Android Malware and OS Security

Ian Denhardt

March 5, 2012

Mobile malware is on the rise...

- 155% increase in mobile malware (all platforms) from 2010 - 2011
- Not nearly as bad as the situation with desktop computers
- What can be done from a mobile phone OS standpoint to keep it under control?
- What is being done?

Quick Overview : What does the hardware give us?

Most platforms that run full operating systems provide at least:

- Two priviledge levels, kernel and user-mode.
- Some form of memory protection, typically paging, which allows us to remap (or unmap) particular regions of memory if we're in kernel-mode.
- an instruction that causes a particular interrupt, that can be used from user-mode (often called syscall)
- A timer.

This is enough to sandbox a user-level application:

- We can map in only the regions of memory we want to provide access to. This covers devices too, since most devices are accessed via memory mappings.
- The only way the user process can access anything else is through the syscall instruction, which calls the kernel's interrupt handler.
- A process can't hog the cpu either; we can set the timer before executing user code, and when it goes off, we will regain control.

Android Security Model

Android runs atop a Linux kernel, so it uses Linux's security primitives.

- Each process is assigned a user
- Users may be members of groups
- One user (root) is special, and is allowed access to everything on the system
- Other users are subject to permissions checks; to write to the sdcard, I might need to be a member of the "sdcard_rw" group, but perhaps anyone can read from it.

On standard Linux, users are human beings, but typically only one person uses a mobile device!
Instead, android maps users to applications - If your web browser isn't in the sms group, it can't send expensive texts when compromised.

Installation

- When an android user is prompted to install an application, they are presented with a list of permissions, each of which corresponds to a group.
- If they approve, a Linux user is created (e.g. app_42) and it is added to the corresponding groups.

This should provide some limits on what a malicious app can do; If your keyboard app can't access the internet or any storage, it probably can't be a very effective keylogger.

In practice...

Various issues in practice:

- Device owners can't deny individual groups
- Too many apps ask for far more than they need (quite a few keyboards have full network access).
- Google bills the app market as a good place to get software, but plenty of malware finds its way in (and this is where most of it comes from)
- Apps that come with the phone can't be removed, some of them arguably malware themselves.
- Exploits of the kernel are always possible.

References

- John Leyden (The Register) "Cops cuff premium-rate SMS Android malware suspects" http://www.theregister.co.uk/2012/02/28/french_android_malware_arrests/
- Juniper Networks, "2011 Mobile Threat Report" http://www.juniper.net/us/en/local/pdf/additional-resources/jnpr-2011-mobile-threats-report.pdf?utm_source=promo&utm_medium=right_promo&utm_campaign=mobile_threat_report_0212
- Android Security Overview, http://source.android.com/tech/security/index.html