Smartphone Security
Limitations: Conflicting Traditions

Nathaniel Husted <nhusted@indiana.edu>
Hassen Saidi <saidi@csl.sri.com>
Ashish Gehanni <ashish.gehani@sri.com>
Smartphones

- Cellphone
- Sensor Platform
- Always on Internet
- Communicators
- Cognitive Extensions
Android

- Google’s Smartphone Operating System (OS)
- “Open Source” (but Closed Platform)
- 38% worldwide marketshare, 50% in 2012 [Gartner01]
Android Malware

- Most targeted platform
- “Rooting” Attacks
- Information Theft
- Fraudulent Texts
- Denial-of-Service
- And there’s nothing we can do...
Security Limitations Caused By Conflicting Traditions

Cellular Tradition VS. General Computing Tradition
Overview

❖ What are these traditions?
❖ How are they represented on a smartphone?
❖ Why are they conflicting?
❖ Why are they limiting our security?
❖ What can be done about it?
General Computing Tradition

- General Computing
  - Open
  - Free as in Freedom
  - Modular
  - **Owner Governed**
Cellular Tradition

- Cellular
  - Closed
  - Stable
  - “Undocumented”
  - Provider Governed

The Smartphone Platform

- Operating System
- WiFi Camera Camcorder
- General Purpose CPU
- Computing Environment
- Baseband Chip
- Cellular Environment
But Isn’t Android Open?

What Do We Mean by Open?
An Open Platform

1. A complete set of source code is required to run all features of the platform must meet the Open Source Definition.
An Open Platform

2. The software must include at least a minimal set of build and use instructions
An Open Platform

3. The device owner must be able to modify the software on the device without violating warranties, use agreements, software controls, or hardware controls.
An Open Platform

1. A complete set of source code is required to run all features of the platform must meet the Open Source Definition

2. The software must include at least a minimal set of build and use instructions

3. The device owner must be able to modify the software on the device without violating warranties, use agreements, software controls, or hardware controls.
Android’s Openness

1. A complete set of source code is required to run all features of the platform must meet the Open Source Definition

Mostly

Drivers and some low level source code is closed
Android’s Openness

2. The software must include at least a minimal set of build and use instructions

Yes!

Full range of documentation
Android’s Openness

3. The device owner must be able to modify the software on the device without violating warranties, use agreements, software controls, or hardware controls.

No!

Violate warranties, service agreements, or destroy the device
## Android's Openness

<table>
<thead>
<tr>
<th>Requirement 1:</th>
<th>Mostly</th>
<th>Drivers and some low level source code is closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement 2:</td>
<td>Yes!</td>
<td>Full range of documentation</td>
</tr>
<tr>
<td>Requirement 3:</td>
<td>No!</td>
<td>Violate warranties, service agreements, or destroy the device</td>
</tr>
</tbody>
</table>
Security Limitations
How do we ease the conflict?

❖ Compromise?
❖ Incremental Provider Updates
❖ Mobile Hypervisors
Incremental Provider Updates
Incremental Provider Updates
Incremental Provider Updates
Incremental Provider Updates
Mobile Hypervisors

The Hypervisor Platform

- Operating System
- Computing Environment

- Hypervisor
- Phone Application
- CPU
- WiFi
- Camera
- Camcorder

- Baseband Chip
- Cellular Environment
Conclusion

❖ Smartphones are here to stay
❖ They are more than just phones
❖ Security on smartphones is limited by governance conflicts
❖ OS Developers, Phone Manufacturers, and Cellular Providers can and must help
❖ *Think Open Platform!*
QUESTIONS?
References