Evaluating Digital Forensic Options for the Apple iPad

Andrew Hay, Dennis Krill, Benjamin Kuhar, Gilbert Peterson

Chapter 20

EVALUATING DIGITAL FORENSIC OPTIONS FOR THE APPLE iPAD

Andrew Hay, Dennis Krill, Benjamin Kuhar and Gilbert Peterson

Abstract The iPod Touch, iPhone and iPad from Apple are among the most popular mobile computing platforms in use today. These devices are of forensic interest because of their high adoption rate and potential for containing digital evidence. The uniformity in their design and underlying operating system (iOS) also allows forensic tools and methods to be shared across product types. This paper analyzes the tools and methods available for conducting forensic examinations of the Apple iPad. These include commercial software products, updated methodologies based on existing jailbreaking processes and the analysis of the device backup contents provided by iTunes. While many of the available commercial tools offer promise, the results of our analysis indicate that most comprehensive examination of the iPad requires jailbreaking to perform forensic duplication and manual analysis of its media content.

Keywords: Apple iPad, forensic examinations, iOS logical file system analysis

1. Introduction

Launched in April 2010, the iPad [2] joined the iPhone and iPod Touch to become the latest mobile device to adopt Apple's iOS operating system [13]. With three million devices sold in the first 80 days since its launch [2] and 250,000 third party applications available on the platform [13], the iPad is a major addition to the crowded mobile computing market. The iPad supports multiple networking protocols and GPS, and provides up to 64 GB of storage [4]. As such, it represents a fusion of technology, which is of interest to digital evidence examiners for many of the same reasons as traditional computing hardware and mobile phones.

In contrast with the relatively open security models embraced by OS X and the iPods that preceded it, iPhone OS (the predecessor to iOS) is a closed operating environment without a traditional file system and device disk mode. The version of iOS launched with the iPad additionally supports security features such as application sandboxing, mandatory code signing and 256-bit AES hardware-based data encryption [3]. These features prevent many of the traditional digital forensic media duplication and analysis processes from being employed effectively on the iPad.

The iPhone OS has also largely invalidated the existing strategies available for iPod forensics [16, 22], a problem addressed by Zdziarski in the case of iPhone forensics [25]. Zdziarski's process requires an iPhone to be hacked in a process known as "jailbreaking." Before the development of this process, several commercial forensic tools were developed. Hoog and Gaffaney [11] present a survey of the principal tools, and Mislan [19] discusses iOS analysis in the context of general mobile device forensics. However, existing tools and methods do not yet support iOS versions 3.x and above, and the manual extraction and analysis of the iTunes backup file from a computer system paired with an iPad. Both these research gaps are addressed in this paper.

This paper makes four contributions to the field of mobile device forensics. The first is a survey of commercial software tools marketed for the forensic analysis of iPads, which have yet to be formally reviewed by NIST [20]. The second is a variation of Zdziarski's method [25] for manually imaging iPad media using jailbreaking techniques. The third is the enumeration of the forensically relevant content available on an iPad and the specification of the locations of target files in the file system. The fourth contribution is an analysis of the reviewed tools and methods along with a technique for recovering evidence from the device backup file generated by iTunes. Note that our analysis does not include any of the optional security measures that may be enabled on an iPad, such as remote wiping, passcode locking and iTunes backup encryption. Zdziarski [25] has addressed the issue of passcode locking for iPhone OS 2.x, but iTunes backup encryption remains a major obstacle for many of the tools and techniques discussed in this paper.

2. Commercial Software Tools

Three untested commercial tools with iPad compatibility are currently being marketed: Lantern [14], Mobilyze [5] and Oxygen Forensics Suite 2010 [21]. Our analysis focuses on the evidence extracted using these tools and the suitability of these tools when there is an expectation of forensic soundness. Note that the analysis is limited to the free trial versions of the marketed products whose capabilities may differ from the fully licensed versions.

Lantern (version 1.0.6.0 demo; now 1.0.9 with iOS 4.2.1 support) provides an easy-to-use interface for reviewing a limited subset of iPad data. The extraction of information is quick – it took less than seven minutes for an iPad configured with minimal media content. Multiple processing errors were listed in the error log after extraction, but no explicit warnings were raised to notify the user that something had gone wrong.

Lantern [14] extracts evidence into two categories: media and everything else. The product does not support the manual browsing of extracted data, most of which is hidden in a single file with a proprietary format. All media is stored and hashed individually; everything else is maintained in the archive file, whose hash value is displayed on the main Lantern screen. Files are hashed using MD5, but there is no facility for verifying that the exported evidence matches the archive because of a format conversion during exportation. A usability bug was identified with respect to Lantern's export data feature: the output file is written without an extension, although the file is in the CSV format and can be manually opened as such.

Mobilyze (version 1.1), which is now part of the BlackLight Forensic Suite, provides graphical information in an intuitive and organized fashion, but it only permits the viewing of a limited selection of iPad data. Device acquisition took 27 minutes – the process copies the majority of files in the iPad's user partition and includes all the media resources. The copied files are individually hashed with MD5 and the values stored in a separate log. The files are archived in a non-propriety package that can be browsed manually using the OS X file browser or the built-in Mobilyze browser. Mobilyze also offers built-in viewers for SQLite and Property List files, for which no graphical module is available.

One of the major advantages of Mobilyze is the flexible and intuitive evidence tagging and reporting facility. All the items of interest in the graphical modules can be tagged, and individual data files may be examined using the application file browser. Tagged items are formatted in rich HTML when exported as a report, with the data files clearly displaying the associated file path and hash value.

Oxygen Forensics Suite 2010 (version 2.8.1) reportedly supports evidence acquisition and analysis for more than 1,650 mobile devices. The Oxygen Connection Wizard requires the installation of OxyAgent on many target devices before acquisition can take place. Device acquisition took seventeen minutes; the process individually hashes all the files. The Oxygen Connection Wizard provides an extraction option for a full reading of the iPad file structure, but this option yields files from the user partition and does not capture any system resources. The company has released several updates since our analysis. However, while the release notes include several references to Apple, they do not specifically mention the iPad or the latest Apple iOS.

3. Manual Search Methodology

The iPad file system is "jailed" by firmware restrictions that prevent users from accessing it directly. The device itself has no disk mode to facilitate the viewing and copying of media content. Apple's philosophy is that all interactions with the device should occur through the iTunes portal. Similarly, third party applications are restricted to executing in a sandbox to prevent subversion of the iOS environment.

Three methods exist for manually recovering digital evidence from an iPad, and several tools are available for performing an analysis. Apple may assist law enforcement examinations with disk-mode unlocking of the device, essentially enabling the iPad to function as a regular external USB drive. The other two methods are jailbreaking the iPad and analyzing the iTunes backup file.

3.1 Jailbreaking and Imaging

One means of gaining root access to the iPad file system is to jailbreak its firmware. With such access, the examiner can install third party packages to image and transfer the device data to a computer via SSH. Zdziarski [25] has described the jailbreaking process for the iPhone OS v2.x. Updates are available for law enforcement [26], but further information is only available via an access-controlled site [12]. Unfortunately, the jailbreaking and package installation strategies that are publicly described are not effective for the iOS software on an iPad, although the imaging and transfer steps remain largely unchanged.

For iOS version 3.2, the user space jailbreaks, Spirit [23] and JailbreakMe [6], provide access and automatically install the Cydia package manager [10] to support the installation of additional software. Cydia is decidedly forensically-unfriendly – it installs several files in the user partition, and while its code is open source, the same cannot be said for its jailbreaking technique. Current (publicly available) jailbreaks for the iPad do not permit an examiner to access the device by installing a forensically-friendly jailbreaking tool that write-protects the user partition. Additionally, while the Spirit jailbreak can be performed without user manipulation, both methods require interaction with the iPad user interface to install the OpenSSH and netcat packages required for imaging and transfer.

Two jailbreaks can be applied to devices running iOS 4.2.1: PwnageTool 4.1.3 [7] and redsn0w 0.96b6 [7]. PwnageTool does not currently support the iPad and requires packages to be manually installed. However, redsn0w 0.9.6b6 is effective on an iPad running iOS 4.2.1 and can also install the Cydia package manager.

Zdziarski [25] justifies the forensic soundness of using a jailbreak based on hashing the entire user partition prior to imaging its contents. The problem with implementing this step is that the Cydia package manager does not contain an MD5 implementation. Using the MD5 version supplied with the OpenSSH package resulted in the spontaneous rebooting of an iPad during our tests.

Two additional forensic challenges exist with regard to jailbreaking. Relying on an available jailbreak means that there is no guarantee that a suitable tool will be available for a particular device and iOS version when an investigation is to be performed. Also, after an iPad is jailbroken, it requires software to be installed in order to recover the data. This software, which is installed from the Cydia server, is outside the direct control of the examiner and, therefore, does not qualify as a trusted executable.

Zdziarski's [25] jailbreaking method copies only a subset of an iPad's file system (which Zdziarski calls the "user partition"). This restriction derives from the fact that an iPad must be booted to create an image of its media, and only the portions of the file system that are mounted as read-only can be imaged. Ideally, a forensic media duplication process should be comprehensive. The system partition accounts for a limited portion of the file system and this partition generally does not contain files of investigative value. Most free space, all application support files, and third party executables are stored in the user partition.

The entire file system of an iPad can be examined on a jailbroken device using an iOS device browser leveraging the AFC2Add package provided by Phone Disk [17]. To obtain root access, the AFC2Add package must be installed using Cydia, which unlocks the device and provides full access over USB when the device is running. This combination permits the iPad's root contents to mount in the OS X file browser as a MacFUSE file system. Because MacFUSE mounted volumes communicate via the Apple Filing Protocol, they do not receive a disk identifier and low-level copying of their contents is not possible. However, the file system can be navigated and individual files hashed and copied from the command line. Examiners must take precautions when using this method because the mounting is done as read-write, which means that the files can be altered inadvertently.

3.2 Performing Custom Examinations

None of the commercial tools assessed proved to be as versatile and comprehensive as a manual examination of the iPad's file system. What constitutes relevant evidence depends on the circumstances of a case, but not one of the commercial tools can provide results for all scenarios. This means that an examiner will eventually have to conduct a manual process. Certain software resources and methods allow an examiner to comprehensively search for and analyze the contents of an iPad after a forensic duplicate of its media has been created.

Two reoccurring file types employed as support files across several iPad applications are Property List (Plist) files and SQLite database files. The Mac OS X command line offers built-in capabilities to read both file types using the `defaults` command for Plist files and the `sqlite3` client for databases. Alternatively, PlistEdit Pro [9] and Base [18] may be used; these applications use syntax highlighting to present the text content of files in a readable format. The exact format of the Plist files varies by application; in many cases, they can be read by an equivalent application on the Mac if the iPad version of the file is copied to the equivalent OS X user directory location. These files also store application configuration settings and state information.

In contrast, database files are not as interchangeable, although iTunes does have the ability to sync contacts, calendars, bookmarks, notes and media database content with a computer for viewing. Note that a one-way sync is not possible from an iPad, so an examiner should only attempting syncing after a forensic duplicate has been created. An alternative to syncing is the PhoneView application for the Mac [8]. This tool provides graphical browsing and searching of contacts, notes, open websites, browser history, bookmarks and media.

3.3 Analyzing the iTunes Backup File

Syncing an iPad with a computer creates an iTunes backup file, which holds the majority of the data stored on the iPad. Analysis of the iTunes backup file can be performed independently of the iPad. A disadvantage of this method is that the iPad should have been previously configured to not encrypt backups. The location of the backup file varies by host operating system [1]. The backup file contains several binary Plist files stored in a directory named with a unique identifier; a summary of the backup file contents is provided at Apple's support site [1]. iPhone Backup Extractor [15] can be used to convert the binary Plist format to permit the contents of the backup file to be viewed using the OS X

Table 1. Documents.

	Lantern	Mobilyze	Oxygen	Manual	Backup
.html			X	X	X
.doc/.docx				X	
.ppt				X	
.rtf				X	
.txt				X	
.xls				X	
.pdf			X	X	X

file browser. A method for analyzing an iPad media image can then be employed.

4. Results and Analysis

All testing was conducted on a 16 GB iPad without 3G capability running iOS version 3.2. The sources of evidence analyzed included documents, media (audio, video and images), support files for default applications (Mail, Notes, Contacts, Safari, YouTube, Maps, Calendar), third party application directories and various miscellaneous files. Since the device was jailbroken, the path of each type of content (potential evidence) refers only to the user partition mounted at `/private/var`. The tools and methods considered were: Lantern, Mobilyze, Oxygen, media image analysis and backup file analysis. Zdziarski's method [25] was modified with JailbreakMe [6] and Cydia [10] to obtain a low-level image of the user partition, which was analyzed using the tools listed in Section 3.2.

The following sections present the results obtained for the evidentiary items of interest. Note that a table entry marked "X" denotes that a particular tool returned a meaningful output for the item. An entry marked "*" denotes that problems were encountered in obtaining or analyzing the output.

4.1 Documents

This test focused on the ability of the tools and methods to locate HTML, ASCII text, Adobe PDF, Microsoft Word, Microsoft PowerPoint and Microsoft Excel documents. These file types can be found in several locations including email, web cache and third party application directories. As shown in Table 1, only the manual method accessed all the known documents of interest. A search on file extension using Mobilyze's global search field did not list any documents, while Oxygen only

Table 2. Media.

	Lantern	Mobilyze	Oxygen	Manual	Backup
Audio	*	X	*	X	
Video	*	X	*	X	
Images	*	X	X	X	X

detected a few file types. The backup contents were searched using OS X's Spotlight indexed search technology, but it only located documents associated with third party applications.

4.2 Media

Media on an iPad includes content synced from a computer or downloaded directly using the device. Music and video are stored in `/iTunes_Control` while images are located in `/mobile/Media`. The subdirectory `/mobile/Media/DCIM/100APPLE` contains all the images saved via the iPad browser and email clients. Synced images are isolated in `/mobile/Media/Photos`.

While an entertainment media library may have questionable value as evidence, there is the potential for user-generated content to be stored. For performance reasons, the iPad is prolific in caching image resources and stores many display views; these can be used to establish usage patterns and behavior. Table 2 compares the abilities of the tools and methods to locate media resources.

Manual image analysis, using the Coverflow, Quicklook and SmartFolders features of the OS X file browser, was found to be suitable for opening all the media files. In the case of Lantern, the Photo Directory button on the summary screen failed to perform any action, and the Media and Photos features were unable to open or export any displayed results. Oxygen failed to identify most video content on the device, and did not support previews of audio or video within the application or via Windows Media Player. Also, the Photo Thumbnails feature was conspicuously empty despite being shown in the Images tab of the application's file browser. Oxygen and Mobilyze misclassified audiobook files as video. Lantern and Mobilyze failed to recognize `.gif` and `.tif` image files in their respective photo browsers. While the iTunes backup did produce media results, they did not include content from iPad Photos or iPod applications. All the commercial tools had difficulties with the playback of DRM protected media content; only Mobilyze offered to open iTunes for its authorization and playback.

Table 3. Mail, notes and contacts.

	Lantern	**Mobilyze**	**Oxygen**	**Manual**	**Backup**
IMAP		*		X	
POP		*		X	
Attachments				X	
Notes	X	X	X	X	X
Contacts	X	X	X	X	X
Contact Images		X	X	X	X

4.3 Mail

An iTunes backup file includes POP mail messages that are viewable using Emailchemy [24], but not messages sent using IMAP. The backup file also includes account settings that are stored at `/mobile/Library/Preferences/com.apple.accountsettings.plist`.

Partially-cached message content from an IMAP account can be extracted using a manual two-step process: search and then find. The search step uses a string search to identify files of interest. The focus should be on files with the `.emlxpart` extension that correspond to incomplete cached contents of IMAP messages. Alternatively, any SQLite client can be used to search the table-based organization of message headers [25]. Individual files can then be viewed in an application capable of reading rich text, HTML and images such as TextEdit. The context independent file browsing capabilities of FTK make it the only known option for viewing cached IMAP attachments.

Our test device contained IMAP and POP mail messages with a variety of attachments. Table 3 summarizes the results of our analysis. Mobilyze provided a location on the device information screen for Mail, but it reported a null value and the link to view the contents was absent. Also, the Mail folder contents were missing in the file browser.

4.4 Notes and Contacts

The notes application maintains a single SQLite database at `/mobile/Library/Notes/notes.db`. Table 3 shows that all the tools and methods tested opened the contents with ease.

The iPad Contacts application stores data in two SQL databases: `/mobile/Library/AddressBook/AddressBook.sqlitedb` with contact information and `/mobile/Library/AddressBook/AddressBookImages.sqlitedb` with the associated contact image files. As shown in Table 3, only Lantern failed to show the images associated with contacts. It also represented business contacts as blank entries and suffered from readability problems.

4.5 Safari

The Safari browser stores relevant information in several different files:

- `/mobile/Library/Safari/Bookmarks.db`
  – Bookmarked websites.

- `/mobile/Library/Safari/History.plist`
  – Previously visited websites.

- `/mobile/Library/Cookies/Cookies.plist`
  – Cookies installed by visited websites, possibly including website account information.

- `/mobile/Library/Caches/Safari/Thumbnails`
  – Cached images.

- `/mobile/Library/Caches/Safari/RecentSearches.plist`
  – Last twenty search strings entered in the Safari search bar.

- `/mobile/Library/Safari/SuspendState.plist`
  – Last Safari configuration (open windows and associated URLs) before it was quit.

- `/mobile/Media/WebClips`
  – Bookmarks saved to the iPad's home screen; these are shortcuts to websites that launch in Safari and use the favicons of the pages as their home screen icons.

PhoneView was used to successfully display and search the bookmarks and history files. Since the history file is a Plist file, it could also be copied to the `~/Library/Safari` directory on a Mac and viewed using the OS X version of Safari. This method also enabled some cached web content from the history file to be browsed via Coverflow. The cookies file could be copied to `~/Library/Cookies` on the Mac and its contents viewed by selecting *Security* → *Show Cookies* from the Safari application preferences. As a performance measure, many apps (including Safari on the iPad) cache a screenshot of the last view of an application before quitting to increase the perceived speed on relaunch. This image can be identified based on its creation date in the Thumbnails directory, enabling the verification of the last item viewed in Safari.

Table 4 presents the test results. Lantern, Mobilyze and Oxygen all contain an interface element for displaying bookmarks, but they failed to

Table 4. Safari.

	Lantern	Mobilyze	Oxygen	Manual	Backup
History	X	X	X	X	X
Cookies				X	X
Recent Searches				X	
SuspendState				X	X
Bookmarks	*	*	*	X	X
Cache				X	
WebClips				X	X

show the results that were known to be present. The marketed version of Oxygen provides an optional cache analyzer, but this is not included in the trial version of the tool used in our tests. The backup file excluded most cached application contents, including those from Safari.

4.6 YouTube

YouTube is a default application in iOS that has several support files. A value string in the history dictionary is accessed on the web by prepending the value with `http://www.youtube.com/watch ?v=` in a web browser. The values are found in `/mobile/Library/Preferences/com.apple.youtube.plist`, which maintains a list of video bookmarks, the last search string used in the application search bar and the video viewing history. The user's account name is stored in `/mobile/Library/Preferences/com.apple.youtubeframework.plist`.

Table 5. YouTube.

	Lantern	Mobilyze	Oxygen	Manual	Backup
Account Details				X	
Search History				X	X
Viewing History				X	X
Bookmarks		*		*	*

The results of the tests are shown in Table 5. Mobilyze provides an interface element for YouTube Bookmarks on the device information screen, but it displayed a null value despite the fact that several bookmarks were created during our tests. Curiously, the Bookmarks key in the first Plist file also had a null value, indicating that YouTube Bookmarks may not be stored locally, but are pulled from the web. The iTunes backup file version of `com.apple.youtubeframework.plist` excludes the key and value for *YouTubeAccount*, although the file itself is present.

Table 6. Maps.

	Lantern	Mobilyze	Oxygen	Manual	Backup
Last Lat./Long. Viewed				X	X
Search History	X	X		X	X
Map Tile Cache				X	
Bookmarks	*		*	X	X

4.7 Maps

Since the iPad incorporates GPS hardware, data from its Maps application can be used to determine where the device has been. Search and location history data, in particular, can be invaluable in investigations.

The Maps application leverages the Google Maps API and has several support files:

- `/mobile/Library/Preferences/com.apple.Maps.plist`
  – Last latitude and longitude viewed.

- `/mobile/Library/Maps/History.plist`
  – History of address lookups, including latitude and longitude, query name and city.

- `/mobile/Library/Maps/Bookmarks.plist`
  – List of custom pins, including the name and location of each pin.

- `/mobile/Library/Caches/MapTiles/MapTiles.sqlitedb`
  – Cache of the most recently viewed map tiles.

The test results are shown in Table 6. Lantern includes an interface feature for Maps Bookmarks, but it was discovered to be empty in our tests. Oxygen includes a Geo tab with the globe icon in its file browser, which we expected to display files relevant to geographical locations; however, this tab was found to be empty in our tests.

4.8 Calendar

The best way of viewing the Calendar application contents is to allow iTunes to sync the information from the iPad, and view the contents on a Mac. The database `/mobile/Library/Calendar/Calendar.sqlitedb` can be analyzed using the same methods as for SQLite databases. The Event table is very useful because it lists every recent and upcoming event along with its summary, location and time.

Table 7 shows that all the tools and methods were able to extract information from the Calendar database. Mobilyze, which provides a link

Table 7. Calendar and applications.

	Lantern	Mobilyze	Oxygen	Manual	Backup
Calendar	X	*	X	X	X
Third Party Apps	*	X	*	X	X

for Calendars on the device information screen, displayed the dates as Unix timestamps, rendering the results practically unreadable. Oxygen provided the most complete graphical display of Calendar information, including alarms and recurrence information.

4.9 Installed Applications

Supporting files for third party applications (apps) that are downloaded and installed via the App Store on an iPad (or synced from iTunes) are stored in subdirectories under `/mobile/Applications`. Application data storage varies widely in terms of content and organization. Of the commercial tools reviewed, Mobilyze provides the best interface for analyzing these files. Manual analysis is not difficult, but can be time consuming if there are many files to sort through because the apps are not organized into named folders. Due to the iPad's sandboxing, most supporting files remain local to the individual application folder. The application directories `/Library/Caches` and `/Library/Preferences` should be analyzed for evidence associated with application support files. Due to the proliferation of caching, many applications store vast amounts of data. In our tests, news and media applications were found to generate hundreds of images. Consequently, this resource should not be overlooked in an investigation.

Table 7 presents our test results. Lantern includes a 3rd Party App Directory button on the device information screen, but it failed to perform any actions in our tests. Oxygen includes an Applications tab in its file browser, but it failed to list any information. The iPhone Backup Extractor can be used to individually extract supporting files for apps (each file is stored separately from the general iOS backup file).

4.10 Miscellaneous Evidentiary Sources

Several other locations may contain information of value to an investigation (e.g., information related to the computers and networks connected to by the iPad). The sources include:

- `/root/Library/Lockdown/data_ark.plist`
  – Device and account information, including device name, time

zone, list of App Store applications downloaded by each iTunes Store account (not just currently installed), current iTunes Store account and any additional accounts, and the user's AppleID.

- `/root/Library/Lockdown/pair_records`
  – Property lists associated with computers that have been paired with the iPad (see [25] for details).

- `/root/Library/Caches/locationd/cache.plist`
  – Latitude and longitude of the most recently used wireless access point and the most recent GPS coordinates.

- `/preferences/SystemConfiguration/com.apple.wifi.plist`
  – Names and configurations of known wireless access points.

- `/mobile/Library/Keyboard/dynamic-text.dat`
  – User-defined dictionary that could be used to decipher the jargon used in text communications.

- `/mobile/Library/Keyboard/en_US-dynamic-text.dat`
  – Keyboard cache with text recently entered by the user; it lacks context but can be viewed as plain text.

- `/mobile/Library/HomeBackground.jpg`
  – Current home screen wallpaper.

- `/mobile/Library/Lock Background.jpg`
  – Current locked screen wallpaper.

- `/mobile/Library/Caches/com.apple.UIKit.pboard`
  – Current content of the iPad's pasteboard; can be viewed as plain text.

- `/mobile/Media/iTunes_Control/Device/SysInfoExtended`
  – Plist file containing the iPad's UDID and Serial Number.

The test results are shown in Table 8. Lantern mistakenly identifies the user dictionary as the keyboard cache on its website, describing this feature as "a keylogger for the iPhone;" this bug appears to simply reference the wrong text file. Oxygen includes a Wi-Fi connections feature that lists information about known wireless networks, including the SSID, joined date and time, last used date and time, and location of each access point. This feature appears to leverage the Google Gears Geolocation API, and was unique among the products tested. However, the location data was not 100% current and the location of one access point was reported incorrectly.

Table 8. Miscellaneous evidentiary sources.

	Lantern	Mobilyze	Oxygen	Manual	Backup
iTunes Download History				X	
AppleID				X	
Known Wi-Fi Access Points			X	X	X
Location Services				X	
Desktop Pairings				X	
Keyboard Cache				X	X
User Dictionary	X			X	X
Wallpaper				X	X
Pasteboard Contents				X	
Bluetooth Address	X		X		
Wi-Fi Address	X		X		
Device Name	X	X	X	X	
Serial Number	X	X	X	X	
Unique Device ID	X	X	X	X	
Product Version		X	X		
Build Version		X	X		

5. Conclusions

Mobile devices are of growing interest to digital forensic examiners because of their increasing pervasiveness and evidentiary potential. These devices present unique challenges to forensic examiners because of the high degree of variance in hardware, propriety operating environments and custom third party software. Keeping the digital forensics community abreast of the tools and techniques applicable to the dizzying array of devices available today is an ongoing, iterative process.

This paper has focused on the iPad, a member of the family of mobile hardware that uses Apple's iOS mobile operating system. The paper has surveyed existing tools and methods for forensic duplication and media examination. Comparison of the results obtained using the commercial tools with those obtained using a manual process reveal that manual media imaging and analysis provide the most comprehensive results. However, legal and technical challenges are inherent in obtaining a bit-for-bit copy of the iPad's media.

Our future work will examine how the optional security features available in iOS (remote wiping, passcode locking and iTunes backup encryption) impact the efficacy of the tools and methods discussed, and how the security features can be bypassed when conducting forensic examinations. Another key research problem is to obtain unfettered access to the iPad's media so that it can be fully imaged without relying on firmware hacks or assistance from Apple.

Note that the views expressed in this paper are those of the authors and do not reflect the official policy or positions of the U.S. Air Force, U.S. Department of Defense or the U.S. Government.

Acknowledgements

This research was supported by the U.S. Air Force Cyberspace Technical Center of Excellence and the U.S. Air Force Office of Scientific Research (AFOSR/RSL).

References

[1] Apple, iPad: About backups, Cupertino, California (support.apple.com/kb/HT4079), 2009.

[2] Apple, Apple sells three million iPads in 80 days, Cupertino, California (www.apple.com/pr/library/2010/06/22ipad.html), June 22, 2010.

[3] Apple, iPad in Business: Security Overview, White Paper, Cupertino, California, 2010.

[4] Apple, iPad Technical Specifications, Cupertino, California (www.apple.com/ipad/specs), 2011.

[5] BlackBag Technologies, Mobilyze, San Jose, California (www.blackbagtech.com/forensics/mobilyze/mobilyze.html).

[6] Comex, JailbreakMe 2.0 (www.jailbreakme.com).

[7] Dev-Team Blog, Homepage (blog.iphone-dev.org).

[8] Ecamm Network, PhoneView, Somerville, Massachusetts (www.ecamm.com/mac/phoneview).

[9] Fat Cat Software, PlistEdit Pro, San Jose, California (www.fatcatsoftware.com/plisteditpro).

[10] J. Freeman, Cydia (cydia.saurik.com).

[11] A. Hoog and K. Gaffaney, iPhone Forensics White Paper, 2009.

[12] iPhone Insecurity, Homepage (www.iphoneinsecurity.com).

[13] S. Jobs, Keynote address, *Apple Worldwide Developers Conference* (www.apple.com/apple-events/wwdc-2010), 2010.

[14] Katana Forensics, Easton, Maryland (www.katanaforensics.com).

[15] P. Kennedy, iPhone/iPod Touch Backup Extractor (www.supercrazyawesome.com).

[16] M. Kiley, T. Shinbara and M. Rogers, iPod forensics update, *International Journal of Digital Evidence*, vol. 6(1), 2007.

[17] Macroplant, Phone Disk, Arlington, Virginia (www.macroplant.com/phonedisk).

[18] Menial, Base (menial.co.uk/software/base).

[19] R. Mislan, Cellphone crime solvers, *IEEE Spectrum*, vol. 47(7), pp. 34–39, 2010.

[20] National Institute of Standards and Technology, Mobile devices, Computer Forensics Tool Testing Program, Gaithersburg, Maryland (www.cftt.nist.gov/mobile_devices.htm).

[21] Oxygen Software, Oxygen Forensics Suite 2010, Moscow, Russia (www.oxygen-forensic.com/en).

[22] J. Slay and A. Przibilla, iPod forensics: Forensically sound examination of an Apple iPod, *Proceedings of the Fortieth Annual Hawaii International Conference on System Sciences*, 2007.

[23] Spirit, Homepage (www.spiritjb.com).

[24] Weird Kid Software, Emailchemy, Detroit, Michigan (www.weirdkid.com/products/emailchemy).

[25] J. Zdziarski, *iPhone Forensics*, O'Reilly, Sebastopol, California, 2008.

[26] J. Zdziarski, Jonathan Zdziarski's Domain (www.zdziarski.com/blog/?p=524).