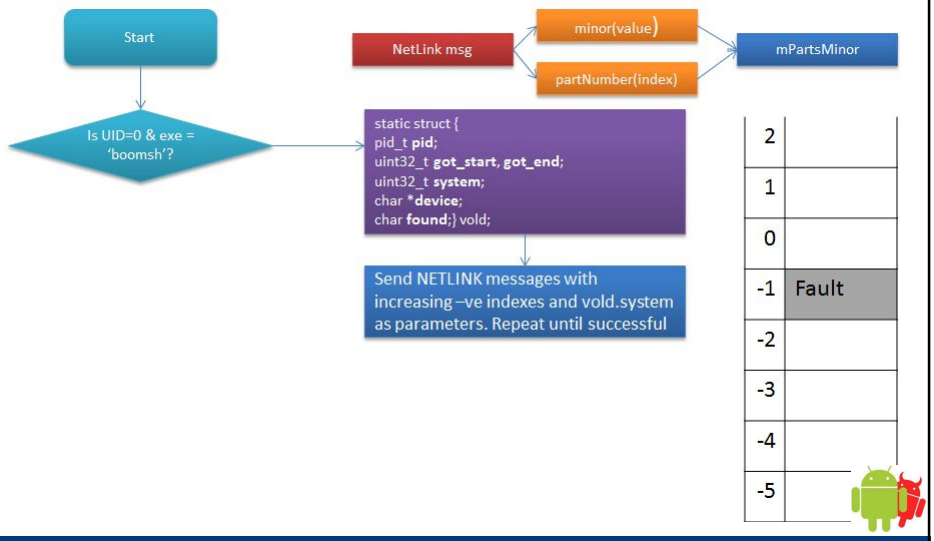
GingerBreak

ILLINOIS SECURITY LAB



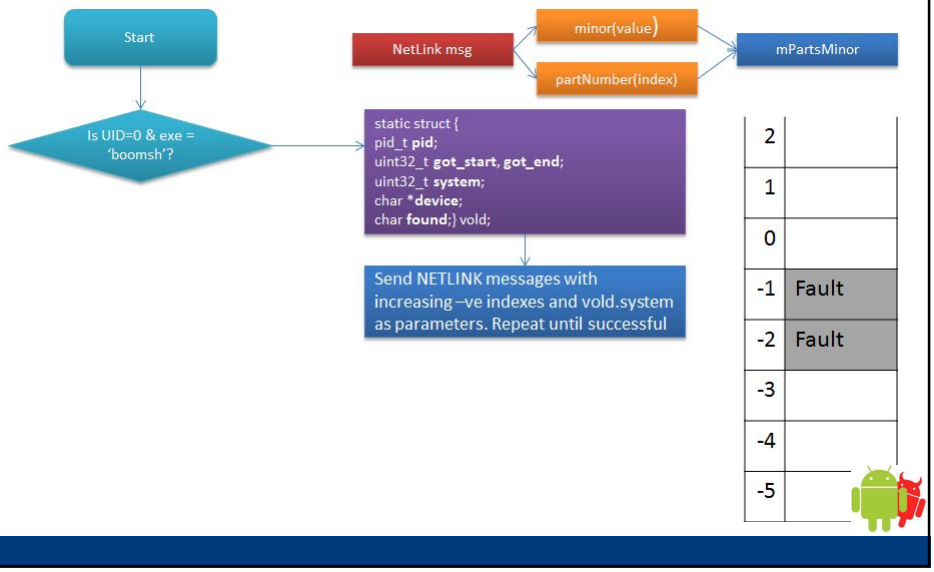
GingerBreak

ILLINOIS SECURITY LAB



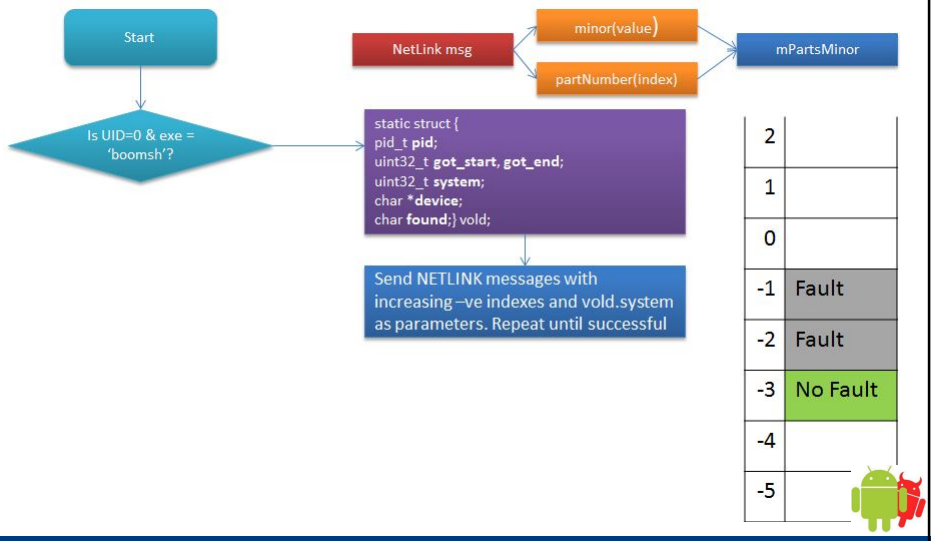
GingerBreak

ILLINOIS SECURITY LAB



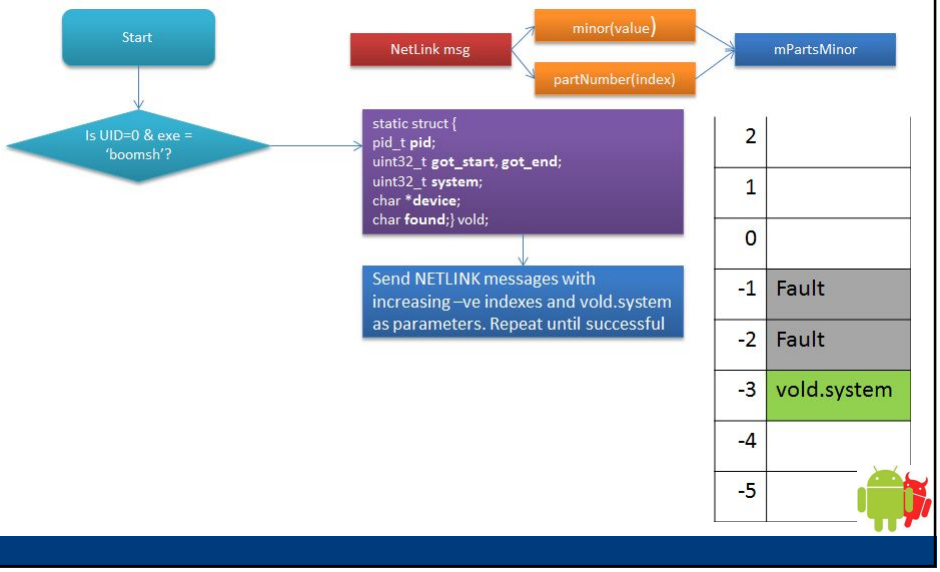
GingerBreak

ILLINOIS SECURITY LAB



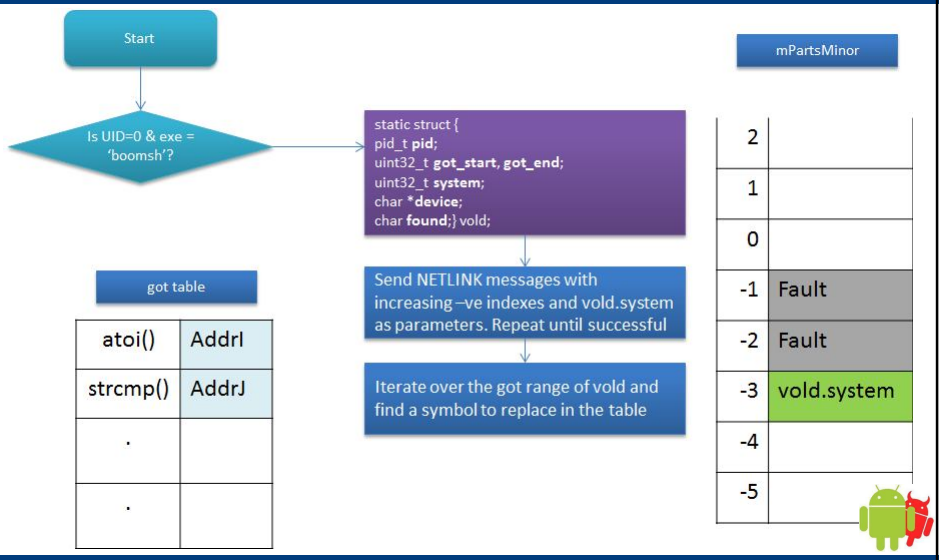
GingerBreak

ILLINOIS SECURITY LAB



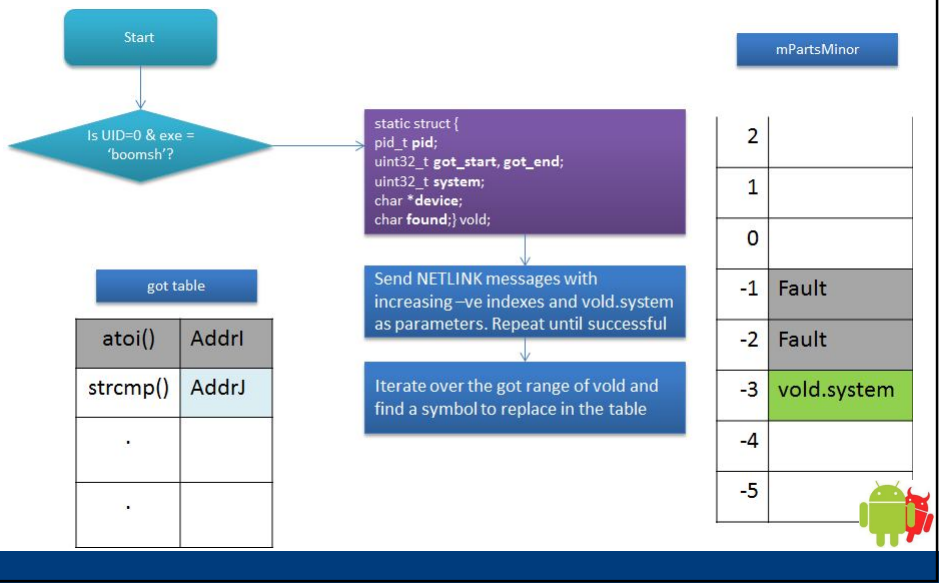
GingerBreak

ILLINOIS SECURITY LAB



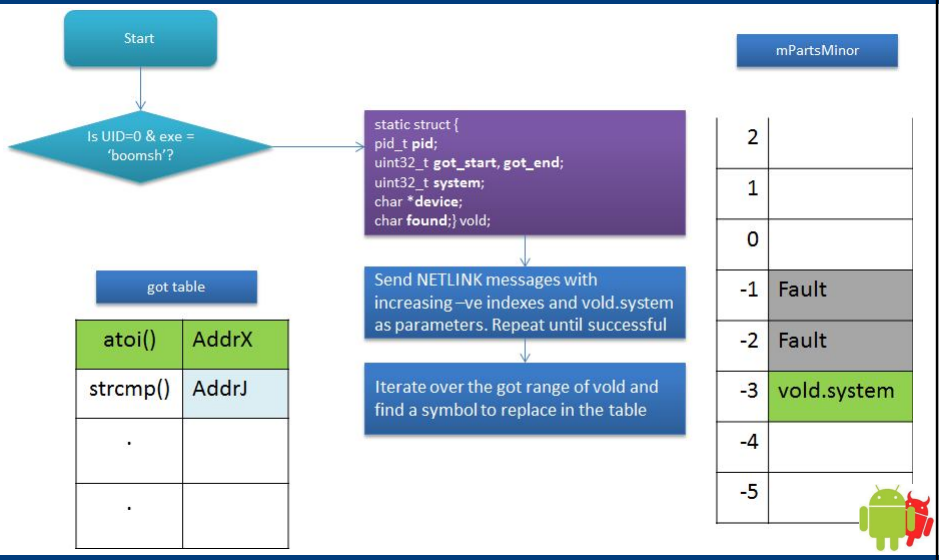
GingerBreak

ILLINOIS SECURITY LAB



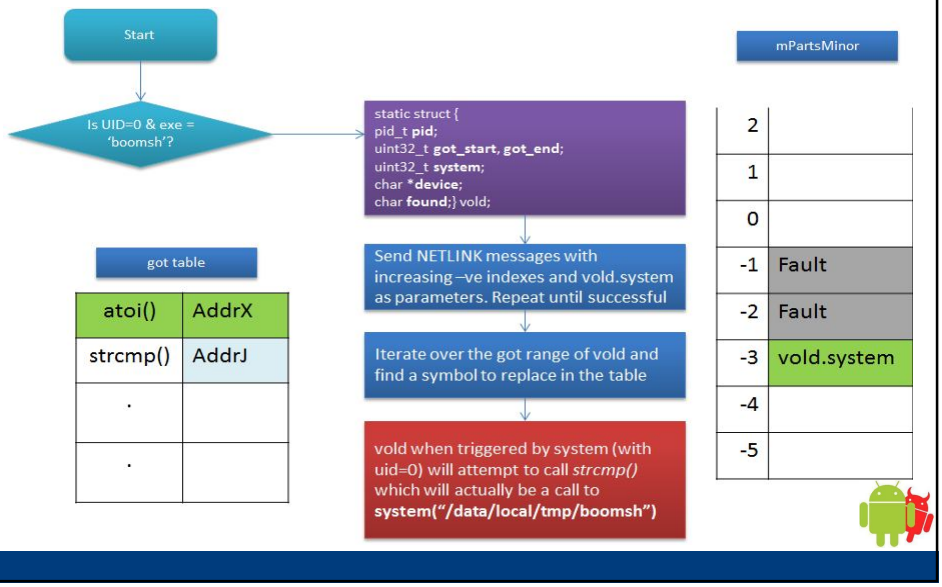
GingerBreak

ILLINOIS SECURITY LAB



GingerBreak

ILLINOIS SECURITY LAB



GingerBreak

ILLINOIS SECURITY LAB

vold when triggered by system (with uid=0) will attempt to call *strcmp()* which will actually be a call to:
system("/data/local/tmp/boomsh")

GingerBreak

ILLINOIS SECURITY LAB

GingerMaster - Ginger Break For Me

- gbfm.png
- registers a receiver so that it is notified when the system finishes booting
- launches a background service which collects information like device id, phone number etc.
- establishes a bidirectional channel with a remote server
  - uploads the information
  - downloads new malware

http://www.csc.ncsu.edu/faculty/jiang/GingerMaster/

Approach

Approach

- Thread Model
  - **User**
    - benign
    - rooted her phone for better performance and full customization capabilities.
  - **Device**
    - an official device (hardware untainted)
    - we trust the device to perform all its legal operations correctly
    - User processes have root privileges

Approach

ILLINOIS SECURITY LAB

- Thread Model cnt'd
  - **Attacker**
    - Passive
      - Silently reads user's private data
    - Establishes a communication channel with a remote computer administered by the attacker
    - Sends the private information to the remote location
    - Stealthy
      - request as less permission as possible
      - do malicious operation in the background
      - conceals its functionality by offering benign operations

ILLINOIS SECURITY LAB

Design - Implementation

Design - Implementation - A

ILLINOIS SECURITY LAB

GET CONTACTS PERMISSION

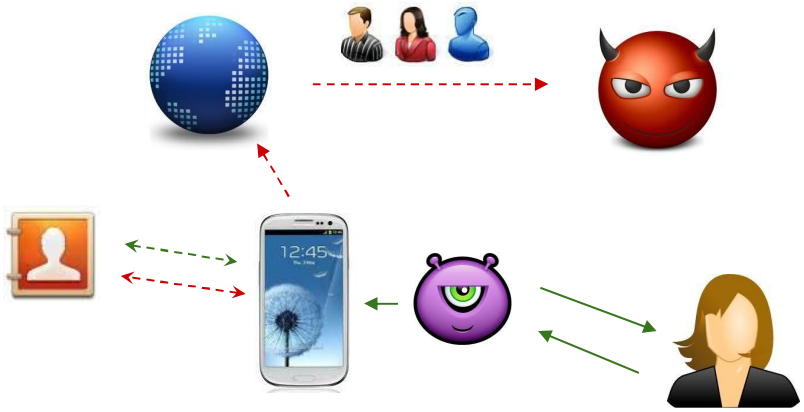
Design - Implementation - A

ILLINOIS SECURITY LAB

GET CONTACTS PERMISSION ✓

Design - Implementation - A

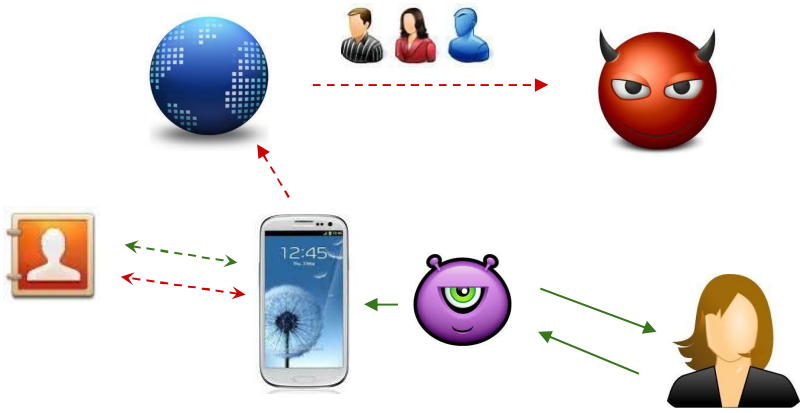
ILLINOIS SECURITY LAB



GET CONTACTS PERMISSION ✓

Design - Implementation - A

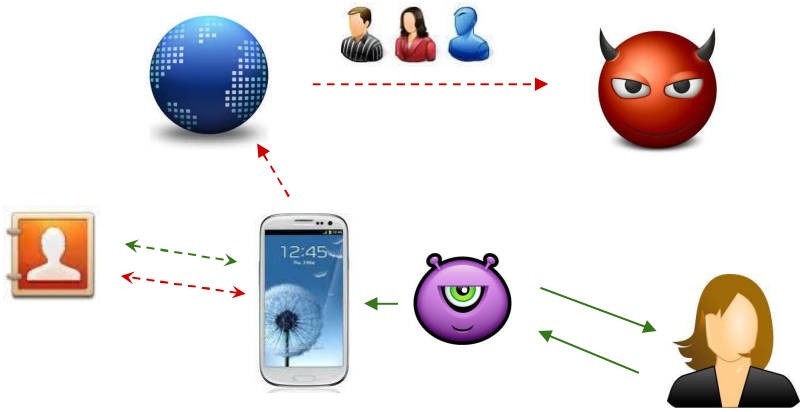
ILLINOIS SECURITY LAB



INTERNET PERMISSION ?

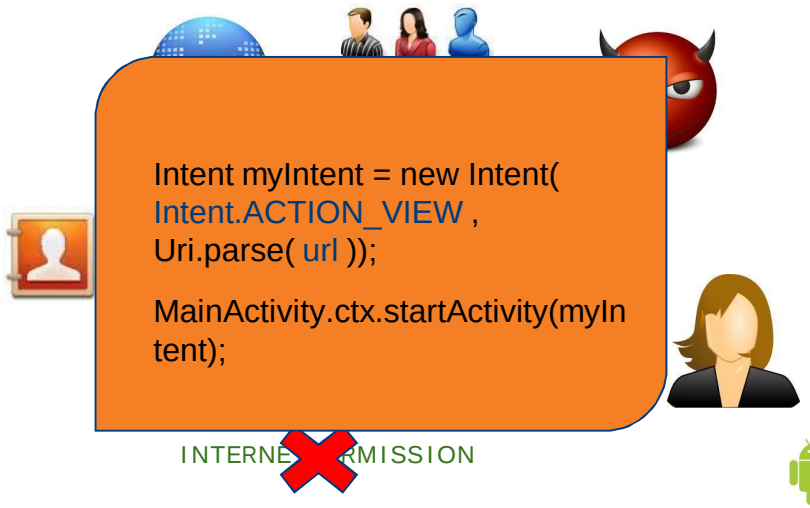
Design - Implementation - A

ILLINOIS SECURITY LAB



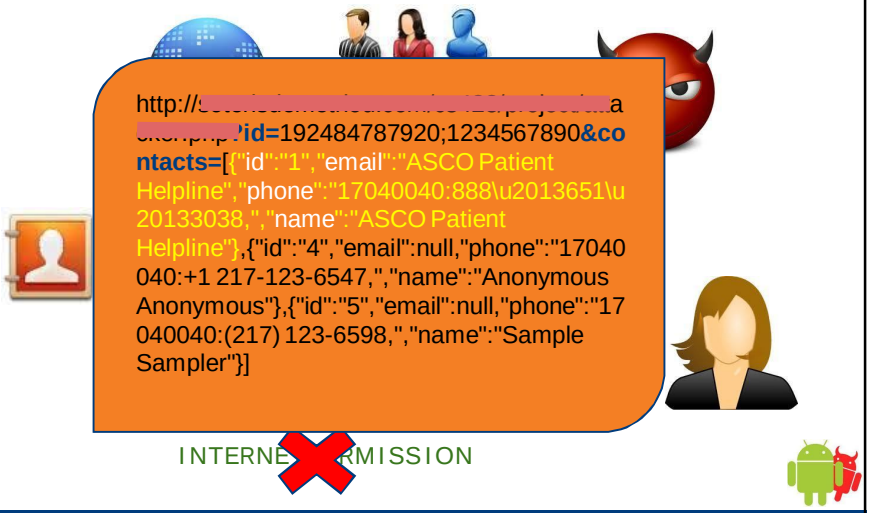
Design - Implementation - A

ILLINOIS SECURITY LAB



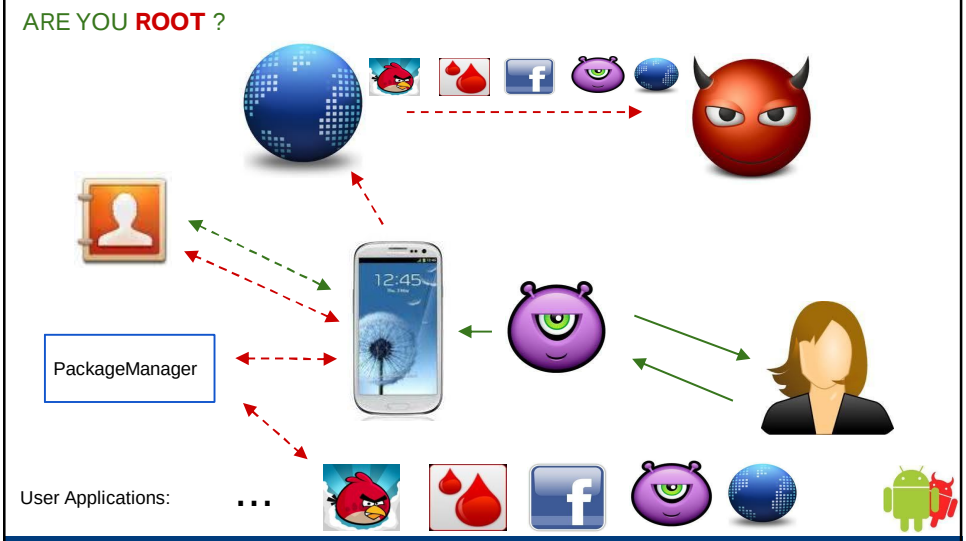
Design - Implementation - A

ILLINOIS SECURITY LAB



Design - Implementation - B

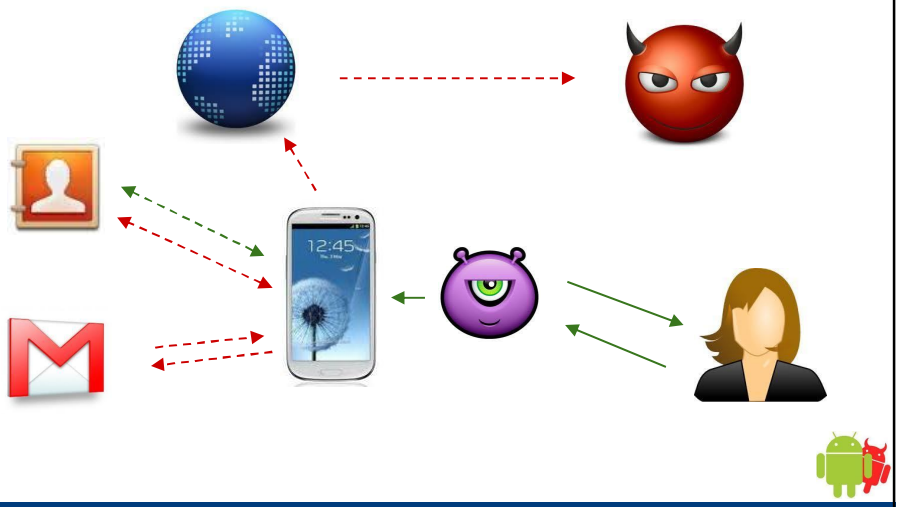
ILLINOIS SECURITY LAB



Design - Implementation - C

ILLINOIS SECURITY LAB

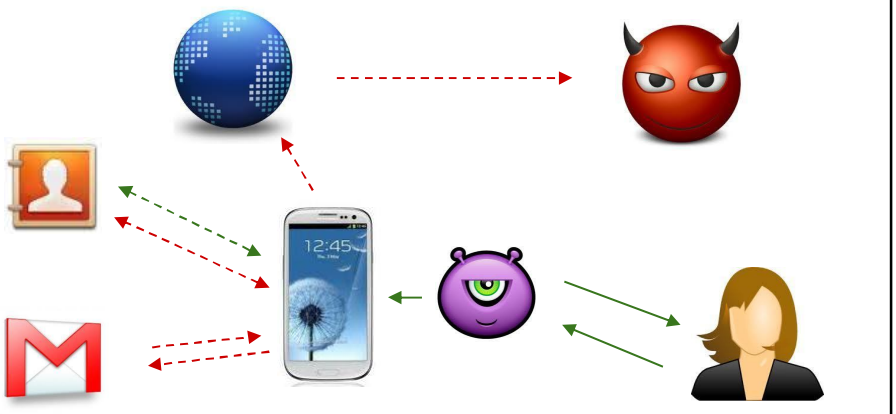
ARE YOU **ROOT** ?



Design - Implementation - C

ILLINOIS SECURITY LAB

ARE YOU **ROOT** ?

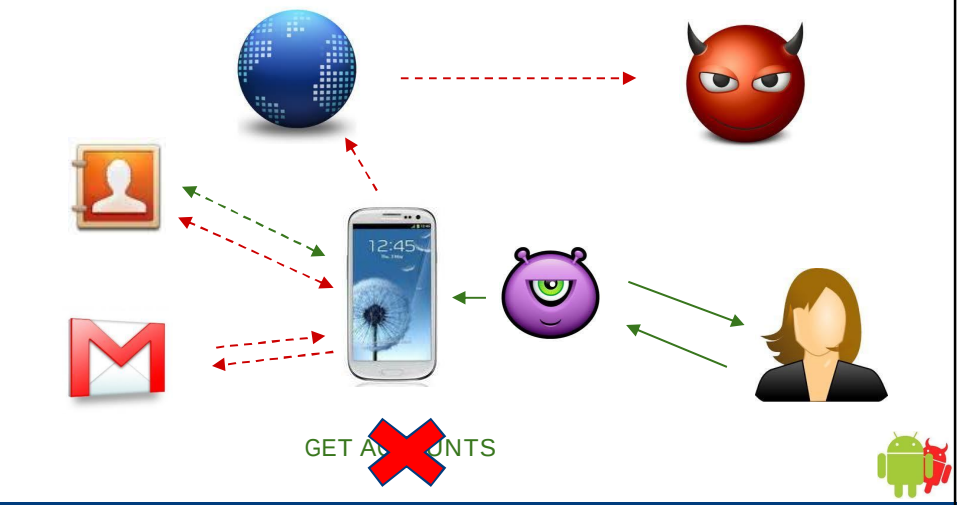


GET ACCOUNTS ?

Design - Implementation - C

ILLINOIS SECURITY LAB

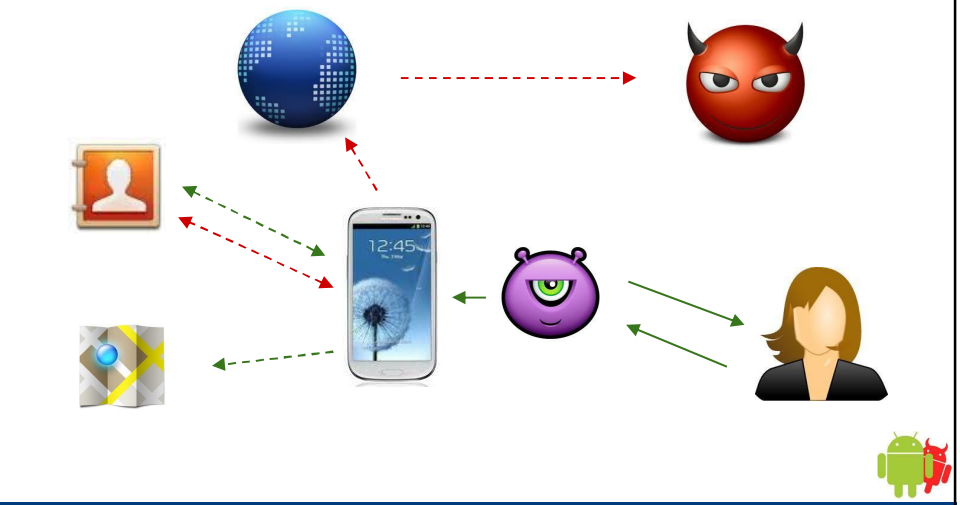
ARE YOU **ROOT** ?



Design - Implementation - D

ILLINOIS SECURITY LAB

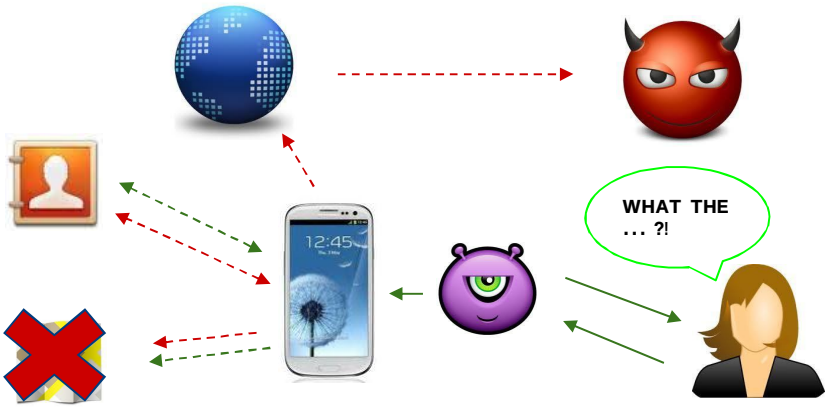
ARE YOU **ROOT** ?



Design - Implementation - D

ILLINOIS SECURITY LAB

ARE YOU **ROOT** ?



Demonstration

ILLINOIS SECURITY LAB

ILLINOIS SECURITY LAB

ILLINOIS SECURITY LAB

Improvements

Improvements

ILLINOIS SECURITY LAB

- Send the data whenever the screen is off
- Save data to a local DB and send them gradually (in chunks) whenever:
  - the screen is OFF
  - WiFi is connected
- Get User's telephone number
- Get Device's Location
- Disable Superuser Notifications
- A malware needs to be stealthy
  - mind app's CPU usage
  - mind app's Memory usage
  - mind the Permissions it requests

References

ILLINOIS SECURITY LAB

ILLINOIS SECURITY LAB

REFERENCES

Cai, L. and Chen, H. 2011. Touchlogger: inferring keystrokes on touch screen from smartphone motion. In Proceedings of the 6th USENIX conference on Hot topics in security. HotSec'11. USENIX Association, Berkeley, CA, USA, 9{9.

Felt, A. P., Chin, E., Hanna, S., Song, D., and Wagner, D. 2011. Android permissions demystied. In Proceedings of the 18th ACM conference on Computer and communications security. CCS '11. ACM, New York, NY, USA, 627{638.

Felt, A. P., Finifter, M., Chin, E., Hanna, S., and Wagner, D. 2011. A survey of mobile malware in the wild. In Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. SPSM '11. ACM,New York, NY, USA, 3{14.

Felt, A. P., Wang, H. J., Moshchuk, A., Hanna, S., and Chin, E. 2011. Permission re-delegation: attacks and defenses. In Proceedings of the 20th USENIX conference on Security. SEC'11. USENIX Association, Berkeley, CA, USA, 22{22.

Framingham, M. 2012. Android and iOS Surge to New Smartphone OS Record in Second Quarter, According to IDC. http://http://www.idc.com/getdoc.jsp?containerId=prUS23638712. [Online; accessed 22-October-2012].

Frank, M., Dong, B., Felt, A. P., and Song, D. 2012. Mining permission request patterns from android and facebook applications (extended author version). CoRR abs/1210.2429.

Grace, M., Zhou, Y., Zhang, Q., Zou, S., and Jiang, X. 2012. Riskranker: scalable and accurate zero-day android malware detection. In Proceedings of the 10th international conference on Mobile systems, applications, and services.MobiSys '12. ACM, New York, NY, USA, 281{294.

Jana, S. and Shmatikov, V. 2012. Memento: Learning secrets from process footprints. In Security and Privacy (SP),2012 IEEE Symposium on. 143 {157.Namestnikov, Y. 2012. It threat evolution: Q2 2012. http://www.securelist.com/en/analysis/204792239/IT_Threat_Evolution_Q2_2012. [Online; accessed 23-October-2012].

Schlegel, R., Zhang, K., yong Zhou, X., Intwala, M., Kapadia, A., and Wang, X. 2011. Soundcomber: A stealthy and context-aware sound trojan for smartphones. In NDSS.

Zhou, W., Zhou, Y., Jiang, X., and Ning, P. 2012. Detecting repackaged smartphone applications in third-party android marketplaces. In CODASPY. 317{326.

Zhou, Y. and Jiang, X. 2012. Dissecting android malware: Characterization and evolution. In IEEE Symposium on Security and Privacy. 95{109.

ILLINOIS SECURITY LAB

http://www.defcon.org/images/defcon-18/dc-18-presentations/Lineberry/DEFCON-18-Lineberry-Not-The-Permissions-You-Are-Looking-For.pdf

GingerBreak(OutOfArrayBounds exploit) - Exploit Walkthrough
http://xorl.wordpress.com/2011/04/28/android-vold-mpartminors-signedness-issue/

GingerBreak - Source Code Downloadable from this link (zip file-binary,c,readme)::
http://c-skills.blogspot.com/2011/04/yummy-yummy-gingerbreak.html

LeNa - technical breakdown
http://blog.mylookout.com/wp-content/uploads/2011/10/LeNa-Legacy-Native-Teardown_Lookout-Mobile-Security1.pdf

RageAgainstTheCage - Logic Flaw exploit to gain Root Access
http://thesnkchrmr.wordpress.com/2011/03/24/rageagainstthecage/

Vulnerabilities DB
http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html
http://stealth.openwall.net/xSports/

Android Security Features
http://source.android.com/tech/security/index.html

Android Rooting
http://www.apriorit.com/our-company/dev-blog/255-android-rooting

http://androidforums.com/casio-g-zone-commando/512496-learning-programming.html

http://threatpost.com/en_us/blogs/new-e-banking-trojans-target-android-users-071012

http://threatpost.com/en_us/blogs/instagram-patches-friendship-vulnerability-privacy-hole-071212

http://arstechnica.com/security/2012/07/android-jelly-bean-hard-to-exploit/

https://blog.duosecurity.com/category/android/

http://www.zer0trusion.com/2011/03/android-local-privilege-escalation.html

http://penturalabs.wordpress.com/2011/03/31/vulnerability-development-buffer-overflows-how-to-bypass-full-aslr/

https://www.defcon.org/images/defcon-18/dc-18-presentations/Lineberry/DEFCON-18-Lineberry-Not-The-Permissions-You-Are-Looking-For.pdf

http://elinux.org/Android_Security

http://www.ibm.com/developerworks/library/x-androidsecurity/index.html

http://immunityinc.com/infiltrate/archives/Android_Attacks.pdf

http://jon.oberheide.org/files/bsides11-dontrootrobots.pdf

http://pages.videotron.com/gravufo/android_flash_en.html

http://www.cvedetails.com/version/103818/Google-Android-2.1.html

http://www.csc.ncsu.edu/faculty/jiang/GingerMaster/

http://xorl.wordpress.com/2011/04/28/android-vold-mpartminors-signedness-issue/

ILLINOIS SECURITY LAB

Conclusion

Conclusion

ILLINOIS SECURITY LAB

- ...but always pay attention to the man behind the curtain

The end!

ILLINOIS SECURITY LAB

Image taken from: http://equal-life.blogspot.com