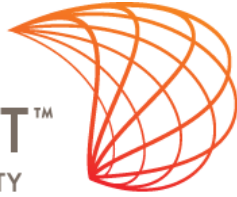**DASIENT**™
SMART WEB SECURITY

# Mobile Malware Madness and How to Cap the Mad Hatters

A Preliminary Look at Mitigating Mobile Malware

Gerry Eisenhaur, Michael N. Gagnon, Tufan Demir, Neil Daswani

# Mobile Malware Madness:
# Recent Case Studies

**DASIENT**™
SMART WEB SECURITY

- DroidDream (Android)
    - Over 50 root'ed apps uploaded to Google app marketplace
    - Conducts data theft; send credentials to attackers

- Ikee.A (iOS)
    - Experiment; changed wallpaper to Rick Astley image
    - Worm capabilities (targeted default ssh pwd)
    - Worked only on jailbroken phones with ssh installed (could have been worse)

- Zitmo (Symbian, BlackBerry, Windows Mobile)
    - Captures mTANs from SMS messages; aimed at defeating 2-factor auth
    - Works in conjunction with Zeus botnet; sent in conjuction with user PC infection
    - Propagates via SMS; claims to install a "security certificate"
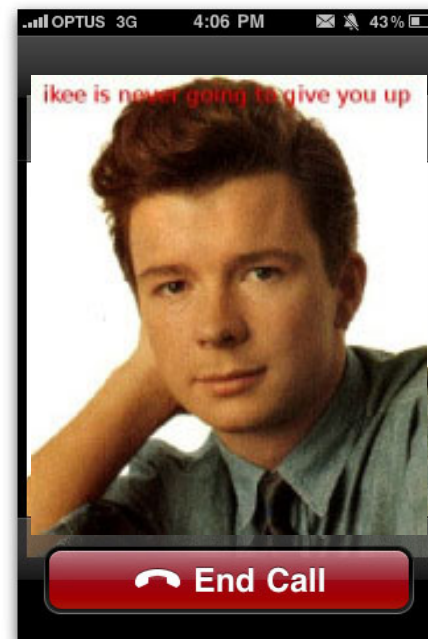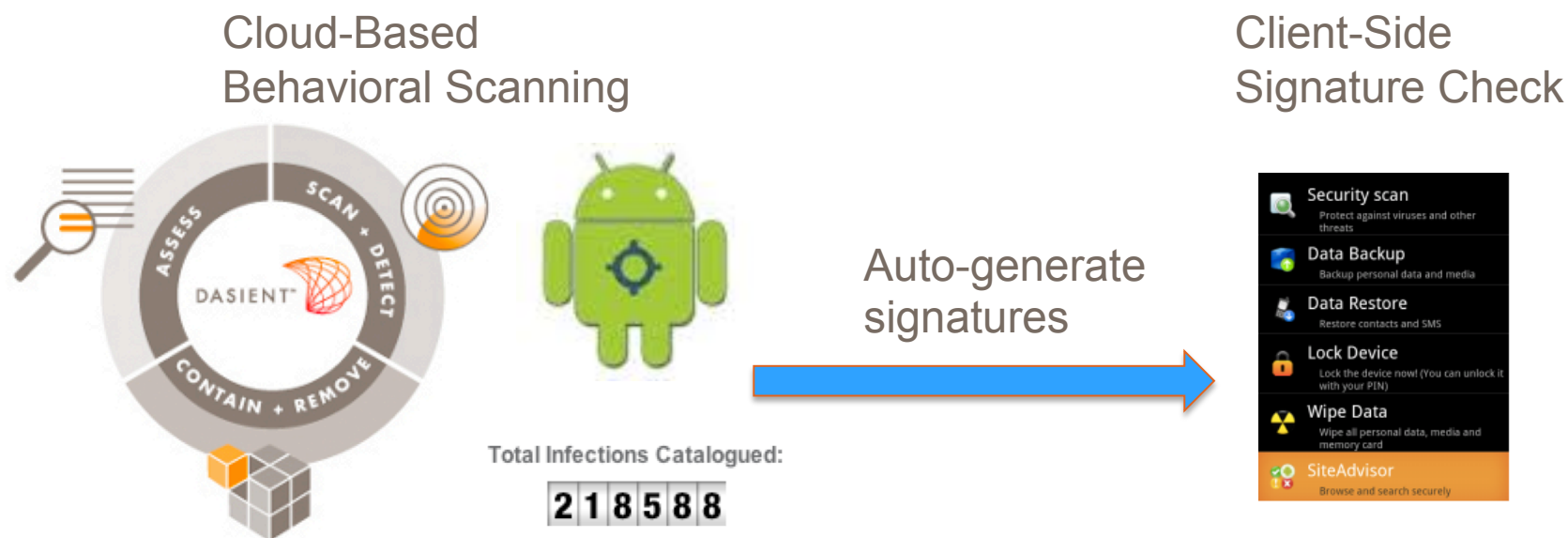
# Behavioral Analysis

- ## What is Behavioral Analysis?
    - Run applications to see what they do – assess their "behavior"
    - Look at "what the code does" not "what it looks like"

- ## Why Behavioral Analysis?  Answer questions such as:
    - What domains and/or IPs are accessed via the network at run-time, and what is the access pattern?
    - How many entries in the address book are accessed?  Is each entry or some threshold number of entries accessed?  Are all or some significant number of entries accessed serially, or is the access pattern random?
    - Are the accessed address book entries sent over the network?  Is sensitive data or PII leaked?

# Behavioral & Signature-Based Detection: Working together

- Behavioral and signature-based detection are *complementary*
  - Behavioral detection on a client will suck battery life – should be done "in the cloud" / on a server
  - Behavioral analysis can provide signatures to client-side anti-virus

Cloud-Based
Behavioral Scanning

Client-Side
Signature Check

Auto-generate signatures



Total Infections Catalogued:
218588

# Principles for Mobile Malware Detection

- **Malware becoming ever more sophisticated and mutable → Must use behavioral detection approaches**
  - Malware variations growing exponentially
  - Easy for criminals to distribute malware
  - Combination of static/signature-based and behavioral approaches needed

- **Some combination of cloud-based analysis + client side protection necessary**
  - Limited resources (CPU, memory) on the endpoint to analyze content (URLs) and applications in real-time
  - Problem compounded on mobile devices
  - Perform analysis in the cloud; prevent and remediate on the client (and/or edge)

- **Take advantage of massive amounts of data**
  - Build libraries of millions of mobile apps over time, each behaviorally analyzed for malicious behavior
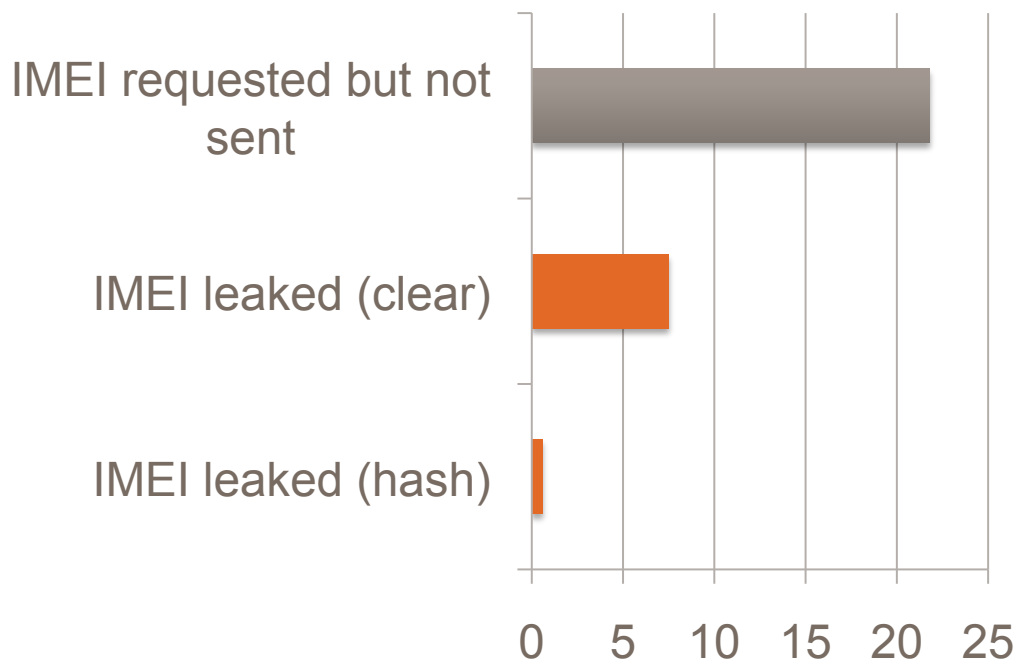
# Behavioral Study of 10K Android Apps

- 10K apps chosen at random from 30 different categories on Google Android Marketplace

- Analyzed in an emulator for 30 seconds

- Minimal simulated user interaction (although we could have done more)

- Found / Studied:
  - Privacy violations: IMEI / IMSI Leakage
  - Network access / Traffic patterns
  - Bandwidth usage
  - SMS activity
  - System call activity

# 8.4% of Android Apps Leaking Personal Information (IMEI)

**DASIENT**™
SMART WEB SECURITY

Only requested IMEI
Actually leaked IMEI

**Android Applications Requesting/Leaking IMEI**
Percent



IMEI requested but not sent

IMEI leaked (clear)

IMEI leaked (hash)

0   5   10   15   20   25

- Static analysis approaches would only identify that IMEI was requested; however, cannot tell if the IMEI was actually leaked
- Static analysis only could yield "false positives"
- Dasient has forensic evidence that the IMEI was leaked in 8.4% of the apps

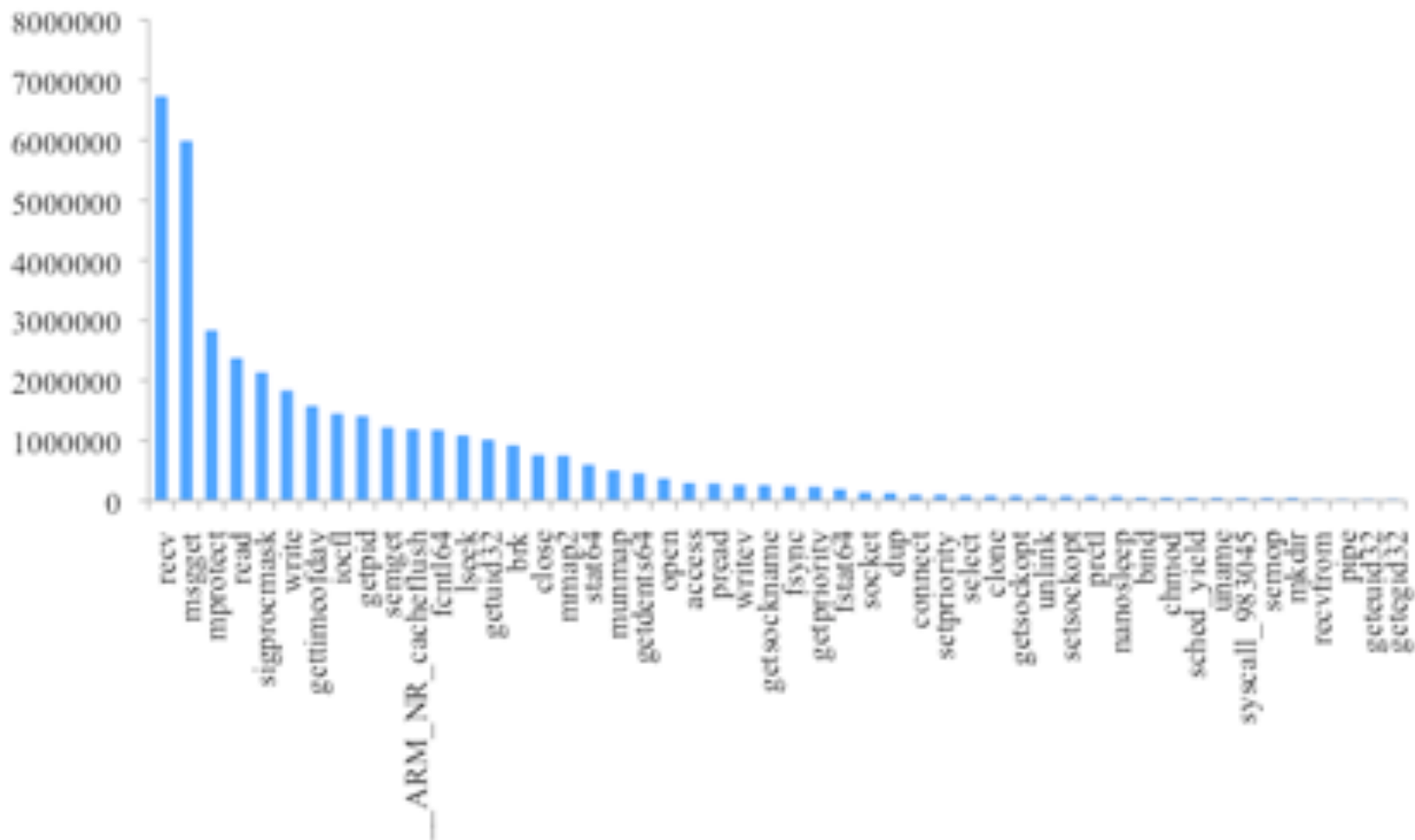Source: Dasient (n=10,000)

# Network Behavior

## Top 50 Domains Accessed by Android Apps
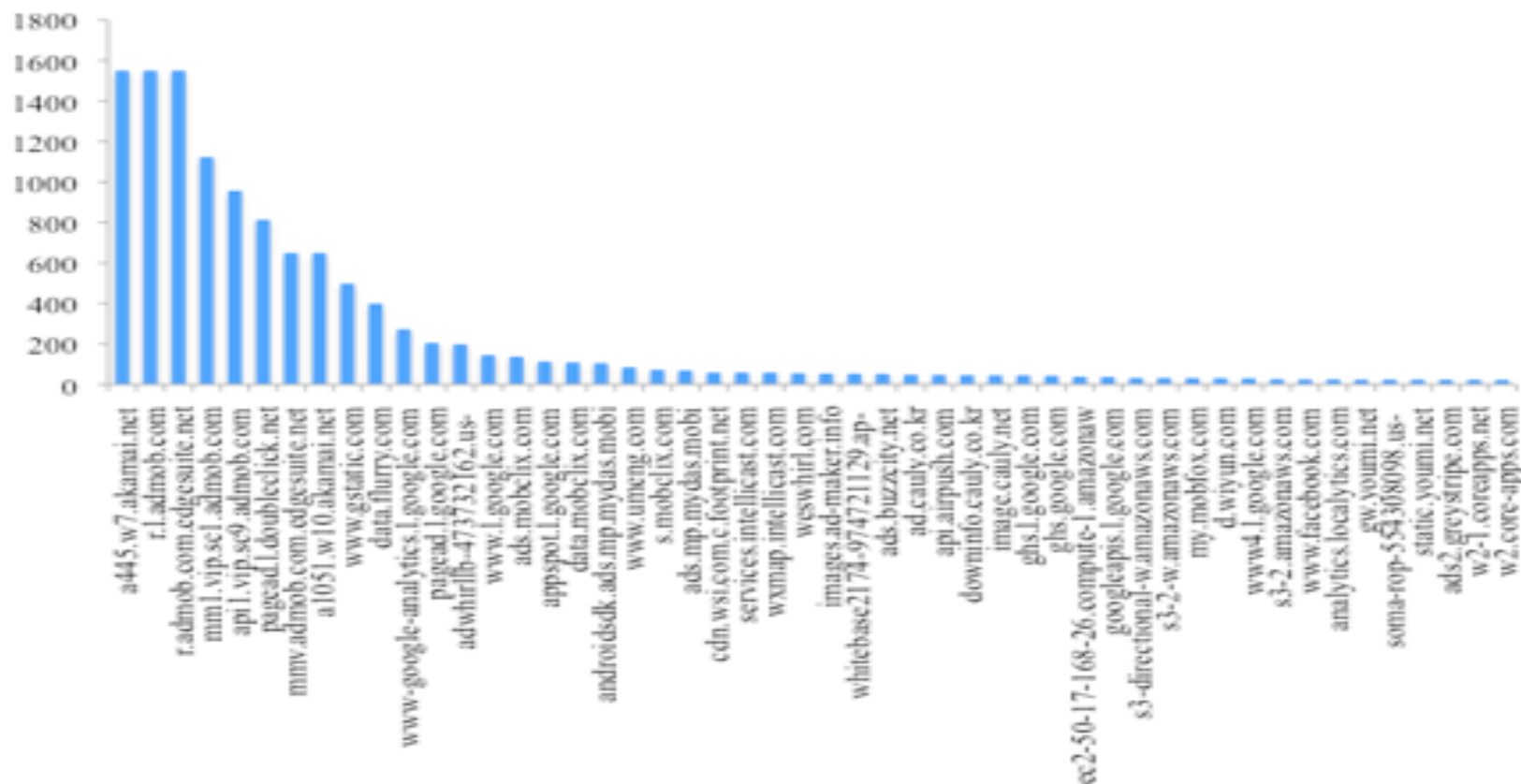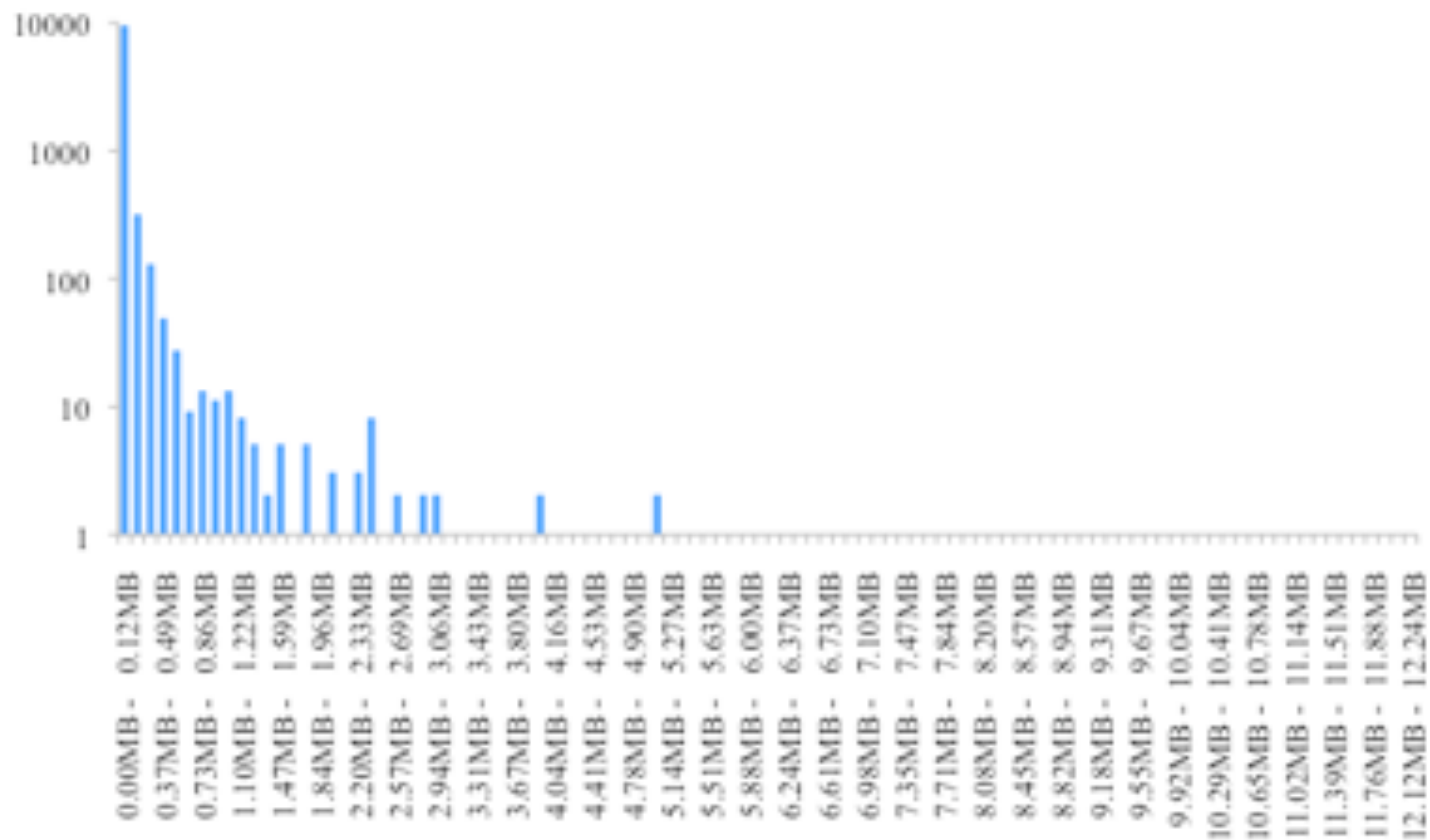


Source: Dasient (n=10,000)

# System Call Activity

**Top System Calls Made by Android Apps**



Source: Dasient (n=10,000)

# Bandwidth Usage

**Bandwidth Usage of Android Apps**



Source: Dasient (n=10,000)

# Ex. Behavioral Signals for Malware Detection

- Hundreds of these

- Example #1: Number of processes
  - Baseline: 58.3, Std Dev: 4.5
  - DroidDream: 660.3 (due to "rageagainstthecage"), Std Dev: 238.8

- Example #2: Undesired SMS activity

| Name | # of msgs | Number |
|------|-----------|--------|
| Fakeplayer.A | 3 | 3353, 3354 |
| Fakeplayer.B | 4 | 7132 |
| Fakeplayer.GEN | 4 | 7132 |
| Bgserv.A | 1 | 10086 |
| Twalkupi.A | # of contacts | contacts |

# Mobile Drive-bys

- No social engineering or user interaction required to compromise device; just visit a web page

- Smart phones are vulnerable, valuable, and use common software packages (e.g., Webkit)

- Prototype attack: Drive-by data ex-filtration – steal Skype Ims
  1. User visits compromised web page
  2. Exploit WebKit vuln CVE-2010-1807
  3. Attacker connects to device via backdoor
  4. Find Skype username
  5. Exploit Skype vuln CVE-2011-1717
  6. Traverse to user's Skype data directory
  7. Send files to attacker's machine
  8. Run SQLite locally & read user's convos

# Notification: Privacy Violation

## Ninja Slider
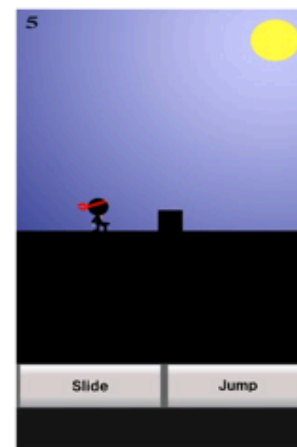Scanned on **July 21 2011** at **9:17 AM EST** from **Android App Store**

## Notification: Privacy Violation

🚩 **Application leaks MD5 hash of user's IMEI number in GET request**
(5284047f4ffb4e04824a2fd1d1f0cd62). Hashes of IMEI numbers are reversible (more information).

GET /sp/load_app_ads?
deviceOsVersion=2.3&deviceName=generic&sdkBuild=109&deviceFamily=google_sdk&deviceBrand=generic&
deviceClass=android&appIdentifier=cross.field.NinjaSlider&udid=**5284047f4ffb4e04824a2fd1d1f0cd62**&
deviceOsVersionFull=2.3.4&sdkVersion=1.2.2&callbackid=0&zid=NjY3OA%253D%253D%250A&adl_app_flg=1&
displaySize=480x800&displayDensity=1.5 HTTP/1.1
Host: sp.ad.adlantis.jp

🚩 **Domains / IPs accessed**
a1621.b.akamai.net                          80.150.193.42
i.adimg.net.edgesuite.net                   80.150.193.27
vatan.adlantis.jp                           59.106.159.228

🚩 **Related permissions**
android.permission.READ_PHONE_STATE

**Publisher**
**Cross Field Inc.**

**Version**
**1.2.7**

**Last Update**
**7/20/2011**

**Package name**
**cross.field.NinjaSlider**

**SHA1 Hash**
22eb1493275efdbff68e8bab58af1e3d60f03fcc

**Screenshot**

# Alert: Malware (DroidDream)

**DASIENT**™
SMART WEB SECURITY

## Bowling Time

Scanned on **March 10 2011** at **2:58 PM EST** from **Android App Store**

⚠ **Attempt to take root access via rageagainstthecage exploit**

🚩 **Direct access to IP: 184.105.245.17**

🚩 **Domains / IPs accessed**
184.105.245.17
a96-17-109-112.deploy.akamaitechnologies.com
api1.vip.sc9.admob.com
adwhirllb-473732162.us-east-1.elb.amazonaws.com

🚩 **Permissions Requested**

| Permission | | Description |
|---|---|---|
| ACCESS_WIFI_STATE | View Wi-Fi status | Allows an application to view the information about the status of Wi-Fi. |
| CHANGE_WIFI_STATE | Change Wi-Fi status | Allows an application to connect to and disconnect from Wi-Fi access points and to make changes to configured Wi-Fi networks. |
| INTERNET | Full Internet access | Allows an application to create network sockets. |
| READ_PHONE_STATE | Read phone state and identity | Allows the application to access the phone features of the device. An application with this permission can determine the phone number and serial number of this phone, whether a call is active, the number that call is connected to and so on. |

🚩 **Public Activities Without Permissions (1)**
com.android.root.main

**Publisher**
**Kingmall2010**

**Version**
**(Unknown)**

**Last Update**
**(Unknown)**

**Package name**
**com.droiddream.bowlingtime**

**SHA1 Hash**
72adcf43e5f945ca9f72064b81dc0062007f0fbf

**Screenshot**

# Summary + Q&A

- Significant growth in mobile malware in past two years
- Behavioral analysis can complement static analysis to identify mobile malware
- Studied 10K Android apps + malware
  - Identified IMEI / IMSI leaks
  - Undesired SMS
  - Identified behavioral signals of malware
- Prototyped mobile drive-by attack


- We're Hiring!  Contact us at careers@dasient.com