Received(Date): Sun, 31 Dec 2006 17:51:07 -0800
From: meriko borogove [redacted]
Subject: p2 security & third party code
To: Scott Forstall [redacted]
Cc: Henri Lamiraux [redacted], John Wright [redacted],Kim Koefoed Vorrath [redacted] meriko borogove [redacted] Dallas De Atley [redacted] Peter Graffagnino [redacted]
Date: Sun, 31 Dec 2006 17:51:07 -0800

Hi Scott -

Sounds like security is on everyone's mind again. Here's a brief overview of the security of the stack, and why it's going to be very difficult for third parties to install their code on a P2. I think it might be helpful to sit down with Dallas for a few minutes - let me know if you'd like to do that so you can ask all your questions in realtime - we'll be around!

So, our security model around preventing third parties from installing and running their own code boils down to our chain of trust around executing code, and the separation of excecutible code from user-modifiable data.

To start: we will sign all code that runs on the device in Cupertino at build time with the Apple CA. We'll use this signature to verify that code comes from Apple.

The SecureROM will only load code that comes from Apple. The bootloader will also only load a kernel that comes from Apple. Thus, all access to the hardware is controlled by known Apple code. We have disabled target disk mode, preventing users from copying arbitrary code into the system partition.

The barrier between the data partition and the system partition provides protection against code downloaded by Mail, Safari, or other network-enabled application. The system partition is mounted as read-only, and the kernel will not allow the system to execute code from the data partition. Applications are effectively "quarantined" by restricting their write access to the user's home directory, which is on the data partition.

Additionally, there's no general purpose install mechanism on P2 available to our users. Software Restore will require special pieces from B&I (also signed) which are not distributed with P2, and

software update has been designed such that the updater mechanism is included as part of the signed payload -- preventing someone from spoofing an update payload to an updater running on the device.

Before GM, we'll tighten the noose even further; none of our code will run as root, and we'll be removing 'dangerous' system calls that lead to common exploits (like buffer overflows).

Questions? Let us know.
meriko

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>date-sent</key>
    <real>1167616267</real>
    <key>flags</key>
    <integer>33815681</integer>
    <key>original-mailbox</key>
    <string>imap://deatley@mail.apple.com/INBOX</string>
    <key>sender</key>
    <string>meriko borogove <shock@apple.com></string>
    <key>subject</key>
    <string>p2 security & third party code</string>
    <key>to</key>
    <string>Scott Forstall <forstall@apple.com></string>
</dict>
</plist>