



# ANDROID

## Mobile Security NIS'11

Nick Krlevich  
<nnk@google.com>  
Android Security Team  
6.29.2011



# Overview



- Why is mobile security important?
- What is Android?
- What are the risks?
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- Android Security Case Studies
  - Example: DroidDream
  - Example: Auth Token Leakage
- Closing / Q&A

# The Big Picture - Worldwide View



World's Population

**7.0 BILLION**<sup>[1]</sup>

Internet Enabled PCs

**OVER 1.7  
BILLION**<sup>[2]</sup>

Mobile Phones in Use

**OVER 5  
BILLION**<sup>[1]</sup>

Sources:

[1] [http://en.wikipedia.org/wiki/List\\_of\\_countries\\_by\\_number\\_of\\_mobile\\_phones\\_in\\_use](http://en.wikipedia.org/wiki/List_of_countries_by_number_of_mobile_phones_in_use)

[2] <http://www.gartner.com/it/page.jsp?id=703807>

# The Big Picture - US Snapshot



TODAY:

**91%**

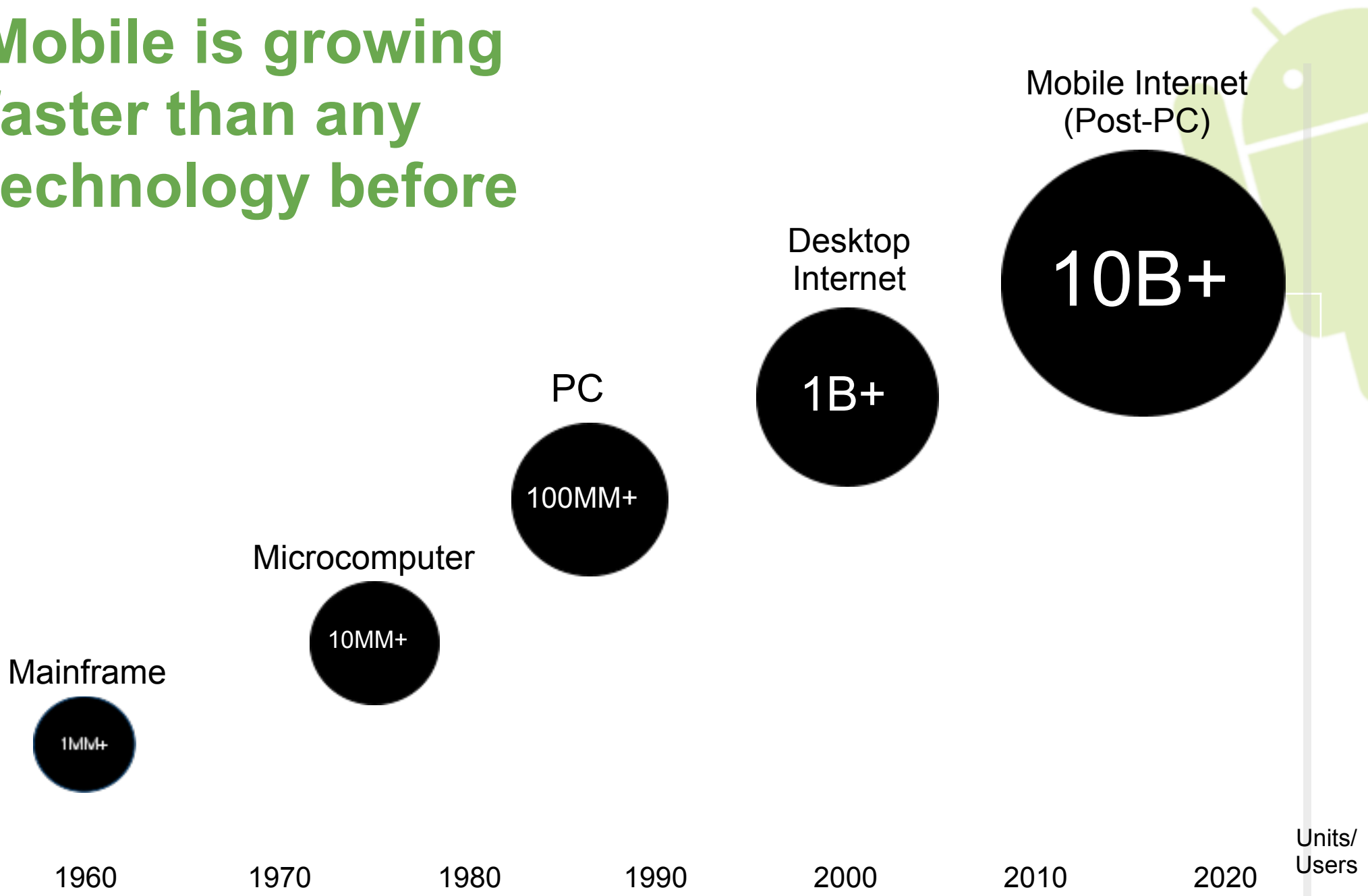
of Americans use a mobile phone

BY CHRISTMAS 2011:

**50%**

of Americans will have a smart phone

# Mobile is growing faster than any technology before



# Mobile Evolution



Not only is mobile growing fast, it's evolving fast.

- 18 month average replacement rate
- There are now more Internet enabled phones than PCs
- OS updates traditionally released one to two times per year
- Most new users have never used a smartphone

Result: Technology is changing faster than many people's ability to absorb the changes.

The fact of the matter is that mobile devices are going to be the majority of the way that people get, manage, and store information.



## What It Means to Security



Mobile devices are major security targets.

# Overview



- Why is mobile security important?
- **What is Android?**
- What are the risks?
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- Android Security Case Studies
  - Example: DroidDream
  - Example: Auth Token Leakage
- Closing / Q&A

# Android at Its Core



## The Platform

- Linux based, free, open source mobile platform
  - Source code at <http://source.android.com>
- Any handset manufacturer or hobbyist can install
- Any developer can use
  - SDK at <http://developer.android.com>
- Can be found on 300+ smartphones and tablets today
- Empowers users and developers

# Android Security – Open Design



*Open design: The design should not be secret. The mechanisms should not depend on the ignorance of potential attackers, but rather on the possession of specific, more easily protected, keys or passwords.*

J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems", pp. 1278-1308, Proceedings of the IEEE 63, number 9, September 1975

# The Android Ecosystem



Over 100  
Million  
Devices  
Activated

Over  
200,000  
apps on  
Android  
Market

Most  
Popular  
Platform

36% of the  
Smartphone  
Market<sup>[1]</sup>

[1] Q1 2011 numbers

Source: [http://news.cnet.com/8301-13506\\_3-20064223-17.html](http://news.cnet.com/8301-13506_3-20064223-17.html)

# Overview



- Why is mobile security important?
- What is Android?
- **What are the risks?**
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- Android Security Case Studies
  - Example: DroidDream
  - Example: Auth Token Leakage
- Q&A

# ENISA - Top 10 Smartphone Risks



1. Data leakage resulting from device loss or theft
2. Unintentional disclosure of data
3. Attacks on decommissioned phones
4. Phishing Attacks
5. Spyware Attacks
6. Network Spoofing Attacks
7. Surveillance Attacks
8. Diallerware Attacks
9. Financial Malware Attacks
10. Network Congestion

# ENISA - Top 10 Smartphone Risks



1. Data leakage resulting from device loss or theft
2. Unintentional disclosure of data
3. Attacks on decommissioned phones
4. Phishing Attacks
5. Spyware Attacks
6. Network Spoofing Attacks
7. Surveillance Attacks
8. Diallerware Attacks
9. Financial Malware Attacks
10. Network Congestion

# ENISA - Top 10 Smartphone Risks



1. Data leakage resulting from device loss or theft
- 2. Unintentional disclosure of data**
3. Attacks on decommissioned phones
4. Phishing Attacks
5. Spyware Attacks
6. Network Spoofing Attacks
7. Surveillance Attacks
8. Diallerware Attacks
9. Financial Malware Attacks
10. Network Congestion

# ENISA - Top 10 Smartphone Risks



1. Data leakage resulting from device loss or theft
2. Unintentional disclosure of data
3. Attacks on decommissioned phones
4. Phishing Attacks
5. Spyware Attacks
6. Network Spoofing Attacks
7. Surveillance Attacks
8. Diallerware Attacks
9. Financial Malware Attacks
10. Network Congestion

# ENISA - Top 10 Smartphone Risks



1. Data leakage resulting from device loss or theft
2. Unintentional disclosure of data
3. Attacks on decommissioned phones
4. Phishing Attacks
5. Spyware Attacks
6. Network Spoofing Attacks
7. Surveillance Attacks
8. Diallerware Attacks
9. Financial Malware Attacks
- 10. Network Congestion**

# Overview



- Why is mobile security important?
- What is Android?
- What are the risks?
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- Android Security Case Studies
  - Example: DroidDream
  - Example: Auth Token Leakage
- Closing / Q&A

# Addressing the risks - Android Security



- **Prevent** security issues from occurring
  - Design reviews
  - Code audits
- **Minimize** the impact of a security issue
  - Application Sandbox
  - Permissions
- **Detect** vulnerabilities and security issues
  - Manual
  - Automated
- **React** to vulnerabilities and security issues swiftly
  - Application and platform autoupdates
  - Remote application removal

## Minimize – Least Privilege

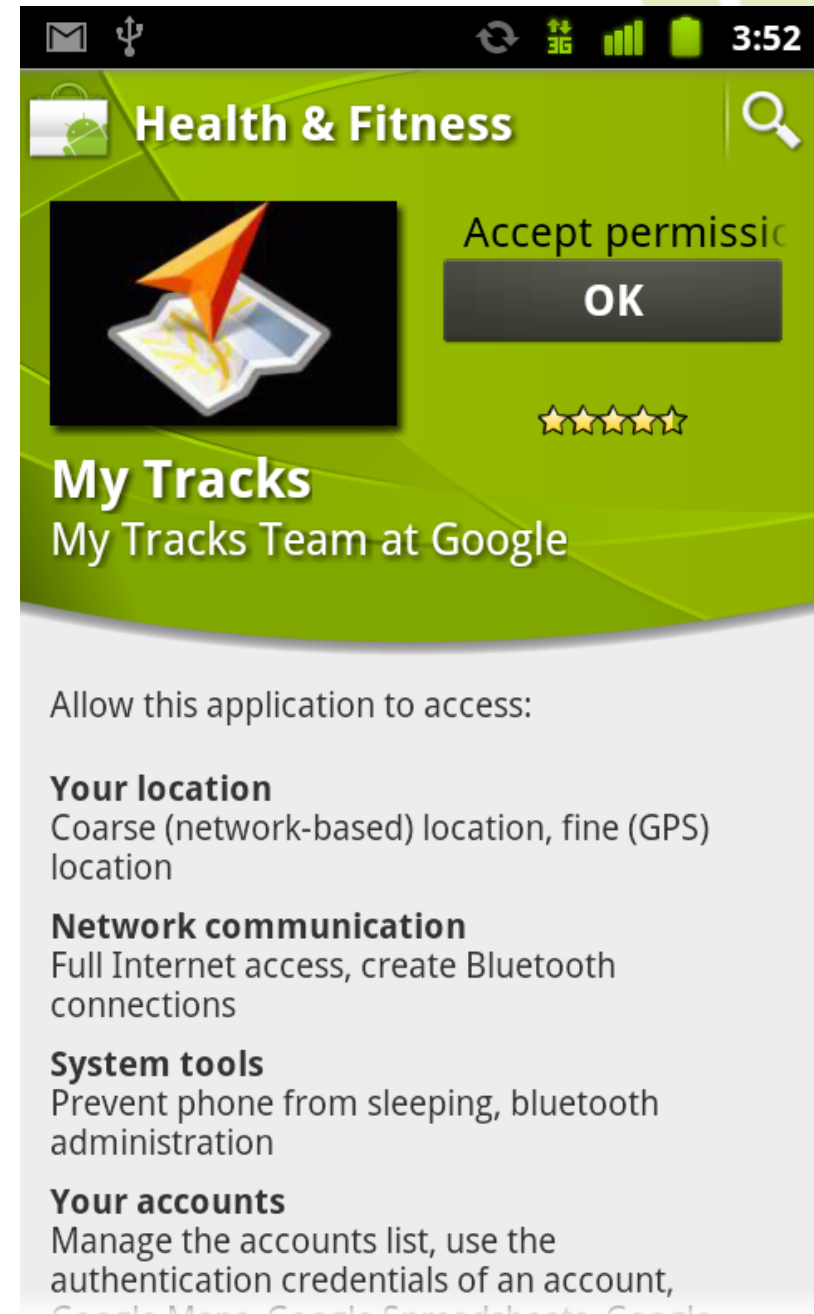


*"Every program and every user of the system should operate using the least set of privileges necessary to complete the job."*

J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems", pp. 1278-1308, Proceedings of the IEEE 63, number 9, September 1975

# Minimize – Permissions

- Whitelist model
  1. Default Deny
  2. User approved exceptions
- Ask users fewer questions
- Make questions more understandable
- ~200 permissions
  - More ⇒ granularity
  - Less ⇒ understandability



## **Minimize** – Separation of Privileges

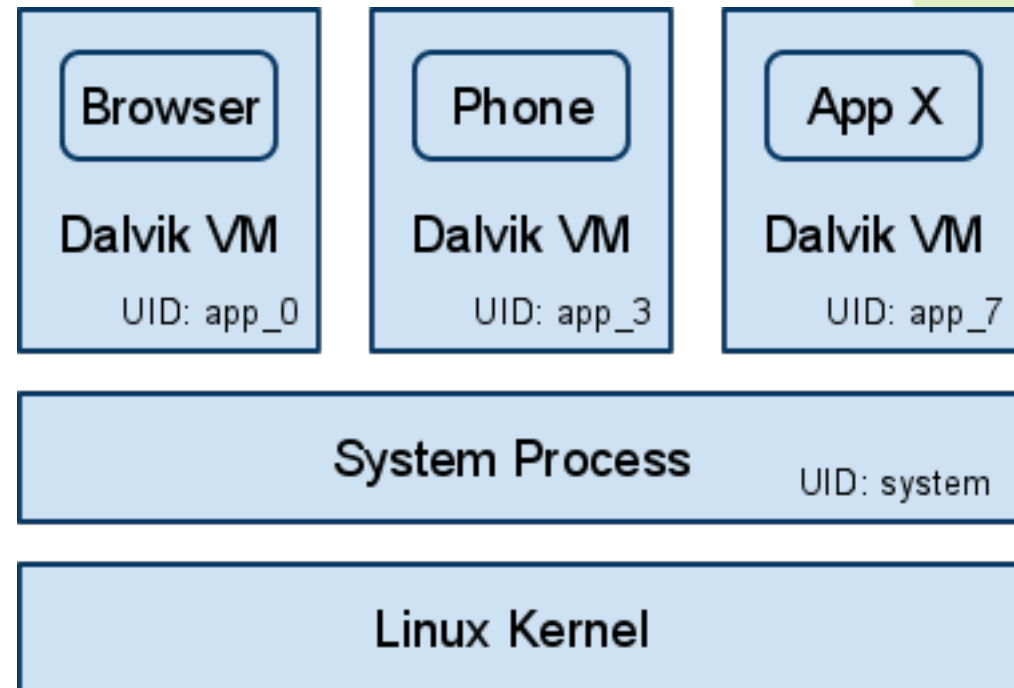


*"The principle of separation of privilege states that a system should not grant permission based upon a single condition"*

# Minimize – Application Sandbox



- Each application runs within its own UID and VM
- Default privilege separation model
- Instant security features
  - Resource sharing
    - CPU, Memory
  - Data protection
    - FS permissions
  - Authenticated IPC
    - Unix domain sockets
- Place access controls close to the resource, not in the VM



## **Prevent** – Device Administrator Interface



*"Human beings, who are almost unique in having the ability to learn from the experience of others, are also remarkable for their apparent disinclination to do so."*

-- Douglas Adams

# Prevent – Device Administrator Interface



## Capabilities:

- Remotely wipe all data from lost or stolen mobile devices
- Lock idle devices after inactivity or immediately.
- Enforce minimum password requirements.
- Require device encryption
- **More to come**

## Users:

- Google Apps Premier and Educational Editions
- Exchange ActiveSync
- Third party apps on Android Market such as anti-virus software

<http://googleenterprise.blogspot.com/2010/10/bring-your-phone-to-work-day-managing.html>

<http://developer.android.com/guide/topics/admin/device-admin.html>

**React** – Over the air updates



*"Autoupdaters are the  
best security tool since  
Diffie-Hellman"*

-- Rich Cannings  
Android Security Team

# React – Over the air updates



- Every modern operating system should be responsible for:
  - Automatically updating itself
  - Providing a central update system for third-party applications
- What does Android do?
  - Auto update capability is baked into the platform.
  - This capability is available not just to Google but other partners.

## **Detect** – Android Market



*"Keep your friends close,  
and your enemies closer"*

*Sun-tzu*

*Chinese general & military strategist (~400 BC)*

# Detect – Android Market



## Core Principle:

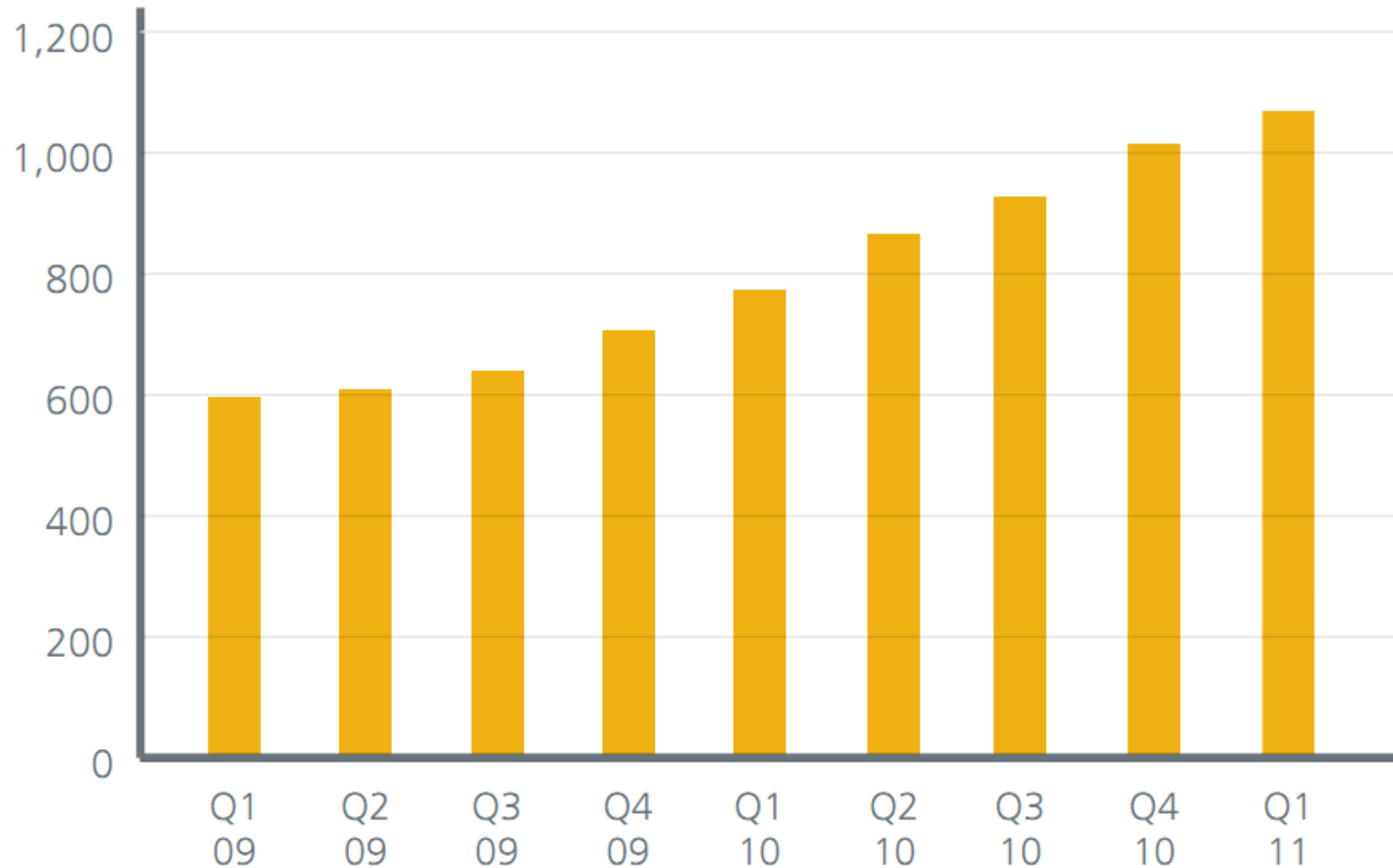
- Protect Android Users while keeping Android Market open

## Multiple layers of protection:

- One central point for all Android applications
- Risk Analysis for developers
- Manual suspension of applications and developers
- Static Analysis for malware and other risk signals
- Dynamic Analysis
- Remote app uninstall to remove apps
- Ability to push an automated cleanup tool if necessary

# Mobile Malware – Overview

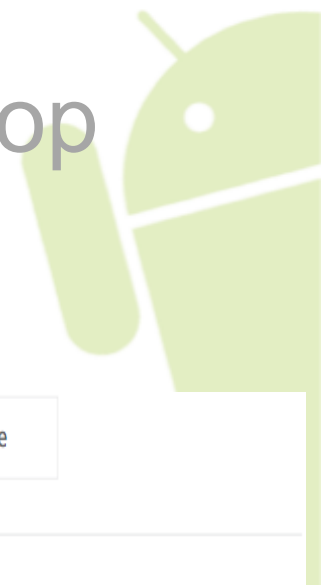
## Total Mobile Malware Samples Across All Mobile Platforms



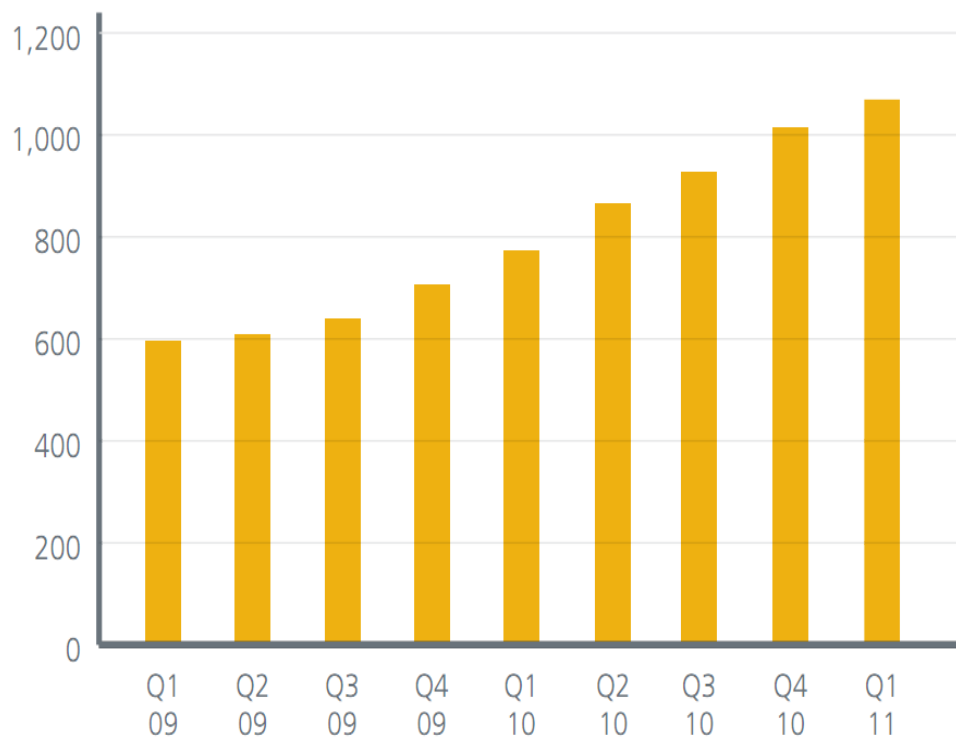
<http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2011.pdf>



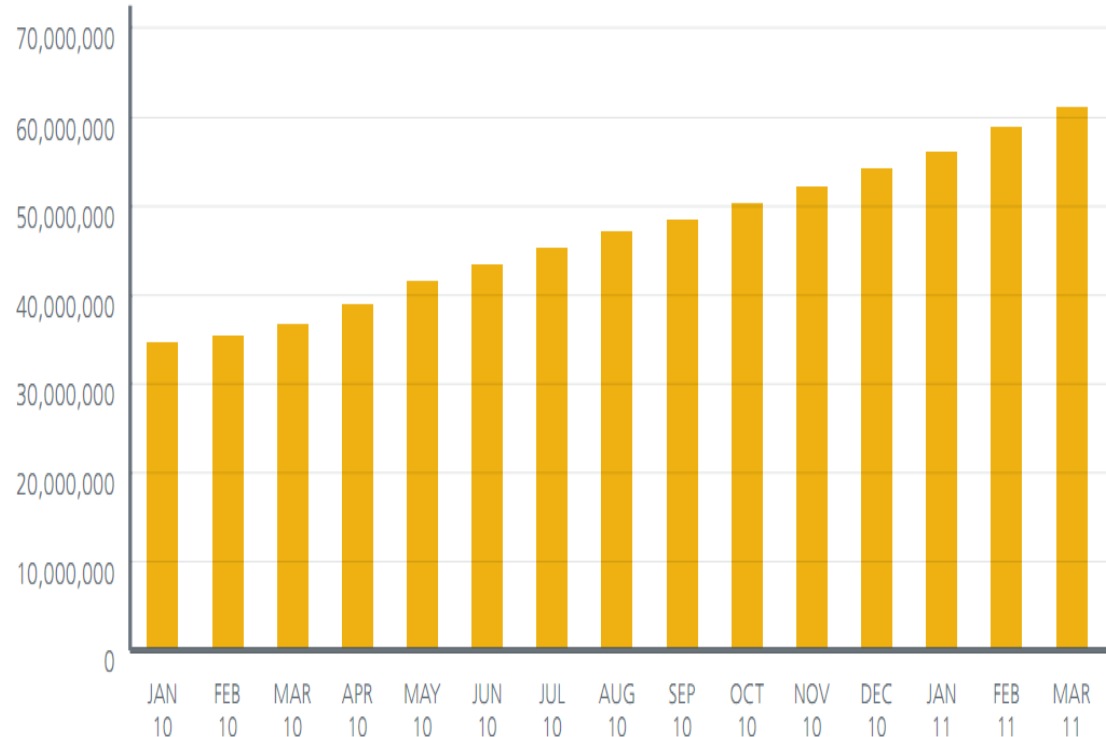
# Mobile Malware – Comparison to Desktop



Total Mobile Malware Samples



Total Malware Samples in the Database

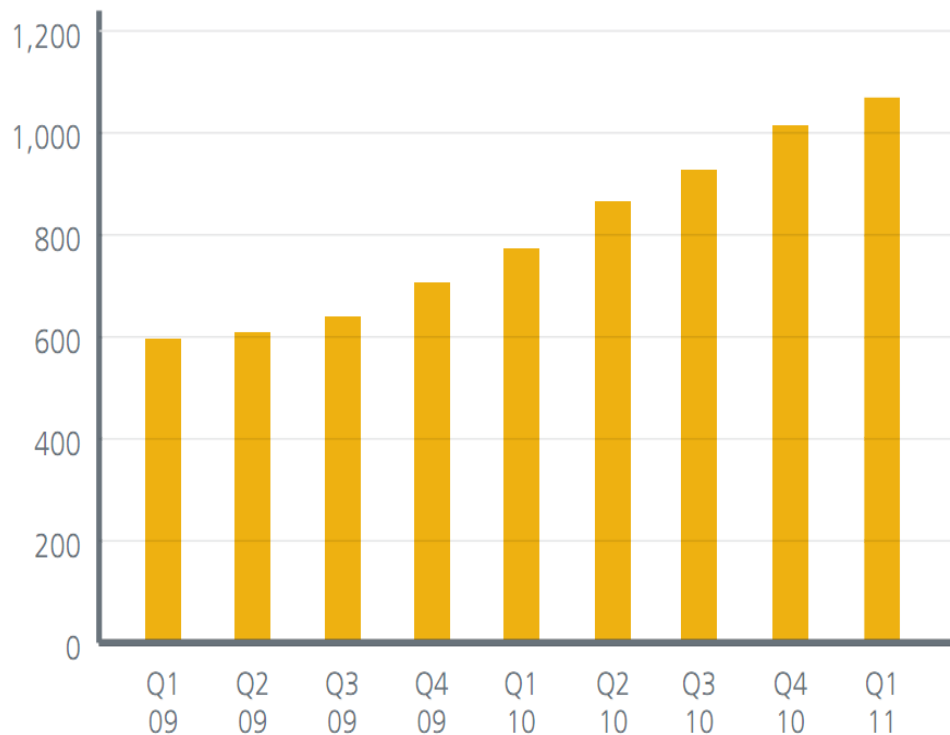


<http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2011.pdf>

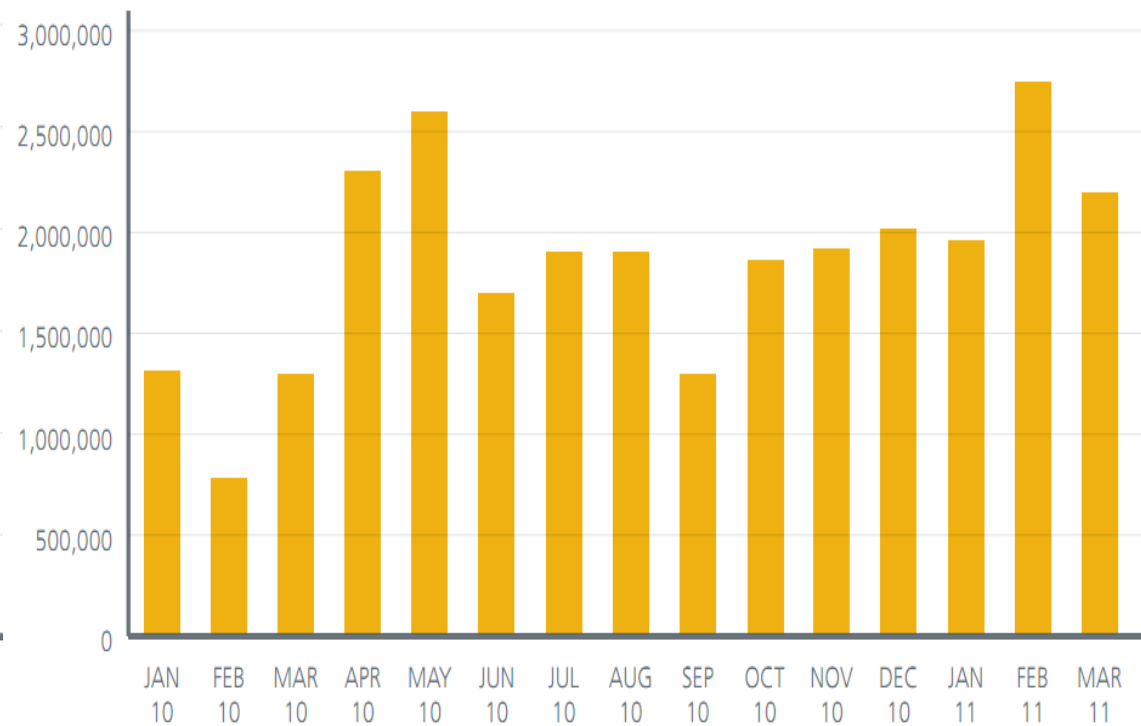
# Mobile Malware – Comparison to Desktop



Total Mobile Malware Samples



New Malware Samples Added to the Database, by Month



<http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2011.pdf>

# Mobile Malware



- Application sandboxing will provide protection against malicious applications.
- Application marketplaces will provide a central point where malware can be monitored and removed safely.
- Press attention today is helping raise awareness of mobile malware before it becomes a significant problem.

# Overview



- Why is mobile security important?
- What is Android?
- What are the risks?
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- **Android Security Case Studies**
  - Example: DroidDream
  - Example: Auth Token Leakage
- Closing / Q&A

# Android Malware – DroidDream



- Malware added to alluring apps
- Identified March 1, 2011
- Command and control down March 2, 2011 (**1 day**)
- Cleanup started March 5, 2011 (**4 days after detection**)
- Affects Android 2.2.1 and earlier
- Uses known, fixed exploits to gain root
- Limited information stolen: IMEI, MEID, Device Model, SDK version
- 270,000 installs
- Installs rootkit and command & control system

```
$ ls -l /system/bin/profile /system/app/DownloadProvidersManager.apk
-rwsr-xr-x root root 3868 2011-03-08 23:40 profile
-rw-rw-rw- root root 14077 2011-03-08 23:40 DownloadProvidersManager.apk
```

- Userland Rootkit
  - Persisted across factory resets
  - Could not be removed with remote application uninstall

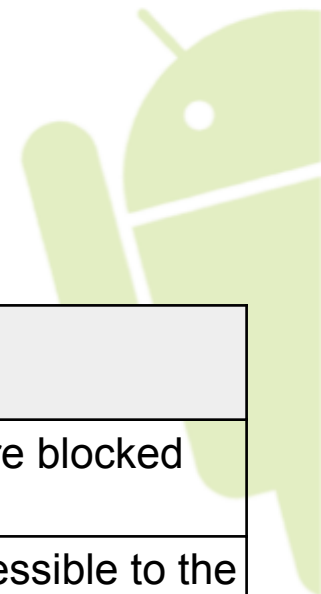
# Defense In Depth



*"The principle of defense-in-depth is that layered security mechanisms increase security of the system as a whole. If an attack causes one security mechanism to fail, other mechanisms may still provide the necessary security to protect the system."*

Source: [https://www.owasp.org/index.php/Defense\\_in\\_depth](https://www.owasp.org/index.php/Defense_in_depth)

# Android Malware – DroidDream



| Layer | Security Control           | On Device | Effective? | Result   |
|-------|----------------------------|-----------|------------|--|
| 1     | Account Risk Analysis      | No        | Somewhat   | Some accounts were blocked                               |
| 2     | Market Content Review      | No        | No         | Applications were accessible to the public               |
| 3     | Permissions                | Yes       | No         | No unusual permissions requested                         |
| 4     | Application Sandbox        | Yes       | Somewhat   | Patched devices were protected from DroidDream's rooting |
| 5     | Incident Response Process  | No        | Yes        | 1 hour from public notice until Google removes apps      |
| 6     | Market Takedown            | No        | Yes        | No further infections possible                           |
| 7     | Remote Application Removal | Yes       | Somewhat   | Patched devices could be cleaned up successfully         |
| 8     | Cleanup Tool               | Yes       | Yes        | Remaining devices were cleaned                           |
| 9     | System Updates             | Yes       | Ongoing    | Security patches continue to be delivered to users.      |

# Android Vulnerability – Auth token leakage



- Announced May 13th, 2011
- Vulnerability
  - Device to server communication over HTTP, not HTTPS
  - Authorization Tokens sent in clear text
  - Very similar to other "Firesheep" like attacks
- Mitigating Factors
  - Authorization Tokens time limited
  - Attack required physical proximity (wifi snooping)
- Fixes
  - Server side fix completed May 20th, 2011
  - Picasa client patch checked in May 24th, 2011

# Overview



- Why is mobile security important?
- What is Android?
- What are the risks?
- Addressing the risks: Android security
  - Overview
  - Application Sandbox and Permissions
  - Auto-updates
  - Android Market
  - Device Administrator Interface
- Android Security Case Studies
  - Example: DroidDream
  - Example: Auth Token Leakage
- **Closing / Q&A**

# Closing – Lessons



- Mobile Security is a new and growing risk.
- Like all risks, it needs to be understood and managed.
- No security solution can be 100% effective.
- The risks will never go away.
- Android grew up in the Internet age, and developed tools to help reduce risks to users and enterprises.
- How we manage risk will ultimately determine how secure we are.

## Questions?

Please join Giles Hogben and myself today from 2:00-3:15 for "Securing the Road Warrior - mobility, walled gardens, consumerisation of IT"

We're hiring! Email [nnk@google.com](mailto:nnk@google.com)

Security Contact: [security@android.com](mailto:security@android.com)



# Copyrights and Trademarks

- Android, Google are registered trademarks of Google Inc.
- All other trademarks and copyrights are the property of their respective owners.

