iPad Forensic data acquisition: Grasping into the black box

Joan Arnedo-Moreno, Luis Gómez-Miralles

KISON Cryptography and Information Security for Open Networks

Universitat Oberta de Catalunya (UOC)

Internet Interdisciplinary Institute (IN3)

July 10th, 2012

Introduction

Starting point: Mobile/Portable devices

- Exponential growth in last decade.
- Nowadays portable computers at our fingertips, capable of almost anything.

Introduction

Evolution of iOS devices in the last years

- 2007: iPhone and iPod Touch. Multi-touch interface, finger-based (no stylus).
- 2008: iPhone O.S. 2.0. App Store. Developers start coding applications extending the device capabilities.
- 2010: iPad, the rebirth of tablets. iPhone O.S. goes 'iOS'.
- 2011: iPad 2. iPhone 4S, Siri voice assistant.
- 2012: Siri comes to the iPad.

"Apple now gets about two-thirds of its revenue from iOS devices - a platform that didn't exist 5 years ago" *(Source: Augustine Fou, Marketing Science Consulting Inc.)*

Motivation

Regardless of debates, there are MANY iOS devices out there

- An iPad can replace a computer, or act as a glorified iPhone.
- A lot of personal data goes through the device.
- App based model (vs browser based).
- Closed system with undisclosed capabilities.

Who knows what's in there? iPad forensics!

Goals

Study how to image user data

- Look for a fast and open procedure.
- Changes to device state must be clear.

Looking where nobody else has looked yet: AirPrint

- How does it work? No technical documentation!
- Reliably acquiring forensic traces.

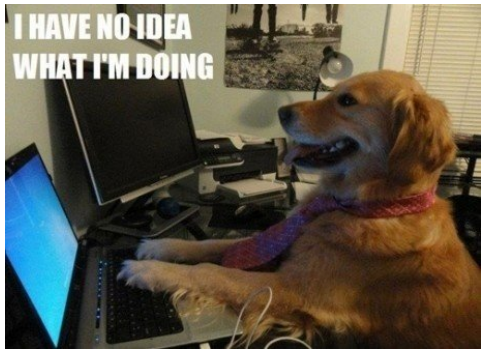
We always assume the role of the "good guys´´ (for real!).

Approaches to iOS forensic acquisition (1/2)

Using iTunes

- Extract iTunes backups from the owner's computer.
- (Optional) Force the device back up to another computer.
- Analyze data contained in these backups.
- Caveat: Misses many things, including unallocated (deleted) data.

Very simple approach, anybody can do it at home...



Approaches to iOS forensic acquisition (2/2)

More complex. They require overriding device controls.

Zdziarski public method

- Relies on device jailbreaking.
- Versatile, but wi-fi throughput is 1 MB/s (about 5-20 hrs).

Proprietary methods by some software vendors

- Rely on very specific vulnerabilities (but no actual jailbreak).
- Allow USB connection with computer.
- Limited to specific iOS version.
- Undocumented, paid, closed-source and/or restricted access.

Our approach

The best of both worlds: The Camera Connection Kit

- Intent: Photo transfer for dummies via Picture Transfer Protocol.
- Hidden feature: implements USB mass storage device class protocol!
- Open USB imaging method which only relies on jailbreak.
- Cheap accessory: about $30.

Image dump throughput: about 29 Mb/s (connector limit).

Step by step process: Device setup

This method works on any device which may be jailbroken.

- Device jailbreak: bypassing vendor restrictions.
  - Jailbreak is usually released days-weeks after every iOS update.
  - Latest iOS (5.1.1) untethered jailbreak: Absinthe 2.0.2.
- Use Cydia to install *openssh* and *coreutils*.
- Connect the iPad to a network and connect to it using SSH.

No need for remote host if necessary, but process may leave traces.


Step by step process: Device imaging

- Plug the Camera Connection Kit.

- Add some USB or SD storage formatted as HFS+.
  - Also works with FAT filesystems.
  - HDD storage requires external power.

- Mount the Camera Connection Kit media with right params.

- Use "dd" to image `/dev/rdisk0s2` to external storage.

- Unmount media.

Now what?

- SQLite databases: address book; call history; SMS messages; e-mails; etc.

- Property lists (`.plist`): e-mail accounts; web browser bookmarks and cookies; locations (Maps) bookmarks and history of lookups; etc.

- Browse: photos, videos, and other documents.

- Carve for known file types.

Presenting: Apple Airprint

- Introduced in iOS 4.2 (November 2010).

- Wireless, driverless printing (to AirPrint printers).

- Easy and simple for the user.

(a) (b) (c)

Analysis of forensics traces (1/3)

AirPrint use leaves some traces on device

- While printing, we monitored files, network connections, and processes.

- Spooler directory: `/var/mobile/Library/com.apple.printd/`
  - Created when printing for the first time.

- Temporary files (`1.pdf`, `2.pdf`, ...) appear here while printing.
  - The temp file is deleted as soon as the job has finished printing.

* Printing graphics (JPG at least) does NOT generate temp files.

Analysis of forensics traces (2/3)

Properties of AirPrint temp files

- The temp PDF files contain the document sent to the printer.

- When the queue is empty again (document printed, temp file deleted):
  - iOS 4: the counter is reset (a new job would generate a new `1.pdf`).
  - iOS 5: the counter keeps increasing until the device reboots.

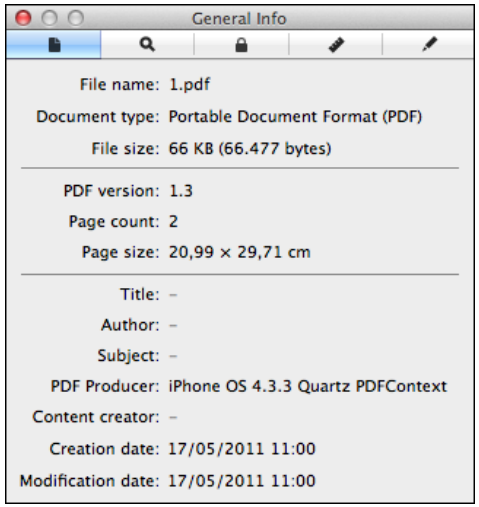
- If many copies are printed, the temp file contains an only copy of the document.

- If a page range is specified, the temp file contains only this page range.

* Exception when printing some native PDF files.

Analysis of forensics traces (3/3)

PDF metadata of temp files



- **'PDF Producer'**:
  iPhone OS x.y.z Quartz PDFContext
- **'Creation date'** == **'Modification date'**:
  Both indicate when the printing job was created.

Recoverability of Airprint traces (1/2)

Test series

iPhone 3G, iOS 4.2.1, no encryption.

1) Safari: print 4 web pages and 6 PDFs.

2) GoodReader: print 10 PDFs.

*3) Photos: print 10 photos.

4) Mail: print 5 messages and 5 attachments.

*5) Turn off. Wait for 30 sec. Turn on.

6) Safari: print 6 DOC and 4 XLS files.

*7) Mail: download 15 MB, repeat step 5.

*8) App Store: download 50 MB.

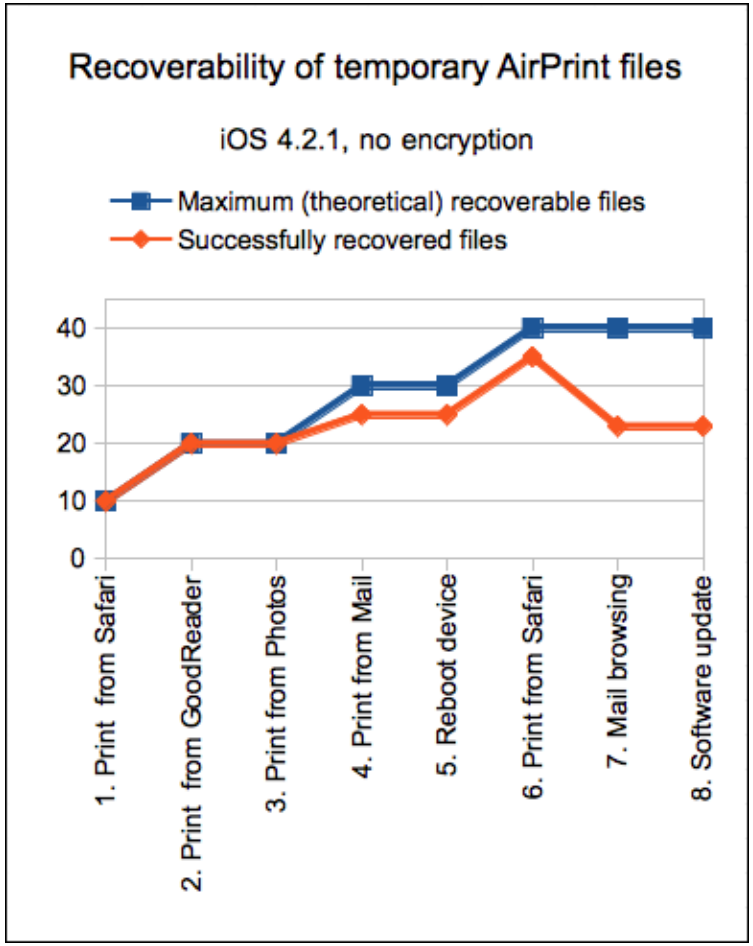
(*) == Tests that do NOT create AirPrint temp files.

Recoverability of Airprint traces (2/2)

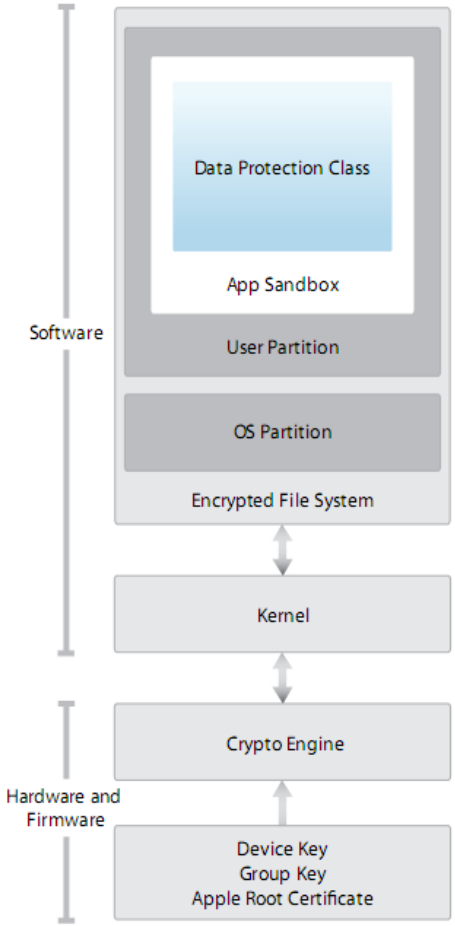
Carve PDFs with `photorec`.

Recoverable: Most artifacts were successfully recovered (even after rebooting). This confirms that AirPrint temp files are flushed to disk.

Persistant: These artifacts do not seem likely to overwrite each other's disk area.



Current challenge: IOS encryption



Now, iOS devices offer hardware-based encryption.

- Still disabled by default in some devices, but...
- Quick Summary: Carving technique does not work :-(

Good news: Feb 2012, iphone-dataprotection

- Bruteforce attack on device keys.
- We still have to look into it.

iPad Forensic data acquisition: Grasping into the black box

Joan Arnedo-Moreno, Luis Gómez-Miralles

KISON Cryptography and Information Security for Open Networks

Universitat Oberta de Catalunya (UOC)

Internet Interdisciplinary Institute (IN3)

July 10th, 2012