iPhone Forensics

Andrew Hoog, updated by Kyle Gaffaney

iPhone Forensics

Andrew Hoog, updated by Kyle Gaffaney

Published by Andrew Hoog March 2009. Updated June 2009 by Andrew Hoog and Kyle Gaffaney

Chapter 1. iPhone Forensics Overview and Techniques

1. Introduction

The iPhone was introduced on January, 2007 and has now surpassed the Blackberry as the second largest supplier of smart phones. Regardless if this is sustainable, the Apple iPhone already has a significant footprint and will appear more frequently in computer forensic cases. The iPhone has an active hacking community which has yielded research and tools which support forensic investigations. Several commercial software packages now offer iPhone support and in September 2008, O'Reilly released “iPhone Forensics, 1st Edition” by Jonathan Zdziarski.

This paper will review forensic tools available for the iPhone, perform forensic analysis with each tool and report on the installation, acquisition, reporting and accuracy of each tool. The 3G iPhone (firmware version 2.2) was used for the testing but this white paper may, over time, include other models and firmware versions.

1.1. iPhone Technical Overview

1.1.1. iPhone 3G Hardware Components

The iPhone, like most complex electronic devices, is a collection of modules, chips and other electronic components from many manufacturers. Due to the complex and varied features of the iPhone, the list of hardware is extensive. The following information is based on the research published online. [XXX]

Table 1.1.

Function	Manufacturer	Model/Part Number
Application Processor (CPU)	Samsung	S5L8900B01 – 412 MHz ARM1176Z(F)-S RISC, 128 Mbytes of stacked, package-onpackage, DDR SDRAM
3D graphic acceleration	Imagination Technologies	Power VR MBX Lite
UMTS power amplifier (PA), duplexer and transmit filter module with output power detector	TriQuint	TQM676031 – Band 1 – HSUPA TQM666032 – Band 2 – HSUPA TQM616035 – Band 5/6 - WCDMA/HSUPA PA-duplexer
UMTS transceiver	Infineon	PMB 6272 GSM/EDGE and WCDMA PMB 5701
Baseband processor	Infineon	X-Gold 608 (PMB 8878)
Baseband's support memory	Numonyx	PF38F3050M0Y0CE - 16 Mbytes of NOR flash and 8 Mbytes of pseudo-SRAM
GSM/EDGE quad-band amp	Skyworks	SKY77340 (824- to 915-MHz)
GPS, Wi-Fi, and BT antenna	NXP	OM3805, a variant of PCF50635/33
Communications power management	Infineon	SMARTi Power 3i (SMP3i)
System-level power management	NXP	PCF50633
Battery charger/USB controller	Linear Technology	LTC4088-2
GPS	Infineon	PMB2525 Hammerhead II
NAND flash	Toshiba	TH58G6D1DTG80 (8 GB NAND Flash)

Function	Manufacturer	Model/Part Number
Serial flash chip	SST	SST25VF080B (1 MB)
Accelerometer	ST Microelectronics	LIS331 DL
Wi-Fi	Marvell	88W8686
Bluetooth	CSR	BlueCore6-ROM
Audio codec	Wolfson	WM6180C
Touch screen controller	Broadcom	BCM5974
Link display interface	National Semiconductor	LM2512AA Mobile Pixel Link
Touch screen Line Driver	Texas Instruments	CD3239

1.1.1.1. CPU

The Samsung CPU is a RISC (Reduced Instruction Set Computer) processor that runs the core iPhone processes and works in conjunction with the PowerVR co-processor for graphics acceleration. The CPU is under clocked to 412 MHz (from a possible 667 MHz) presumably to extend battery life.

1.1.1.2. Baseband

This is the component in the iPhone that manages all the functions which require an antenna, notably all cellular services. The baseband processor has its own RAM and firmware in NOR flash, separate from the core resources and functions as a resource to the main CPU. The Wi-Fi and Bluetooth are managed by the main CPU, although the baseband stores their MAC addresses in its NVRAM. [XXX]

Figure 1.1. iPhone teardown image - top

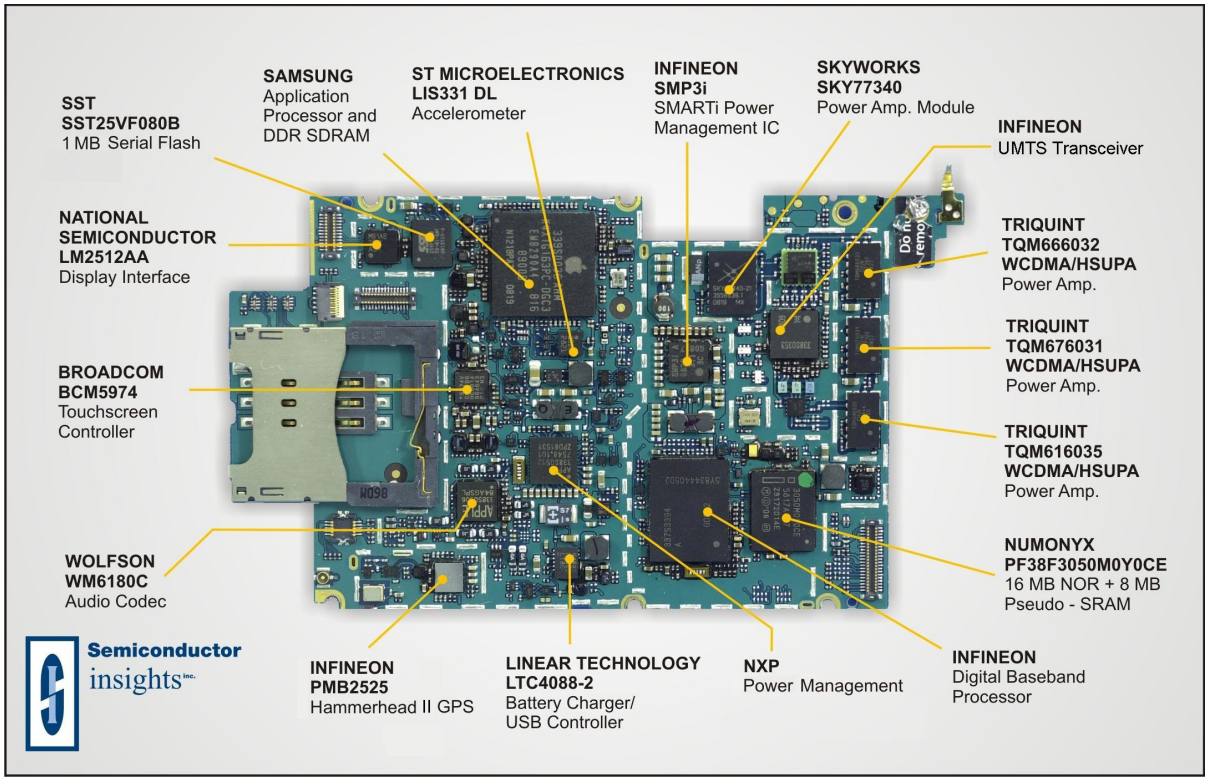
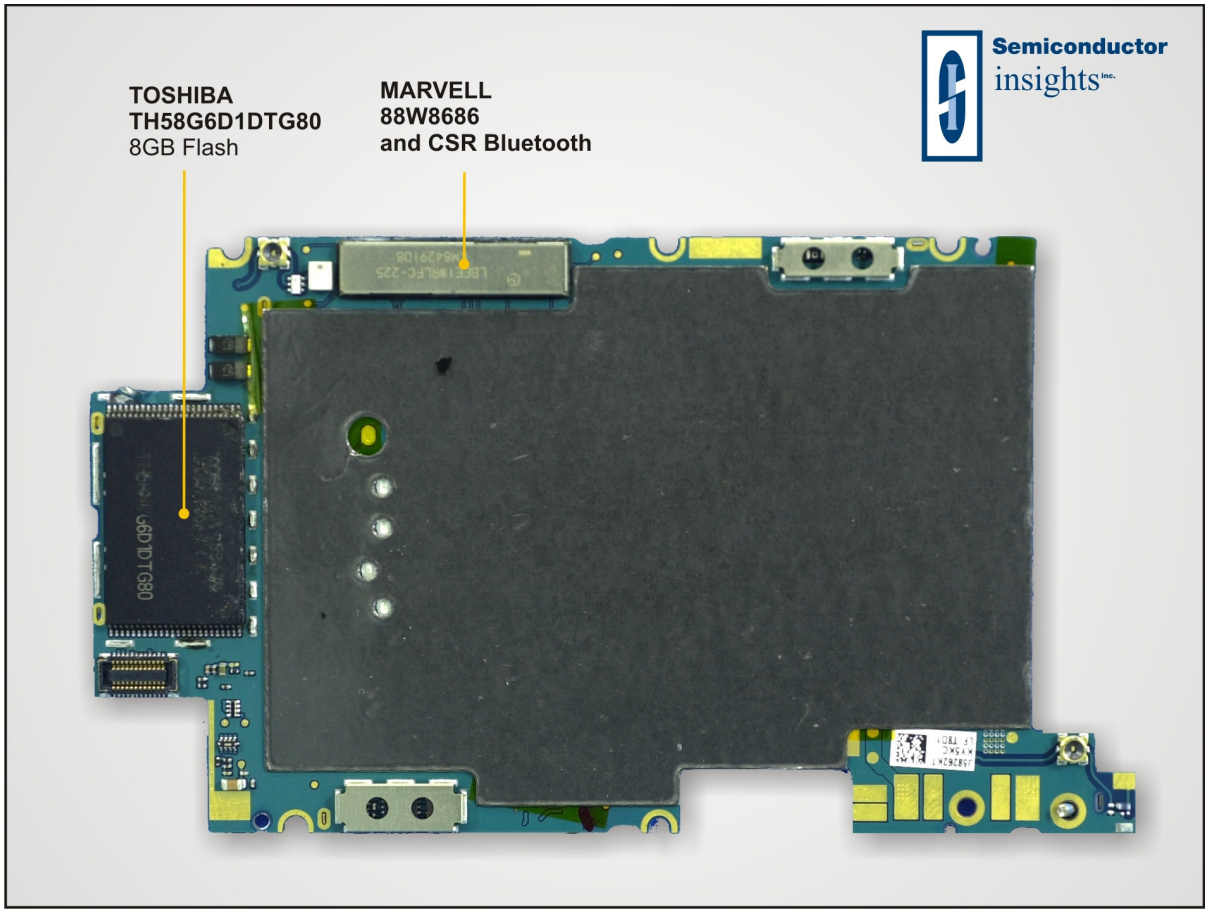


Figure 1.2. iPhone teardown image - bottom



1.1.2. Software

The Apple iPhone's operating system, iPhone OS, is a variant of Apple's core operating system, OS X. Based on the same MACH kernel and sharing some core elements with OS X 10.5 (Leopard), the iPhone is comprised of 4 layers including the core OS, the Core Services API, the Media layer and the Cocoa Tough layer. Entire books are dedicated to the operating system and the development of applications. Research into these areas will improve an analyst's skills and could be central to solving investigations. Also, the iPhone software development kit (SDK) is free to download after registration and is recommended for anyone performing forensic analysis on the iPhone.

1.1.3. iPhone Forensic Techniques

Like any forensic investigation, there are several approaches that can be used for the acquisition and analysis of information. A key aspect of any acquisition, arguably the most important, is that the procedure does not modify the source information in any manner. Or, if it is impossible to eliminate all modifications, the analyst must detail the changes and the reasons why it was necessary.

The following points highlight the various techniques utilized by the products tested.

- Acquire data directly from the iPhone: this approach is preferred over recovering files from the computer the iPhone was synced with (details at http://chicagoediscovery. com/iphone-forensic-howtos/forensic-analysis-iphone-backupdirectory. html). However, the forensic analyst must understand how the acquisition occurs, if the iPhone is modified in any way and what the procedure is unable to acquire.

- Acquire a backup or logical copy of the iPhone file system using Apple's protocol: this procedure will read files from the iPhone using Apple's synchronization protocol but is only able to acquire files explicitly synchronized by the protocol. Many key pieces of information are stored in SQLite databases and these are supported by the protocol. By querying the databases directly, you can generally recover more information such as deleted SMS and emails messages.

- Physical bit-by-bit copy: this process creates a physical bit-by-bit copy of the file system, similar to the approach taken in many personal computer forensic investigations. While this approach has the potential for the greatest amount of data recovered (including deleted files), the process is quite complicated and required modifying the system partition of the iPhone.

Another key point of consideration for an iPhone forensic tool is how it handles an iPhone that has a pass code set. Several products offer different strategies for this situation, each with their own benefits and drawbacks.

1.1.4. iPhone Forensic Scenarios and Analysis Methodology

1.1.4.1. Test iPhone

A 3G iPhone running firmware 2.2 and not jailbroken was used for this forensic analysis. The phone was heavily used including:

- Email, contacts and calendar (Microsoft Exchange Active Sync with Exchange 2007)
- Web browsing (news, online banking, Gmail accounts, Google, MLB, etc.)
- Phone calls, text messages (some deleted)
- App store (Facebook, Remote, Google Earth, Urbanspoon, Crazy Pumpkin, Now Playing, Units, SFNetNews, Stanza, WordPress and TwitterFon)
- Multiple Wi-Fi networks
- Camera and iTunes synced pictures (some deleted)
- Songs via iTunes
- YouTube movies
- Google Maps
- Notes

For obvious privacy reasons, personal information will be redacted as needed throughout the report. A comparison of what each tools is able to extract will be a primary focus of this white paper.

1.1.4.2. Analysis Methodology

Each forensic tool is rated on four general areas based on the following percentages:

Table 1.2. Forensic Tool Analysis Areas

Area	Weight	Description
Installation	10%	This cover installation, activation and updates of the forensic tool
Acquisition	10%	This covers the acquisition process
Reporting	20%	This covers the reporting process
Accuracy	60%	This covers the accuracy and completeness of the information acquired

To determine accuracy of a forensic tool, I compared the results of the acquisition to the expected results and assigned a quantitative number between 0 and 5 for each of the 27 scenarios outlined below. If a tool failed to recover any data in a particular area, it was rated a 0 for that category. A rating of 1 or 2 indicated some information was recovered however it did not meet the expected result. A rating of 3 indicated the tool met the expected results. Ratings of 4 or 5 indicated the tool exceeded the expected result including recovering deleted data and/or more information than other tools were able to recover. For readability, I also included the following text description of each rating:

- 0: miss

- 1-2: Below
- 3: Meet
- 4-5: Above

If a forensic tool provided multiple methods to acquire information from the iPhone and the analysis took place separately, I provide rankings for each method and then the overall tool is assigned a total rank.

The rankings in this white paper are based on my individual experiences and should be considered my opinion only. I am not recommending or endorsing any forensic tool or technique reviewed. I would strongly encourage investigators to personally test the forensic tools themselves (many offer a demo version) and form their own opinions of each product.

1.1.4.3. Test Scenarios

The following chart illustrates the 27 test scenarios and expected results.

Table 1.3. Test Scenarios

Scenario	Description
Call Logs	Determine whether the tool can find call log information on the phone. -iPhone contained full populated Call Log, no entries were deleted. -Expect that tool can connect, acquire and report on full call log containing 100 records. - Expect remnants of purged logs can be recovered and reported.
SMS	Determine whether the tool can find Short Message Service (SMS) information on the phone. - iPhone contained 30 SMS conversations, each with multiple messages. Total messages were 827. Deleted 2 conversations resulting in total of 262 messages. - Expect that tool can connect, acquire and report on 262 undeleted SMS messages. Expect remnants of deleted SMS messages can be recovered and reported.
Contacts	Determine whether the tool can find Contact information on the phone. - iPhone contained 1284 contacts, 14 with images associated. Deleted 2 contacts resulting in total of 1282 and 14 with images. - Expect that tool can connect, acquire and report on 1282 undeleted Contacts, 14 with pictures. Expect remnants of 2 deleted Contacts can be recovered and reported.
Email	Determine whether the tool can find email messages on the phone. - iPhone was synchronized with Exchange 2007 and contained thousands of emails. Specific folders (Inbox, Sent, Drafts) were downloaded and should contain 200, 200 and 7 messages respectively. - Expect that tool can connect, acquire and report on several hundred email messages. Expect remnants of deleted or purged email messages can be recovered and reported.
Calendar	Determine whether the tool can find Calendar information on the phone. - Calendar contained 3,070 appointments and no entries were intentionally deleted however during normal usage, some appointments were likely deleted. - Expect that tool can connect, acquire and report on 3070 Calendar items. Expect remnants of deleted or purged Calendar items can be recovered and reported.

Notes	Determine whether the tool can find Notes information on the phone. - iPhone contained 1 note and 1 note was deleted. - Expect that tool can connect, acquire and report on 1 undeleted note. Expect remnants of the deleted note can be recovered and reported.
Pictures	Determine whether the tool can find image files on the phone. - iPhone contained 41 pictures taken with the onboard camera and 9 that were deleted. iPhone also contained 1 picture that was synchronized from iTunes on a host PC and 1 that was deleted. - Expect that tool can connect, acquire and report on 42 pictures. Expect remnants of deleted pictures can be recovered and reported. Expect that pictures downloaded by various iPhone applications including Safari web browser, Facebook application and more can me recovered and reported.
Songs	Determine whether the tool can find music files on the phone. - iPhone contained 44 songs synchronized via iTunes from a host PC, 38 of which contained DRM protection. No songs were deleted. - Expect that tool can connect, acquire and report on 44 undeleted music files.
Web History	Determine whether the tool can find web browser history information on the phone. - iPhone contained 2 browser history entries and 7 were deleted. - Expect that tool can connect, acquire and report on 2 undeleted browser history entries. Expect remnants of deleted browser history can be recovered and reported.
Bookmarks	Determine whether the tool can find bookmarks from the Safari web browser on the phone. - iPhone contained 11 Safari bookmarks and 1 was deleted. Of the 11, 6 are a standard configuration for Safari. - Expect that tool can connect, acquire and report on 5 user bookmarks. Expect remnants of deleted bookmark messages can be recovered and reported.
Cookies	Determine whether the tool can find web browser cookie information on the phone. - iPhone contained numerous cookie files from web browsing via Safari and other applications. - Expect that tool can connect, acquire and report on Safari cookie files. Expect cookie files of other applications can be recovered and reported.
Applications	Determine whether the tool can find Application information on the phone. - iPhone contained 7 Applications and 3 that were deleted. - Expect that tool can connect, acquire and report on 7 undeleted applications and their associated information. Expect remnants of deleted applications can be recovered and reported.
Google Maps	Determine whether the tool can find Google Maps information on the phone. - iPhone contained the Google Maps application and it was used for location information and directions. No information was deleted from this application. - Expect that tool can connect, acquire and report on Google Maps information including history of location information and directions. Expect remnants of map tiles (images) can be recovered and reported.

Voicemail	Determine whether the tool can find Voicemail information on the phone. - iPhone contained 11 voicemail messages on the phone. - Expect that tool can connect, acquire and report on 11 voicemail messages.
Passwords	Determine whether the tool can find various application and network password information on the phone. - iPhone contained various passwords from Applications and network resources such as VPN, Bluetooth, Apple iTunes ID and more. - Expect that tool can connect, acquire and report on application and network passwords. Expect remnants of deleted passwords can be recovered and reported.
Configuration files	Determine whether the tool can find phone and application configuration files in the XML and Plist formats on the phone. - iPhone contained many XML and Plist configuration files. In the course of normal usage, some configuration information would have been deleted - Expect that tool can connect, acquire and report on many XML and Plist configuration files. Expect remnants of deleted configuration files can be recovered and reported.
Phone Information	Determine whether the tool can report on basic phone information. - iPhone is a GSM device and contains basic identification information such as IMSI, IMEI, ICCID, MSISDN (Phone Number), Serial Number, phone name, Wi-Fi MAC address and Bluetooth MAC address - Expect that the tool can connect, acquire and report on basic phone information listed above.
Video	Determine whether the tool can find video information on the phone. - iPhone contained 1 video and 1 deleted video that were synchronized with iTunes on the Host PC. - Expect that tool can connect, acquire and report on 1 video file. Expect remnants of 1 deleted video file can be recovered and reported.
Podcasts	Determine whether the tool can find Podcast information on the phone. - iPhone contained 1 Podcast and no Podcasts were deleted. - Expect that tool can connect, acquire and report on 1 Podcast.
Speed Dials	Determine whether the tool can find Speed Dial information on the phone. - iPhone contained 4 Speed Dial (Favorites) and no speed dials were deleted. - Expect that tool can connect, acquire and report on 4 Speed Dial favorites.
VPN	Determine whether the tool can find VPN configuration information on the phone. - iPhone contained 1 active VPN profile and 1 deleted VPN profile. - Expect that tool can connect, acquire and report on 1 active VPN profile. Expect remnants of deleted VPN profile can be recovered and reported.
Bluetooth	Determine whether the tool can find Bluetooth pairing information on the phone. - iPhone was paired with 1 Bluetooth headset and no pairings were deleted. - Expect that tool can connect, acquire and report on 1 Bluetooth pairing.

GPS	Determine whether the tool can find GPS information on the phone. - iPhone contains GPS device and software and many applications use this information. - Expect that tool can connect, acquire and report on GPS information including coordinate and date/time from various application usage.
File Hashes	Determine whether the tool creates MD5 or SHA1 hashes for information on the phone. - Expect that tool will create MD5 hashes for files extracted from the iPhone.
YouTube	Determine whether the tool can find YouTube video information on the phone. - iPhone was used to watch YouTube videos via the YouTube Application. - Expect that tool can connect, acquire and report on YouTube videos viewed.
HTML	Determine whether the tool can find cached HTML files on the phone. - iPhone was used to browse many web sites and cached files from this activity are located on the phone. - Expect that tool can connect, acquire and report on HTML files on the phone from Safari and other applications.
Office Documents	Determine whether the tool can find Office documents (PDF, Word, Spreadsheets and PowerPoint) documents on the phone. - iPhone contained office documents that were downloaded through email or the Safari web browser - Expect that tool can connect, acquire and report on office documents located on the phone.

1.1.4.4. Expected Results

Table 1.4. Expected Results

Scenario	Description
Call Logs	100
SMS	30 threads (deleted 2, 262 messages left of ~800)
Contacts	1282 (deleted 2)
Email	Inbox, Sent (200 each), Drafts – 1
Calendar	3070
Notes	1 (deleted 1)
Pictures	41 (deleted 9 camera, 1 sync)
Songs	44
Web History	2 (deleted 7)
Bookmarks	11 (deleted 1)
Cookies	Unknown
Applications	7 (deleted 3)
Google Maps	Yes
Voicemail	At least 1
Passwords	Wi-Fi, VPN, BT
Plists/XML	Unknown
Phone Information	Yes
Video	1 (deleted 1)

Podcasts	1
Speed Dials	4
VPN	1
Bluetooth	1
GPS	Unknown
File Hashes	N/A
YouTube	Unknown
HTML	Unknown
Office Documents	Unknown

1.2. iPhone Forensic Analysis

Whenever possible, I preformed the forensic testing on a Windows XP Professional workstation (SP3) instead of a Mac or Linux workstation to more closely mimic what many analysts use. The tests were performed in the following order:

1. WOLF - Sixth Legion
2. UFED - Cellebrite
3. Device Seizure - Paraben
4. MacLockPick - SubRosaSoft
5. MDBackupExtract – BlackBag Tech
6. Physical DD – Jonathan Zdziarski

For each software application, I will provide a brief overview of the software and forensic process. I will also provide feedback on the installation process, user interface, acquisition process and the results of the acquisition.

Chapter 2. WOLF by Sixth Legion (1.8/5.0)

1. Summary (from Company Information)

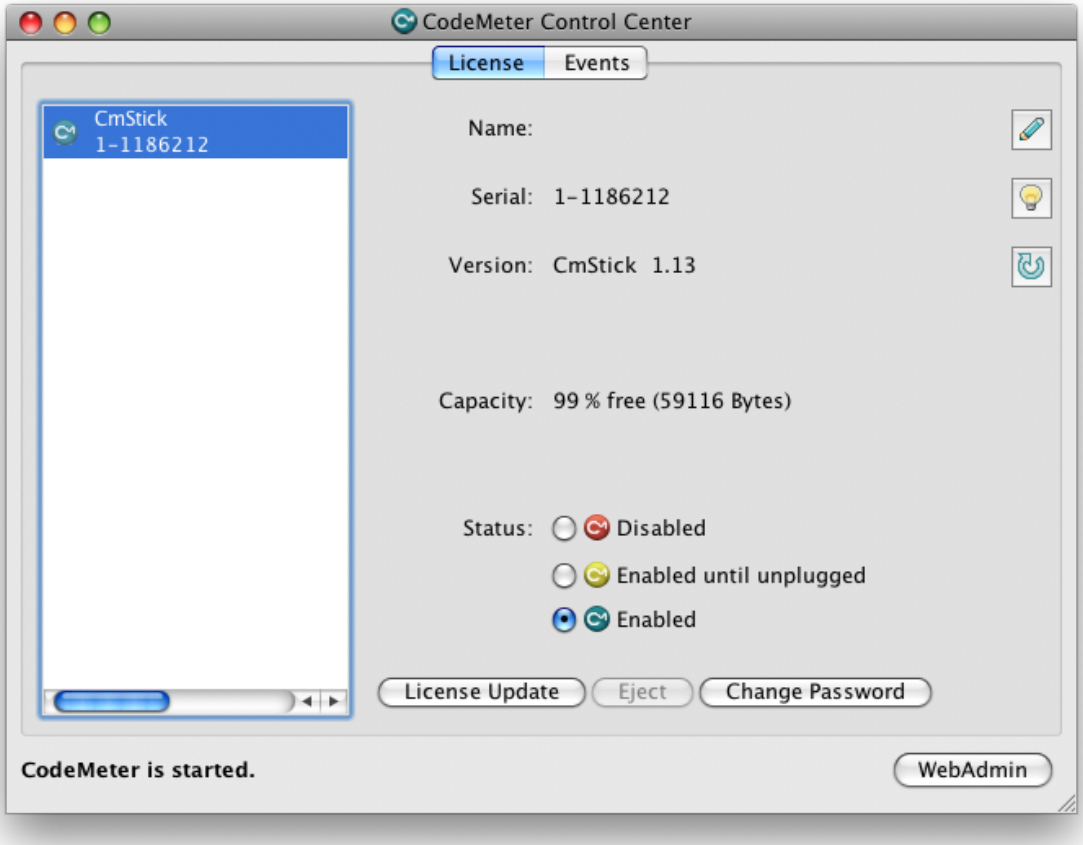
WOLF (by Sixth Legion, LLC., a division of Innovative Digital Forensic Solutions, LLC.) is a forensic tool designed specifically for the iPhone and supports all iPhone models (2G and 3G) running any firmware versions (1.0 – 2.2). The software only runs on Mac OS X (10.4.11 or greater) although a Windows version (called Beowulf) will be released soon. A dongle is required to run the software and you must install the Code Meter framework to activate the dongle. WOLF is able to bypass the security pass code (iPhone, SIM or both) without jailbreaking the iPhone, provided you have access to a physical computer that the phone has been used with. WOLF also claims to be the only iPhone forensic software that does not modify the iPhone (i.e. place an acquisition utility on the iPhone during acquisition) to perform acquisition. WOLF acquires data from the iPhone using a logical copy of the data and presumably cannot recover deleted data. The following data is recovered:

- Handset Info
- Contacts
- Call Logs
- Messages
- Internet Info
- History
- Photos
- Music/Videos

2. Installation

To acquire data from an iPhone, WOLF must be installed on a Mac OX 10.4.11 or higher, running on an Intel CPU. The software requires a dongle and an activation process. CodeMeter software is used to activate and verify the dongle and the installation of this software is straight forward. After running the software and creating your activation key, you compress the resulting file (zip) and then email it to `<activations@sixthlegion.com>`.

Figure 2.1. Code Meter Control Center

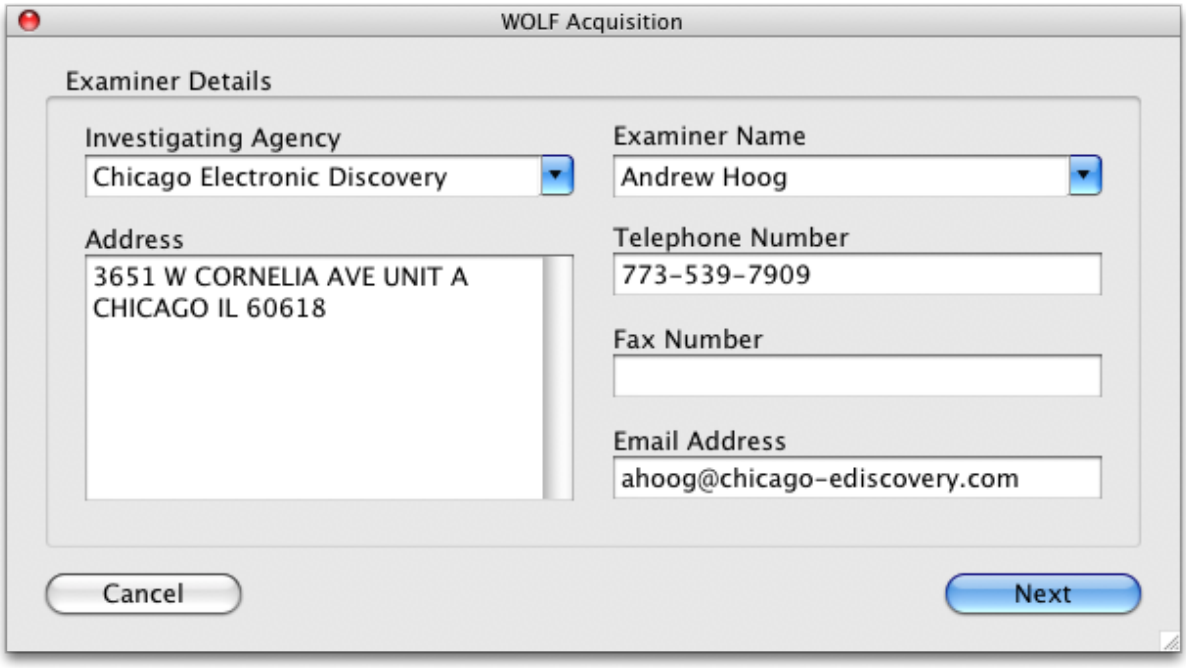


I sent the activation key around 10:30PM on a Sunday evening and received the reply around 2:40PM the following afternoon. After running the License Update wizard again and uploading the licensing file, the software was ready to use.

3. Forensic Acquisition

Performing a forensic acquisition of an iPhone using Wolf is quite intuitive. After the application is properly licensed, you simply run WOLF and click Acquire. You are prompted to input the examiner detail information.

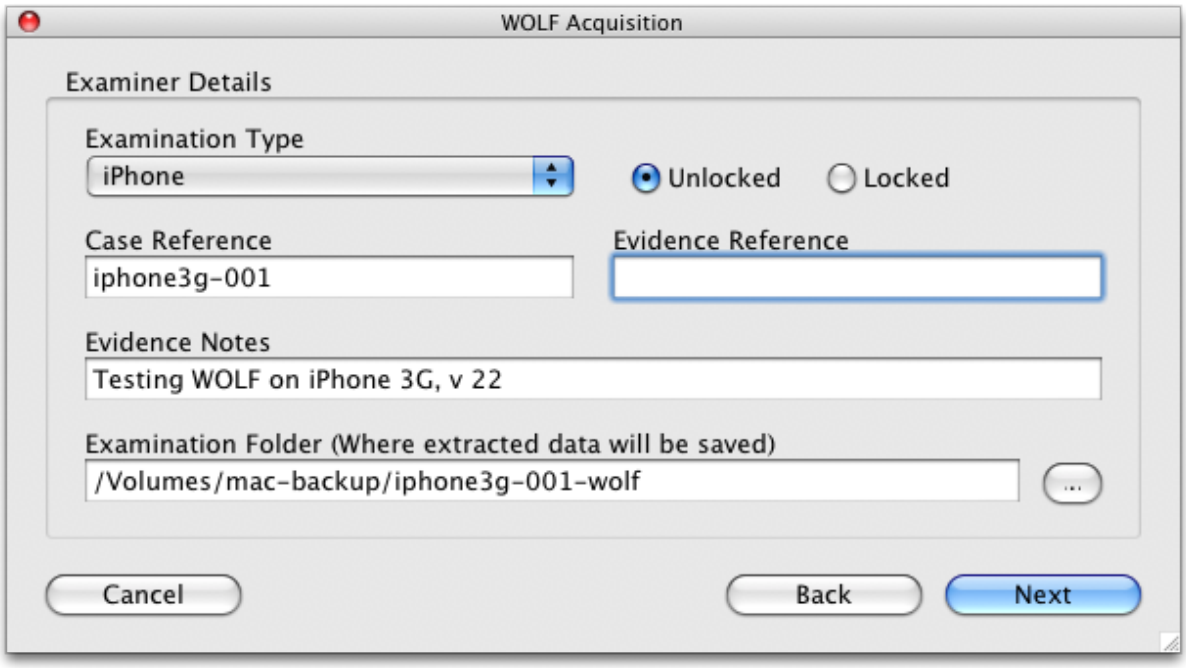
Figure 2.2. Wolf Acquisition



Conveniently, WOLF remembers this information and allows you to select the Agency and Examiner from previous investigations. This is one example that, while minor, shows how an intuitive user interface assists in at least the speed of an acquisition, if not more.

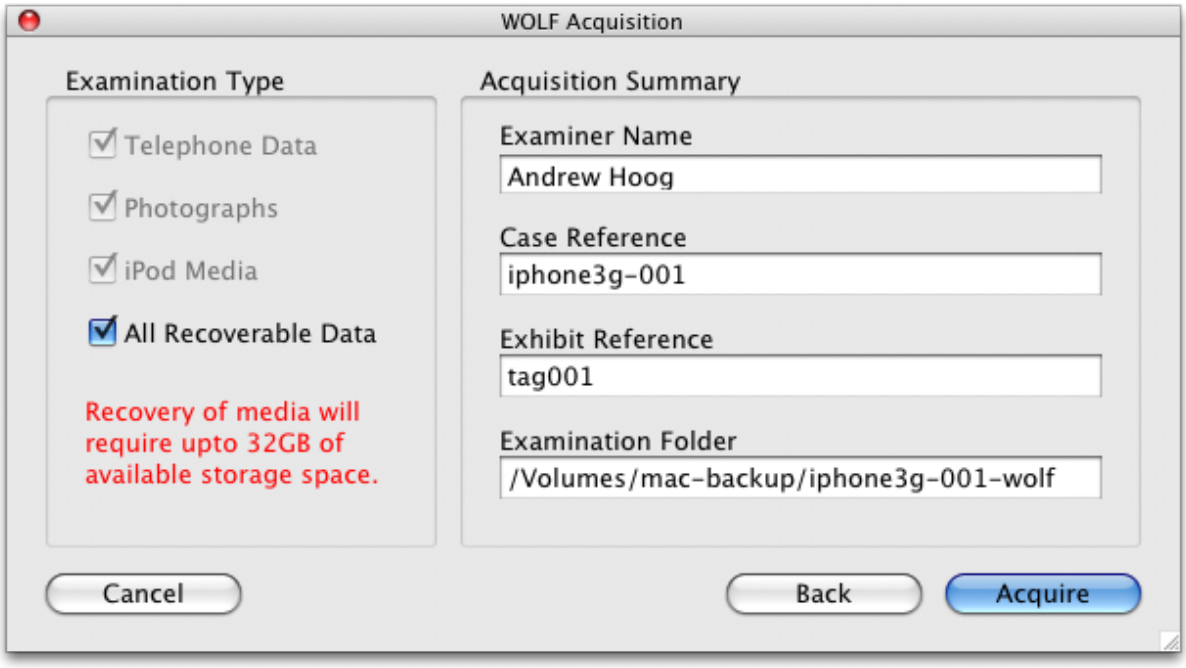
Next, you supply information about the device you are going to acquire (iPhone, iPod Touch or Backup Folder) along with additional descriptive information. An important note about WOLF is that they state they can circumvent the pass code if set on an iPhone, provided you have access to the computer the iPhone was synced with.

Figure 2.3. Wolf Acquisition 2



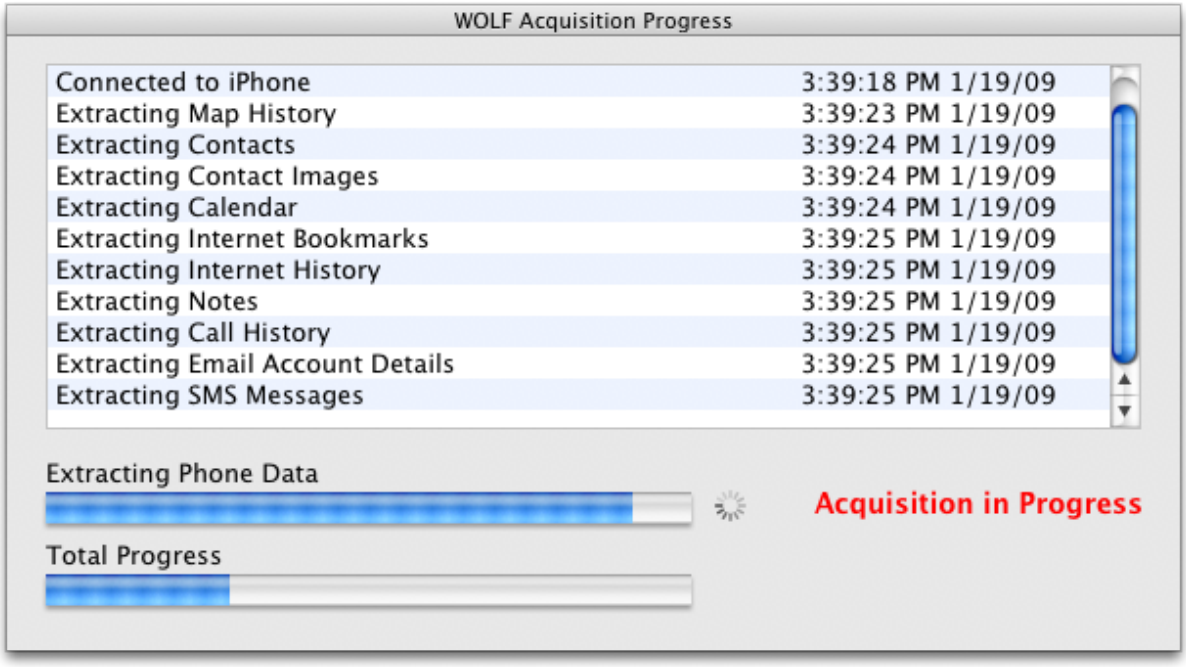
After you confirm this information, you select the type of data you wish to acquire.

Figure 2.4. Wolf Acquisition 3



Once you hit Acquire, the acquisition begins.

Figure 2.5. Wolf Acquisition Progress



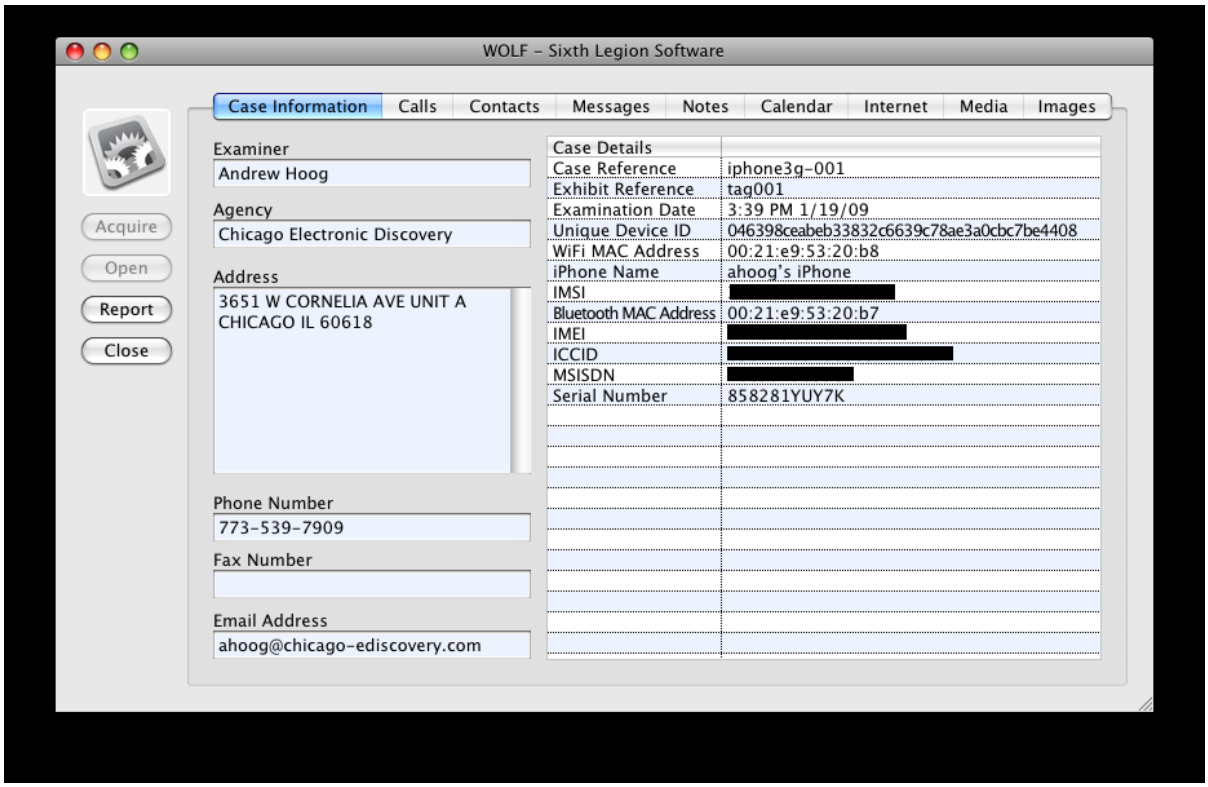
The acquisition only took about 2 minutes. A direct iPhone acquisition process has advantages over relying on the analysis of the backup files which may be out of date or unavailable. However, WOLF does support analyzing the backup files if they are available to you so I tested both the direct and backup acquisition methods.

4. Results and Reporting

After the acquisition is complete, you can view the results within the application or run a series of reports which save the information to HTML. For the purpose of this paper, I show the results directly from the applications as it is more effective than scrolling through long reports.

Like other products, WOLF acquired and accurately presented basic phone information.

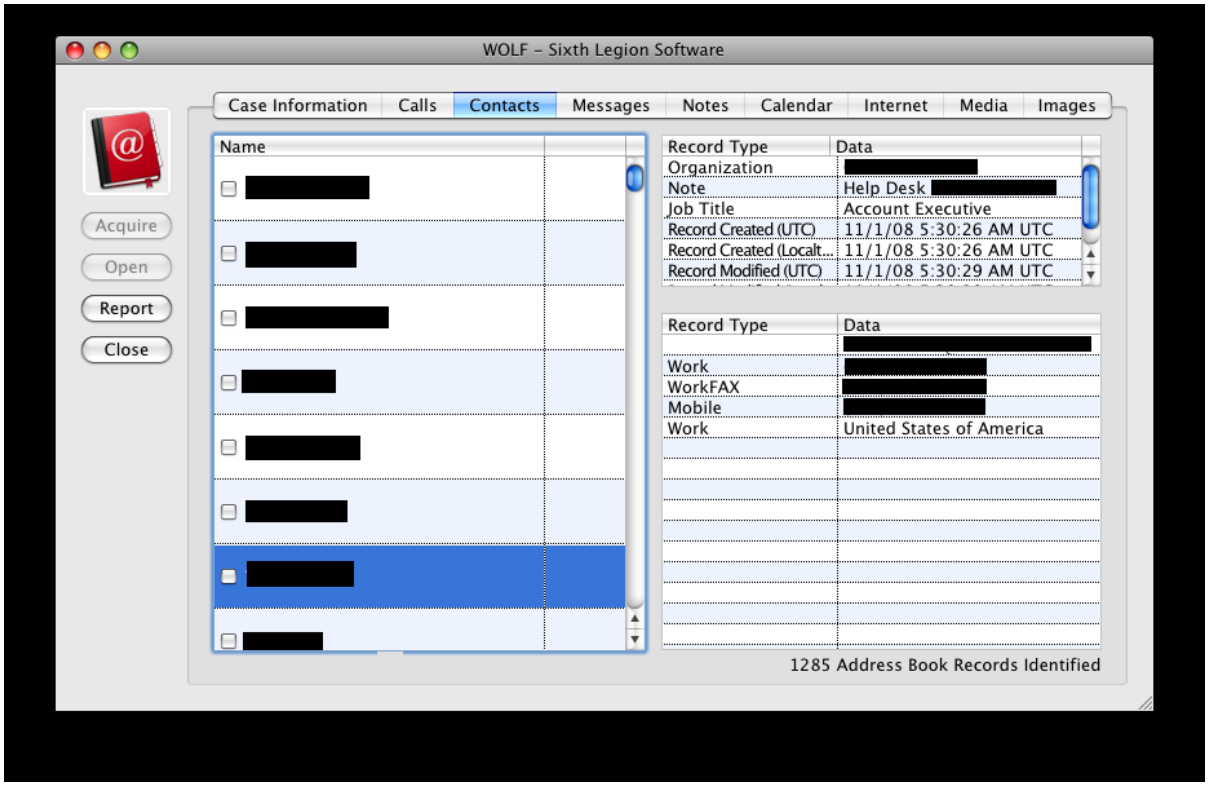
Figure 2.6. Wolf Case Information



However, quite significantly WOLF was unable to recover the call logs or SMS messages. We are working with WOLF at this time to identify the issue and I'm hopeful a resolution is forthcoming.

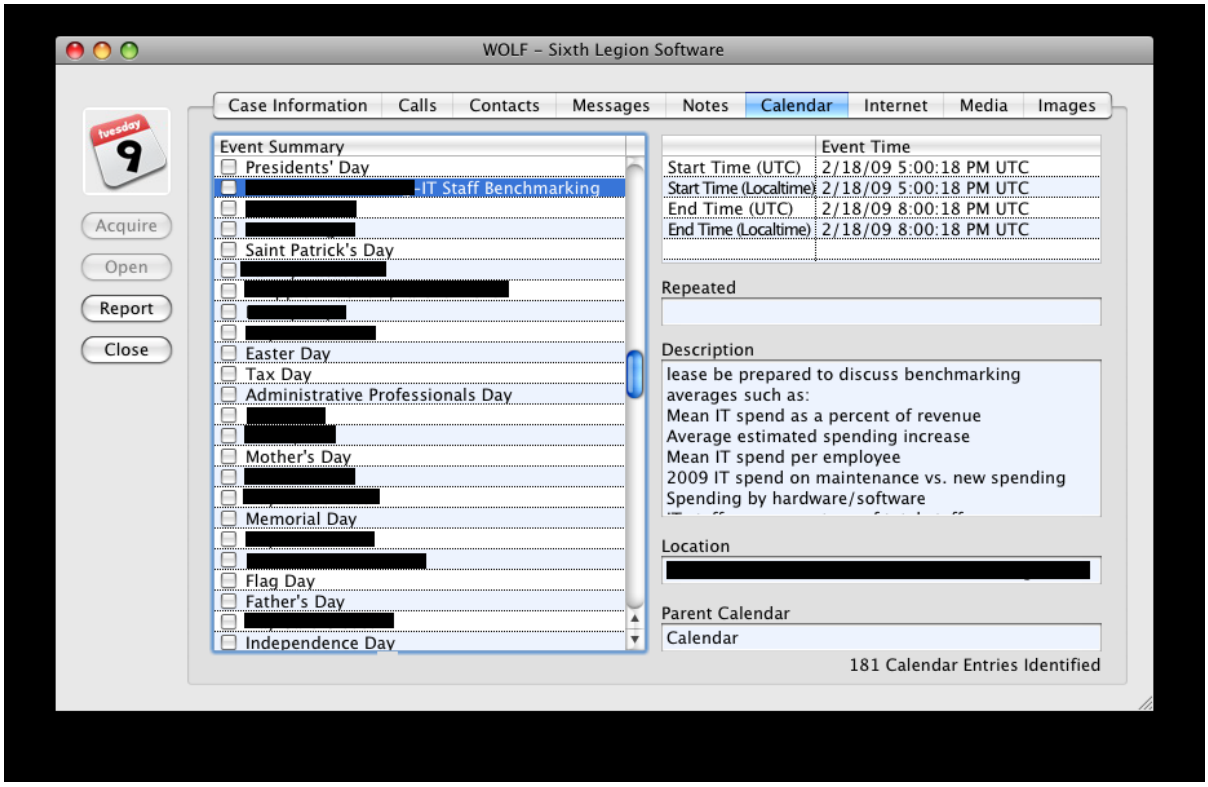
All contacts were recovered and WOLF even provides MAC (Modified, Accessed, Changed) times which is a very helpful feature not found in other solutions.

Figure 2.7. Wolf- Contacts



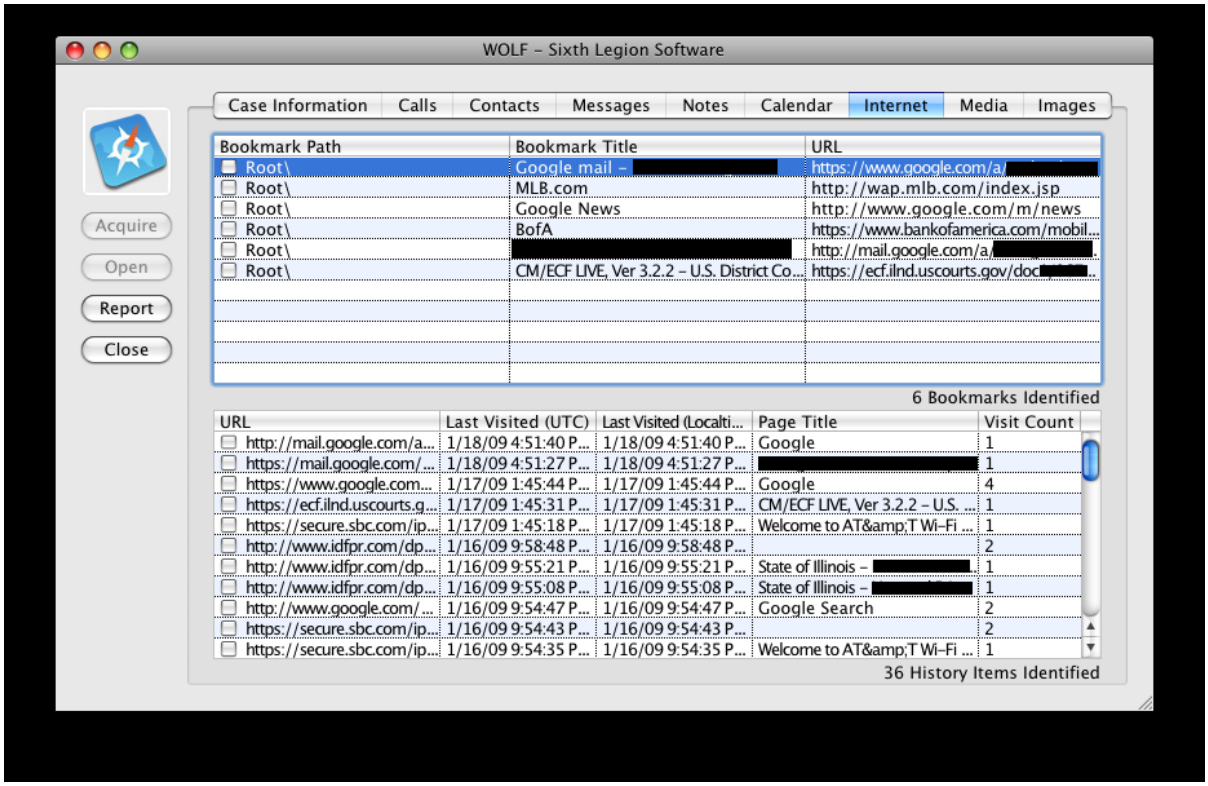
Notes were successfully recovered as were Calendar events, again with MAC times.

Figure 2.8. Wolf - Calendar



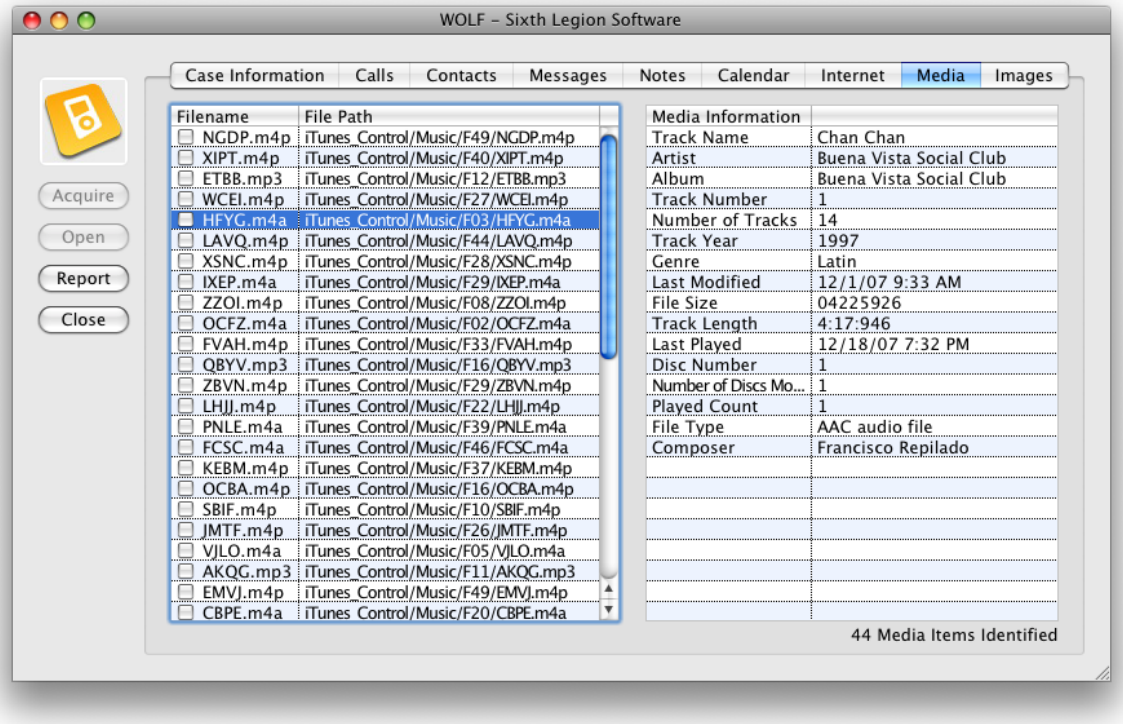
WOLF was able to recover Bookmarks and browsing history which is something several other tools struggled with.

Figure 2.9. Wolf - Internet



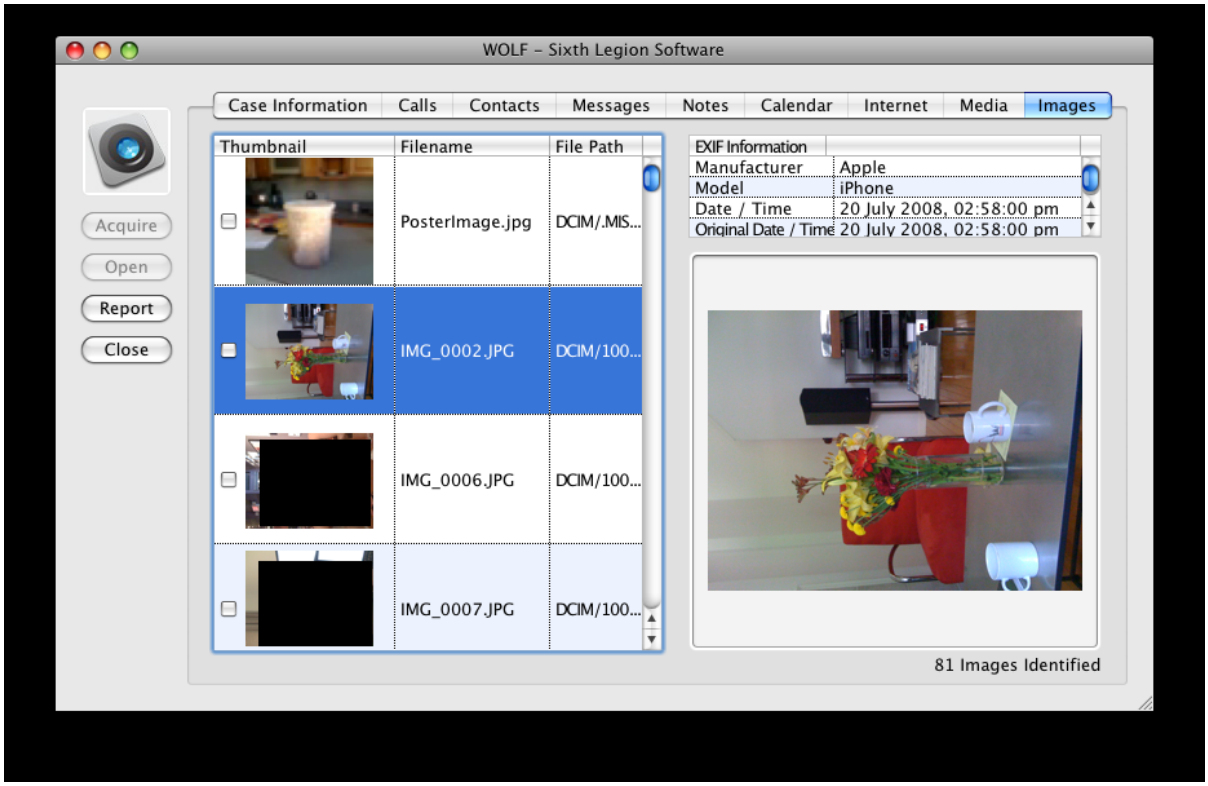
WOLF recovered all songs on the device (regardless if they had DRM enabled, which prevent recovery with some products).

Figure 2.10. Wolf - Media



Finally, WOLF, unlike several other products, was able to recover not only the pictures taken from the iPhone but 31 pictures that were uploaded through iTunes.

Figure 2.11. Wolf - Images



5. Matrix of Results

The following are the results from the WOLF tests.

Table 2.1. Wolf Matrix of Results

Scenario	WOLF - direct	Ranking	WOLF - back-up	Ranking	WOLF Total	Results
Call Logs	0	0	0	0	0	Miss
SMS	0	0	0	0	0	Miss
Contacts	1282	3	0	0	3	Meet
Email	0	0	0	0	0	Miss
Calendar	3070	3	3070	3	3	Meet
Notes	1	3	1	3	3	Meet
Pictures	42	2	43 (2 icons, not synced images)	2	2	Below
Songs	44	3	0	3	3	Meet
Web History	2	3	2	3	3	Meet
Bookmarks	5	3	5	3	3	Meet
Cookies	0	0	0	0	0	Miss
App Info	0	0	0	0	0	Miss
Google Maps	0	0	0	0	0	Miss

Voicemail	0	0	0	0	0	Miss
Passwords	0	0	0	0	0	Miss
Plists/XML	0	0	0	0	0	Miss
Phone Information	Yes	3	Yes	3	3	Meet
Video	1	3	0	0	3	Meet
Podcasts	1	3	0	0	3	Meet
Speed Dials	0	0	0	0	0	Miss
VPN	0	0	0	0	0	Miss
Bluetooth	0	0	0	0	0	Miss
GPS	0	0	0	0	0	Miss
File Hashes	0	0	0	0	0	Miss
YouTube	0	0	0	0	0	Miss
HTML	0	0	0	0	0	Miss
Office Documents	0	0	0	0	0	Miss

6. Conclusions

WOLF is an intuitive and fast forensic solution for the iPhone. Once the problems with the Call Log and SMS issues are addressed, WOLF is a solid forensic solution for the iPhone. However, other products which perform logical file system acquisition allow direct access to SQLite files (and thus the ability to recover some deleted data) as well as other critical files. This access provides the analysts with an opportunity to recover more information.

The following ranking establishes WOLF's overall rating of 1.8 on the four criteria established at the beginning of this white paper.

Table 2.2. Wolf Rankings

Area	Weight	Rank
Installation	0.1	3.0
Acquisition	0.2	2.5
Reporting	0.3	3.0
Accuracy	0.4	1.1
TOTAL		1.8

Chapter 3. Cellebrite UFED (3.0/5.0)

1. Summary (from Company Information)

The Cellebrite UFED Forensic system is a stand-alone device capable of acquiring data from mobile devices (~1600) and storing the information on a USB drive, SD card or PC. UFED also has a built-in in SIM card reader and cloner. The ability to clone a SIM card is a powerful feature as you can create and insert a clone of the original SIM and the phone will function normally. However it will not register on the mobile carrier's network, eliminating the need for Faraday bags and the possibility that the data on the phone will be updated (or erased). The UFED package ships with about 70 cables for connecting to most mobile devices available today. Connection protocols include serial, USB, infrared and Bluetooth although I only utilized the USB approach.

Cellebrite also distributes the UFED Report Manager which provides an intuitive reporting interface and allows the user to export data/reports into Excel, MS Outlook, Outlook Express, and CSV or to simply print the report.

The UFED device fully supports Unicode and thus can process phones with any language enabled. Also, the following data types are extracted:

- Phonebook
- Text Messages
- Call History (Received, Dialed, Missed)
- SIM ID Cloning
- Deleted Text Messages off SIM/USIM
- Audio Recordings
- Videos
- Pictures
- Phone Details (IMEI/ESN phone number)

There is beta support for logical extraction of the phone's file system though a feature Cellebrite labels Memory Dump. In the case of the iPhone, this is achieved without jailbreaking the iPhone. This allows for greater analysis of data on the iPhone.

2. Installation

The UFED package arrived in a soft case containing the UFED device, manuals/CD, USB Bluetooth radio (Cambridge Silicon Radio Ltd.), 250MB USB drive and roughly 72 cables for connecting to supported devices. The manual was sparse but sufficient and very straightforward.

To start things off, I decided to make sure the UFED software was update to date. There are options to update via a PC, USB, SD card or via the Internet.

I decided to test the convenient online upgrade feature. I powered the UFED on and had to first set the date/time which was simple. Next I connected it via Ethernet to a switch running DHCP and went under Services ->Upgrade -> Upgrade Application Now and selected HTTP Server as the source. On my first attempt, the download froze prior to completion and I eventually rebooted the device. The second time I connected it to a different switch and the upgrade went flawlessly. A few minutes later I was on the latest Application software (1.1.0.5) which supplies the UFED application and support for the various phones. Cellebrite seems to add new phone support often and a forensic examiner should check for updates often.

The UFED contains two other pieces of software termed Images. One dubbed Tiny contains the core system software. The other image named Full contains additional core system software. Both were up to date (1.0.2.1 and 1.0.2.2 respectively) and I am unclear if this was due to the Application update I initially performed or was shipped as such. The update process for the Image software is under a separate menu in Services and I suspect the updates are performed independently. One minor note, when I checked the manual online , the PDF with update direction for UFED instead opened a UME-36Pro PDF. The platforms are likely very close and this is also probably easily remedied by searching their site or contacting technical support.

3. Forensic Acquisition

The acquisition of the 3G iPhone was extremely simple and fast on UFED. After powering the device on, I selected Extract Phone Data, Apple, iPhone 2G/3G, USB disk drive (destination), Content types (I pressed F2 to select all including Call Logs, Phonebook, SMS, Pictures, Videos and Audio/Music) and was then instructed to connect the iPhone to the source port with Connect cable 110 and the USB Disk Drive into the target port. The extraction took 6 minutes and was copied into an automatically create folder on the attached USB drive.

I also performed a Memory Dump of the iPhone, which is marked Beta on the main screen. Following the prompts, I was instructed to attach the iPhone and USB drive as before. Bear in mind you will be performing a full backup of the iPhone (possibly 16GB) so ensure you have enough space on the USB drive. The Memory Dump failed the first few times but eventually succeeded. It would pause for several minutes while acquiring large files. I opted to let the acquisition run overnight and from the previous attempts, I knew it took several hours. The resulting data was written to the attached USB drive in an automatically created folder.

4. Results and Reportiong

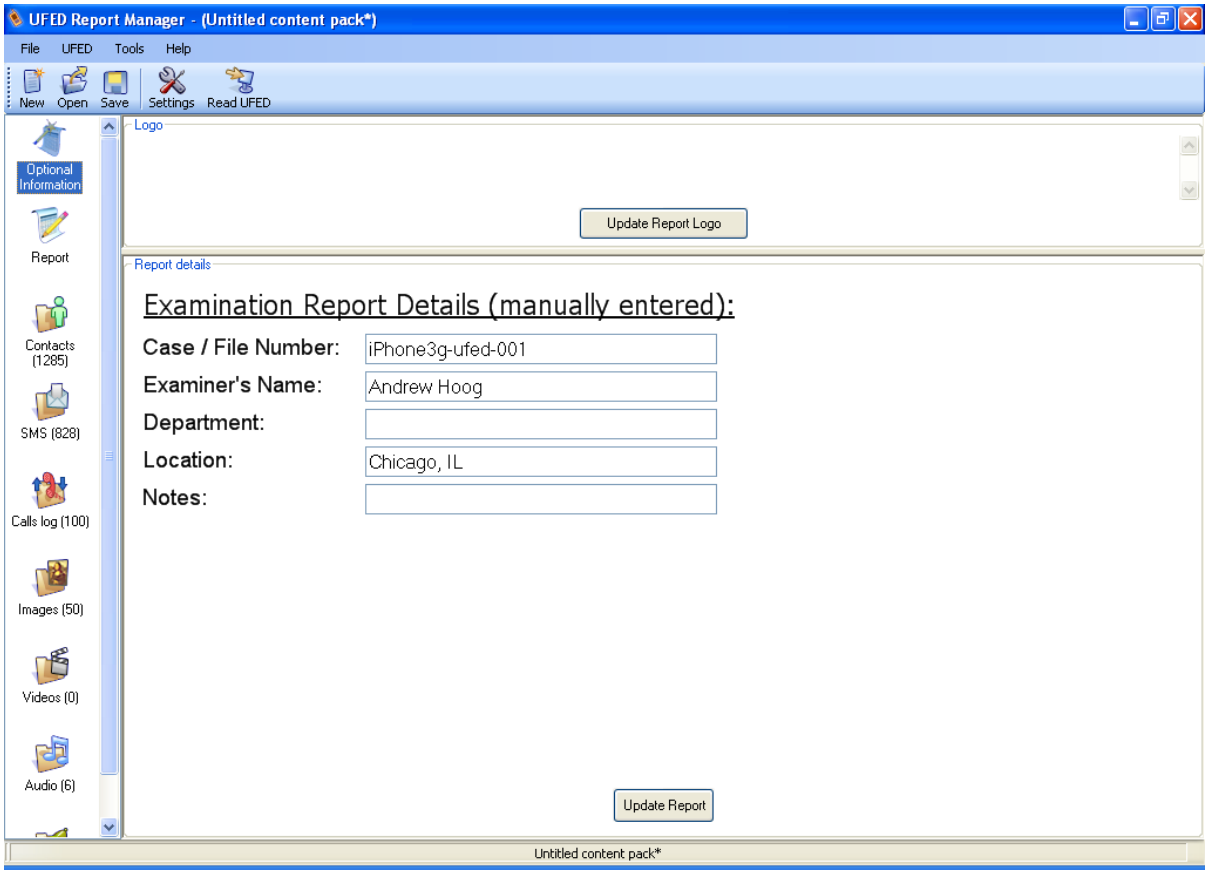
Since UFED has both a “standard” acquisition process and a memory dump (Beta) option, please note there are two sections detailing the results.

4.1. Standard Acquisition Results

The standard acquisition resulted in a roughly 60MB folder containing the extracted audio and images, proprietary files with extensions such as .SMS and .PBB and reports in both HTML and XML containing the following sections: Contacts, SMS, Call Logs, Images, Ringtones (Not Supported), Audio and Video. I was able to easily import this folder into the UFED Report Manager for a more user friendly interface.

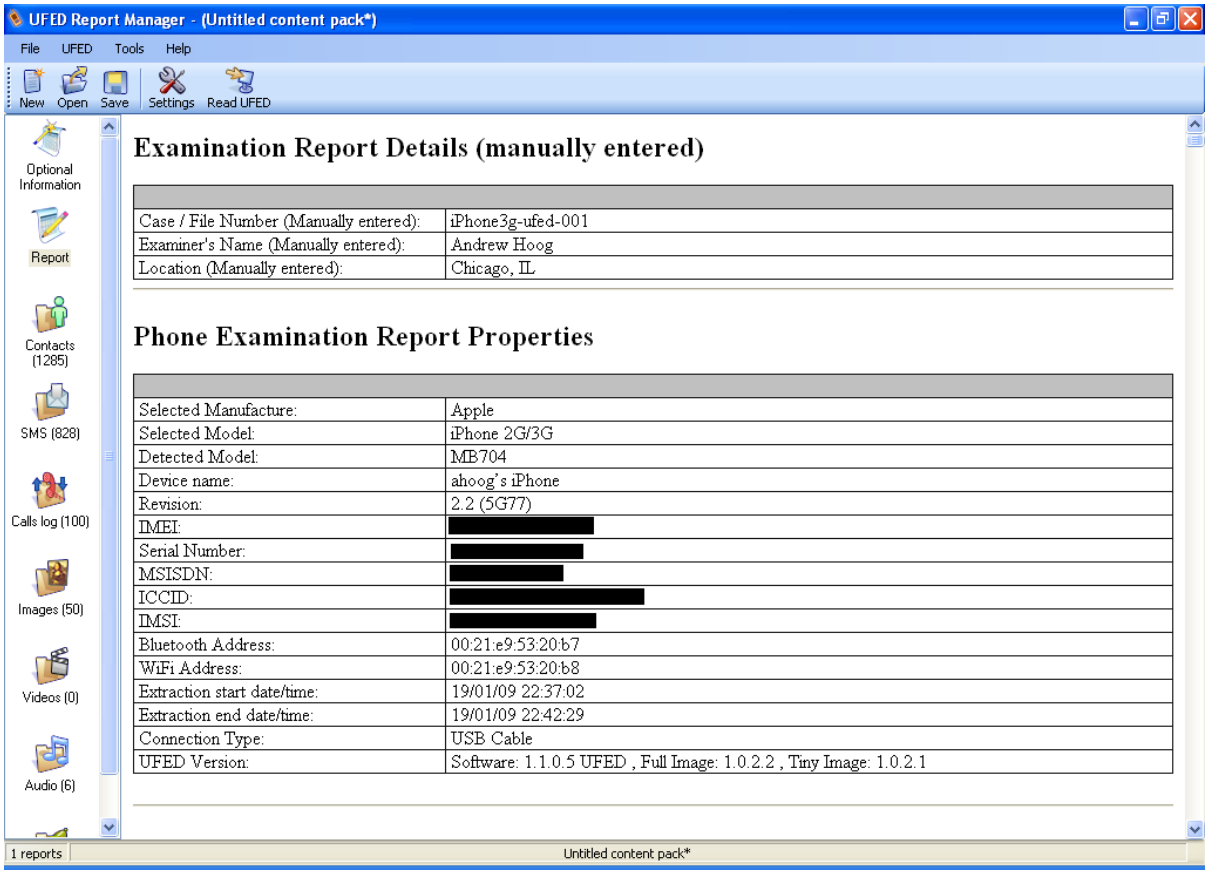
When you run UFED Report Manager, you can import the data from the USB drive by clicking on File -> Open Extraction (from folder). You can then add Optional Information including case, examiner and other investigation information.

Figure 3.1. UFED Report Manager



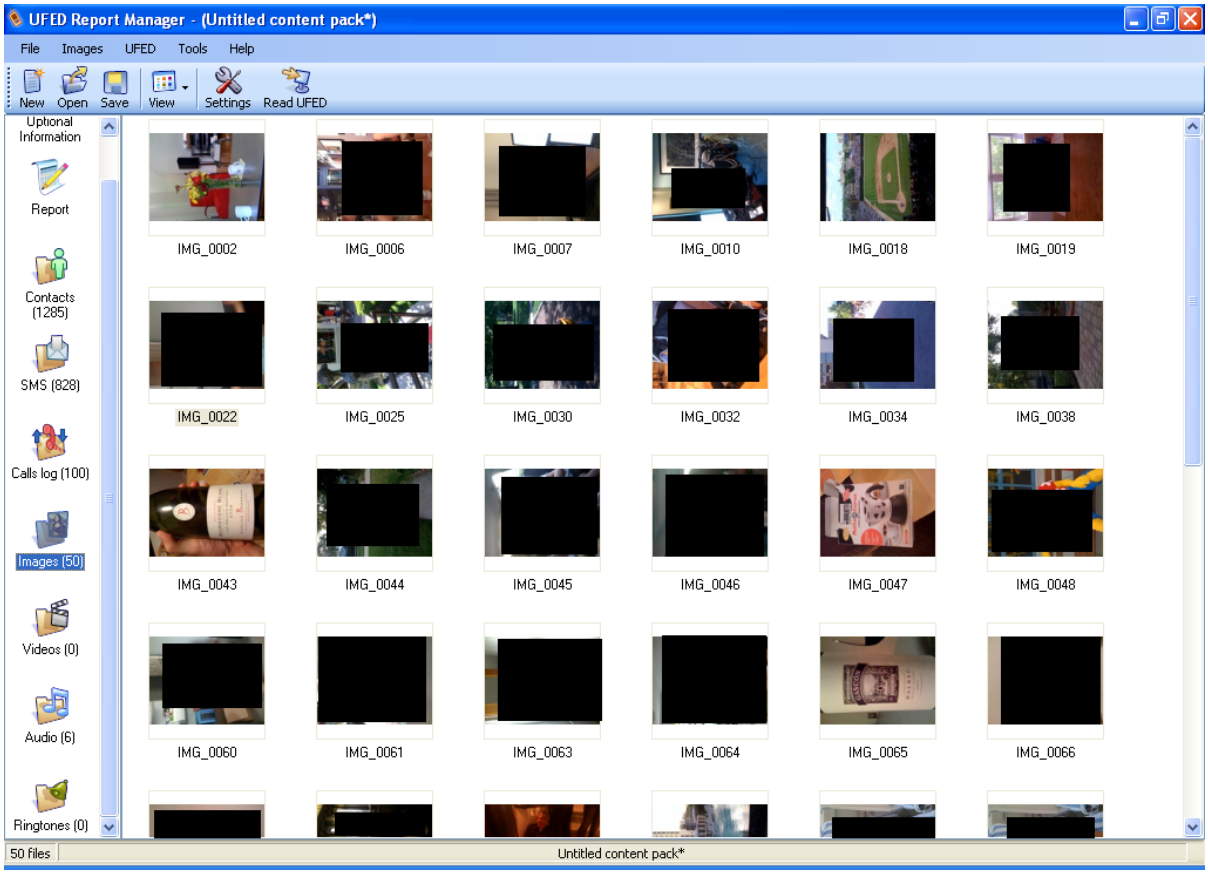
Along the left hand side, you can see the major areas of focus including Optional Information, Report, Contacts, SMS, Call Log, Images, Videos, Audio and Ringtones. The following shows some basic information included in the Report section.

Figure 3.2. UFED Report Properties



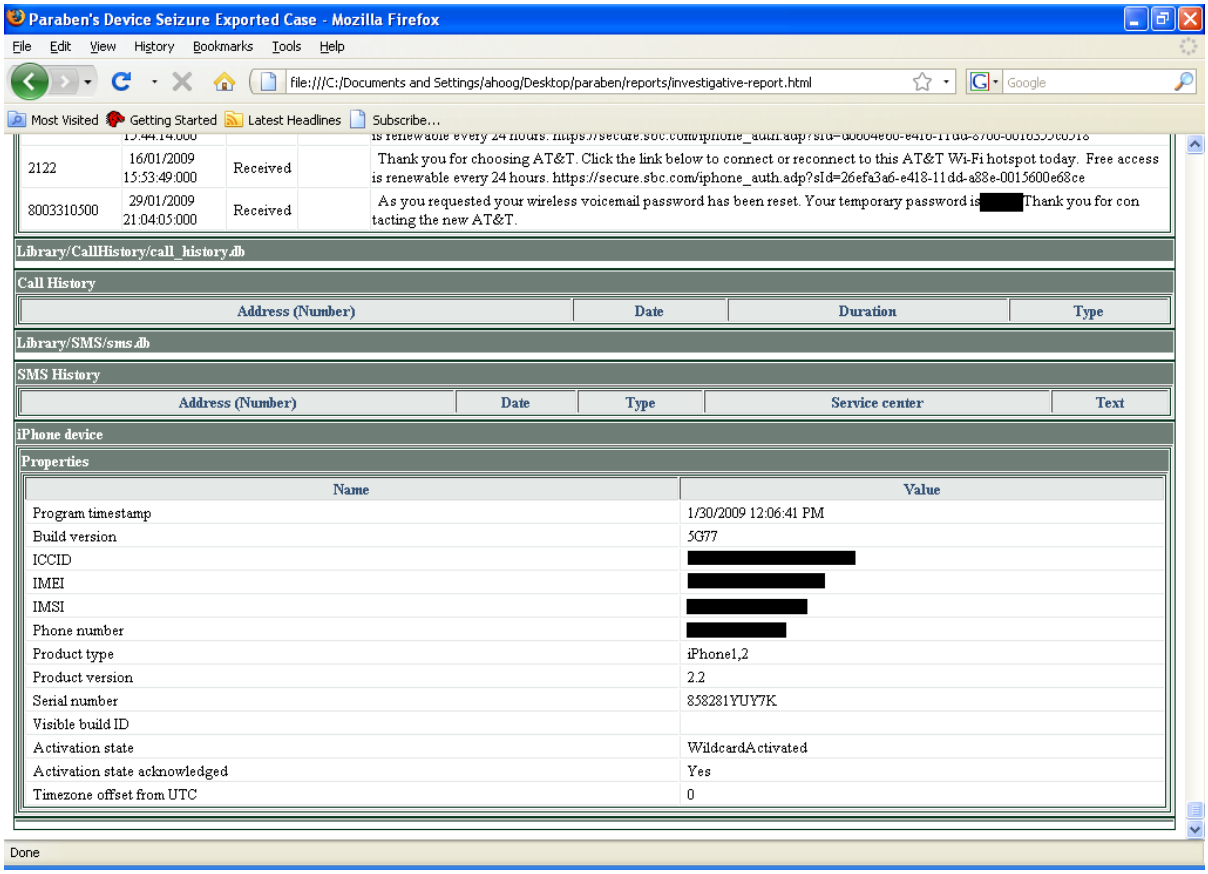
The Images section previews all images found.

Figure 3.3. UFED Images



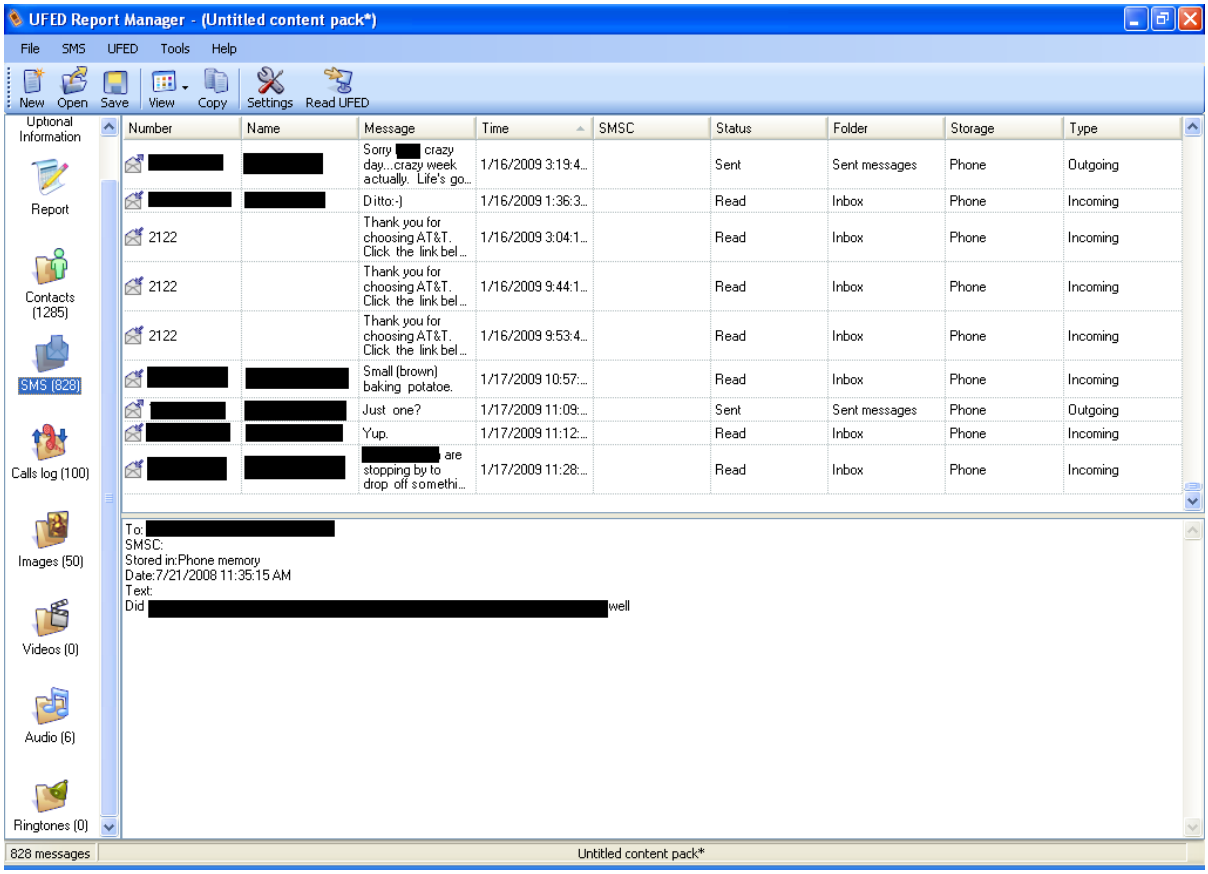
And the Calls Log shows the type of call (Outgoing, Incoming or Missed) as well as the Name (if found in Contacts), phone number, date/time and duration.

Figure 3.4. UFED Call Logs



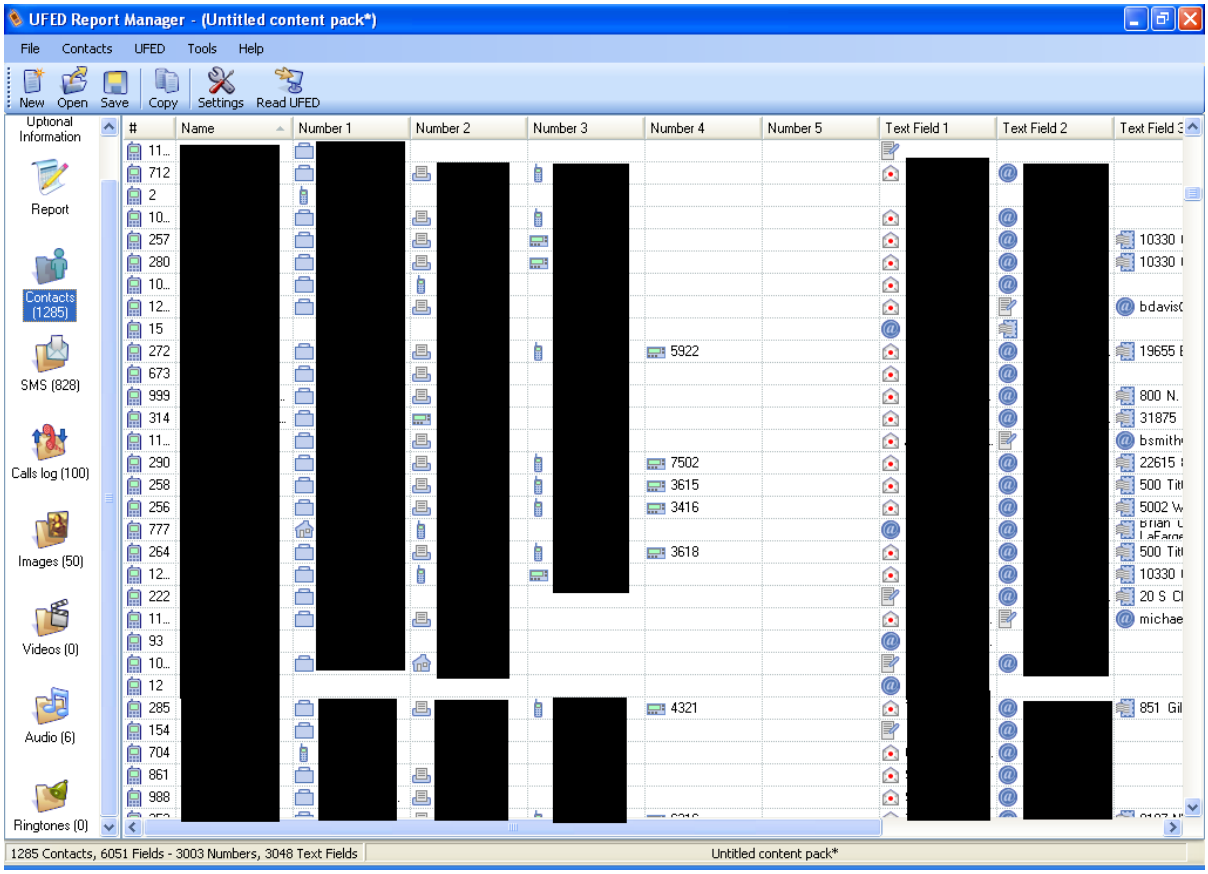
The SMS section shows the full set of messages and a detailed message window. The details include Number, Name, Message, date/time, SMSC, Status (Send, Read, Unsent, etc.), Folder, where it was stored and the type (Incoming or Outgoing).

Figure 3.5. UFED SMS



The final screenshot I took was of the Contacts information, including Name, various numbers, and text fields including Company Name, email address, notes, etc.

Figure 3.6. UFED Contacts



The data can be extracted into Excel (or CSV) as well as importing directly into Outlook or Outlook Express. While this is an interesting feature, I can't think of a situation in which I would import the information into Outlook.

4.2. Memory Dump (Beta) Results

The Memory Dump acquisition (logical) was 282MB and included 382 files. The top level folder including the following 3 subfolders: AFC Service, Backup Service and Lockdown Service. The majority of the data were songs under the AFC Service -> iTunes_Control -> Music directory. The Backup directory contains important database, Plist and other files allowing a more complete recovery of data from the iPhone.

The results of the Memory Dump (Beta) from Cellebrite are actually quite promising. Of the 382 files extracted, it included the following:

- 180 property list (Plist) files containing a wealth of user/configuration information
- SQLite databases including SMS, Notes, Call History, Calendar, Address Book, iTunes Extras, keychain and several App Store program's data (TwitterFon, Facebook and Wordpress)
- 14 XML files (Wordpress)
- 2 ASCII files
- 41 pictures (JPEG) and associated thumbnails (JFIF files with .thm extension)
- 44 MPEG4 songs, 1 Podcast and 1 Video
- 37 miscellaneous data files (requires additional analysis)

The keychain-2.db SQLite database contained information about the networks the user attached to including Wi-Fi, VPN, Bluetooth and the Apple iTunes Store ID. In addition, other SQLite databases under the Documents

folder contained information from some App store programs such as TwitterFon and Facebook and could provide valuable information to the investigator. All of this was done without jailbreaking the phone, a major plus for any forensic investigation. By analyzing the SQLite databases and Plist files, an investigator can recover deleted information and important configuration and usage information.

However, in the Memory Dump, the call_history.db and sms.db SQLite databases were empty. Since this review takes into account the combined acquisition results, UFED was not penalized for this since the standard acquisition acquired the call and SMS data accurately.

5. Matrix of Results

The following are the results from the UFED tests.

Table 3.1. UFED Matrix of Results

Scenario	UFED - direct	Ranking	UFED - Memory Dump	Ranking	UFED Total	Results
Call Logs	100	3	0	0	3	Meet
SMS	262	3	0	0	3	Meet
Contacts	1282	3	1282 (14 w/ images)	3	3	Meet
Email	0	0	some account info, folder info, etc.	1	1	Below
Calendar	0	0	3070	3	3	Meet
Notes	0	0	2(1 recovered in SQLite db)	5	5	Above
Pictures	42	2	41	3	3	Meet
Songs	6	1	44	3	3	Meet
Web History	0	0	2	3	3	Meet
Bookmarks	0	0	5	3	3	Meet
Cookies	0	0	1	2	2	Below
App Info	0	0	5	3	3	Meet
Google Maps	0	0	5 histories	3	3	Meet
Voicemail	0	0	0	0	0	Miss
Passwords	0	0	7	3	3	Meet
Plists/XML	0	0	157	3	3	Meet
Phone Information	Yes	3	Yes	3	3	Meet
Video	1	3	1	3	3	Meet
Podcasts	1	3	1	3	3	Meet
Speed Dials	0	0	4	3	3	Meet
VPN	0	0	Yes	3	3	Meet
Bluetooth	0	0	Yes	3	3	Meet
GPS	0	0	0	0	0	Miss
File Hashes	0	0	0	0	0	Miss
YouTube	0	0	50 URLs	4	4	Above
HTML	0	0	0	0	0	Miss

Office Documents	0	0	0	0	0	Miss

6. Conclusions

Cellebrite's UFED is an excellent product for forensic analysis of the iPhone. By providing two acquisition methods, the investigator can recover a significant portion of the data on the iPhone. The device is also very simple to use, easy to update, performs acquisitions quickly and is portable. The firmware is updated often to support new phones and functionality and the support department was efficient and professional.

The following ranking establishes UFED's overall rating of 3.0 on the four criteria established at the beginning of this white paper.

Table 3.2. UFED Rankings

Area	Weight	Rank
Installation	0.1	5.0
Acquisition	0.2	4.5
Reporting	0.3	3.0
Accuracy	0.4	2.4
TOTAL		3.0

Chapter 4. Paraben Device Seizure (2.9/5.0)

1. Summary (from company information)

Paraben Device Seizure (DS) is a forensic software tool that performs acquisitions on over 2,700 handheld devices (including phones, PDAs and GPS devices) and runs on Microsoft Windows. The package is designed to support the full acquisition and investigation process. Paraben stresses their ability to perform physical acquisition vs. logical ones as it provides the ability to recover deleted files and other important information. They have several packages which include the DS software and various cables for phone acquisitions.

Paraben also has a product (Paraben SIM Card Seizure) which allows an analyst to read or optionally clone a SIM however this does not ship with DS or the entry level packages. If analyzing or cloning the SIM card directly is important to you, make sure you either purchase this separately or a bundle that includes it.

The DS software allows an investigator to perform the acquisition, view data in various formats (ASCII, Hex, file and data viewers, etc.), bookmark important data, export data and run various reports. Paraben states DS can extract the following from cell phones (varies by model):

- SMS History (Text Messages)
- Deleted SMS (Text Messages)
- Phonebook (both stored in the memory of the phone and on the SIM card)
- Call History
  - Received Calls
  - Dialed Numbers
  - Missed Calls
  - Call Dates and Durations
- Date Book
- Scheduler
- Calendar
- To-Do List
- File system (physical memory dumps)
  - System Files
  - Multimedia Files (Images, Videos, etc.)
  - Java Files
  - Deleted Data
  - Quicknotes
- E-mail

2. Installation

My initial installation of Device Seizure (DS) was version 2.2 and I was supplied with a dongle which required activation (a software license key option is also available). I did run into a few problems with the installation and activation and had to work through their Support group to resolve. This was a bit frustrating especially when the Support website would not email me my password (a problem I still have today). However, a phone call to Technical Support resolved the issues and I was up and running shortly thereafter.

I also should note that you need to install their DS Driver package which contains the drivers for various phones and syncing software such as ActiveSync and iTunes. This process was also cumbersome and required me to remove previous versions of installed software. In the end, I had to remove iTunes 8.0.2 and the driver package installed version 7.4.2.4. This required reboots and, frankly, large changes to my system. Since Paraben stated DS would not work in a VMWare environment, I was only left with the option to change the core Window XP install on my dual-boot workstation. Overall, the installation was a difficult and frustrating.

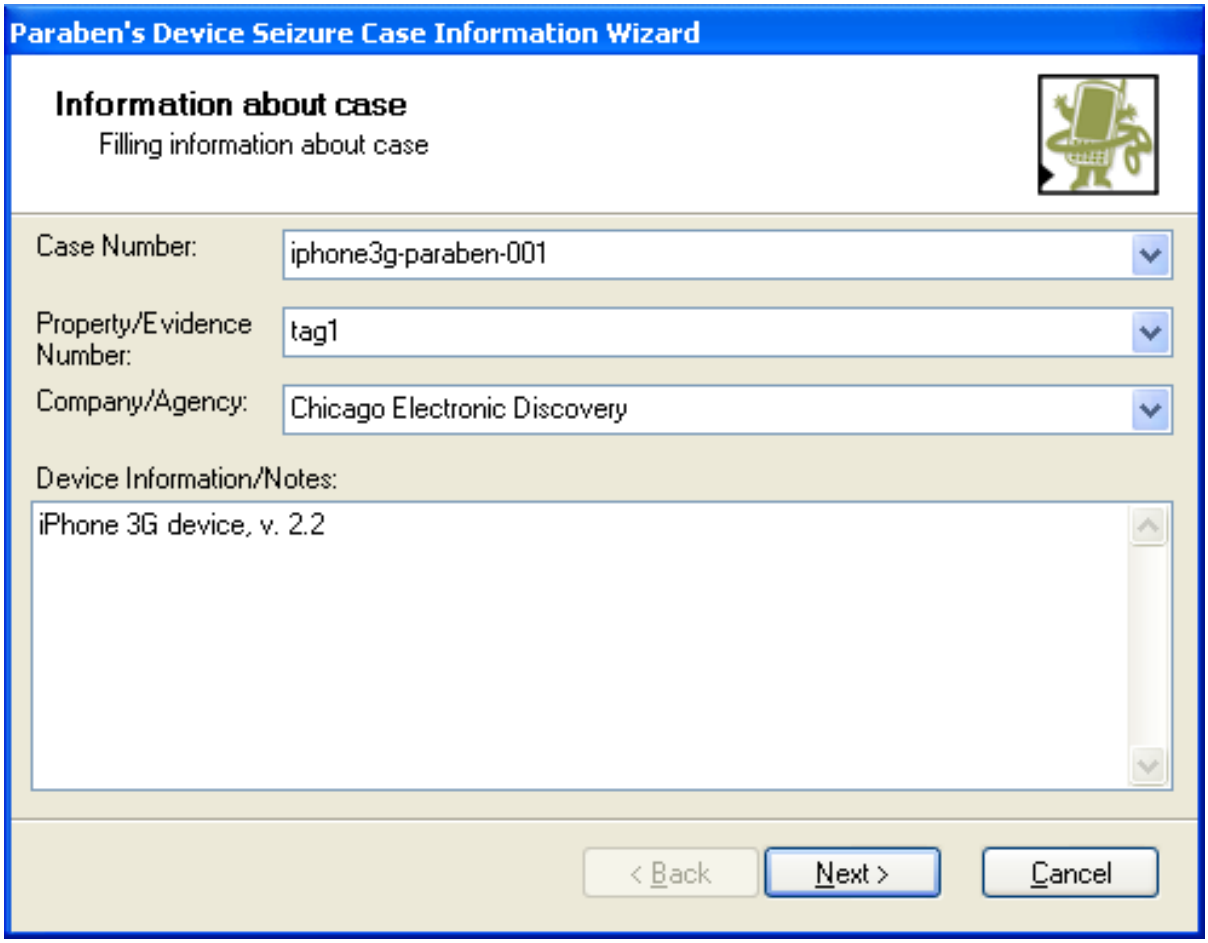
My initial acquisitions of the iPhone with DS 2.2 failed and while I was working with Technical Support, version 3.0 was released. There were anomalies again with the download and installation process. The DS 3.0 install simply ran the currently installed installation process and made no modifications to the system. After I completely uninstalled DS 2.2, I was able to install the 3.0 version. The dongle then needed to be updated and it pointed me to an invalid URL for downloading DS 3.0 (which I had already done so it was not a big deal). After I worked through these issues, though, things ran smoothly.

3. Forensic Acquisition

With the installation behind me, I was ready to start the acquisition. Paraben made this quite easy however there were multiple approaches to choose from and the Help section was not clear on the differences. After speaking with their Technical Support department, I had a better understanding of the two approaches available. An updated Help section on the iPhone would be a welcomed change and would ease iPhone acquisitions for new users.

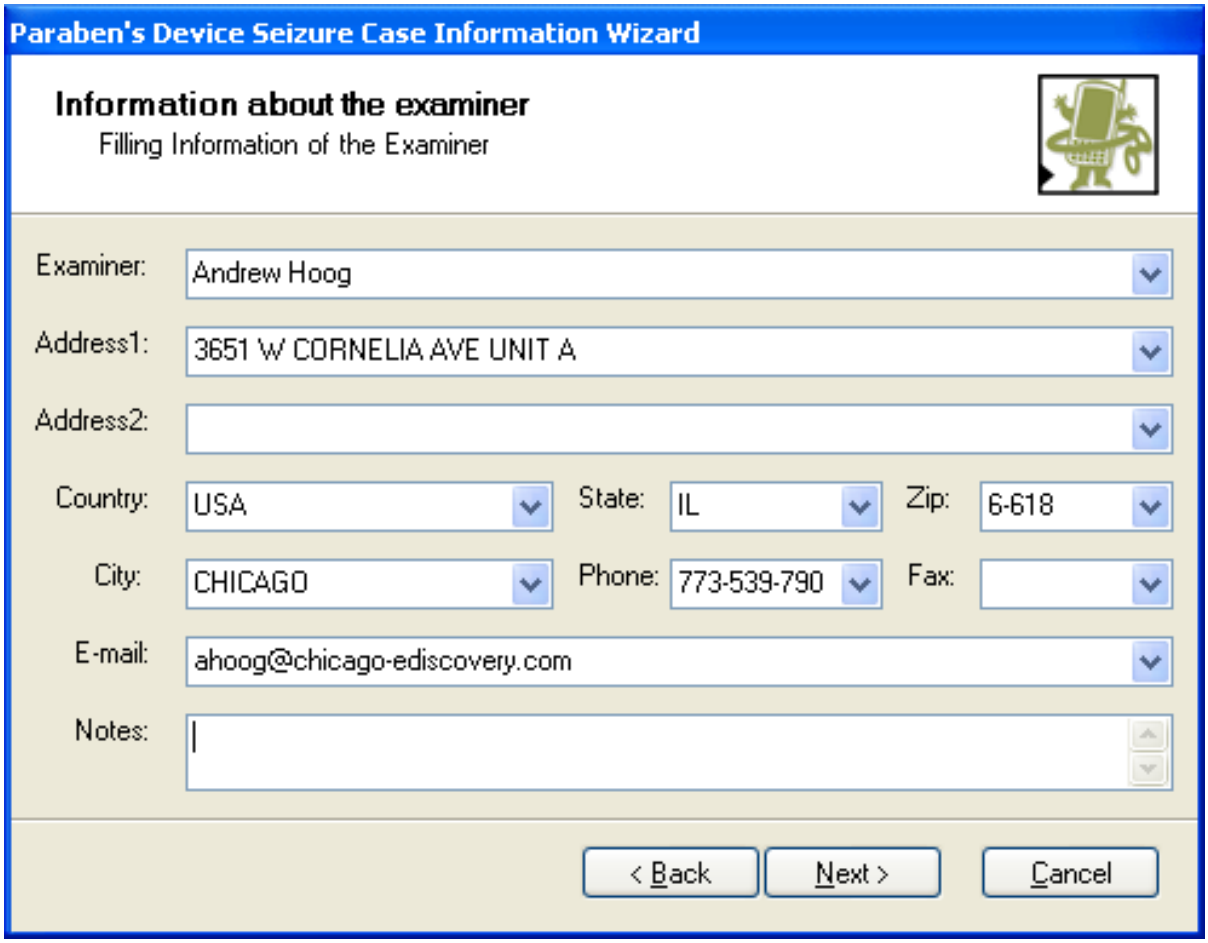
After DS starts, you create a new case and enter basic information.

Figure 4.1. Paraben Device Information



Next, you specify information about the examiner.

Figure 4.2. Paraben Examiner Information

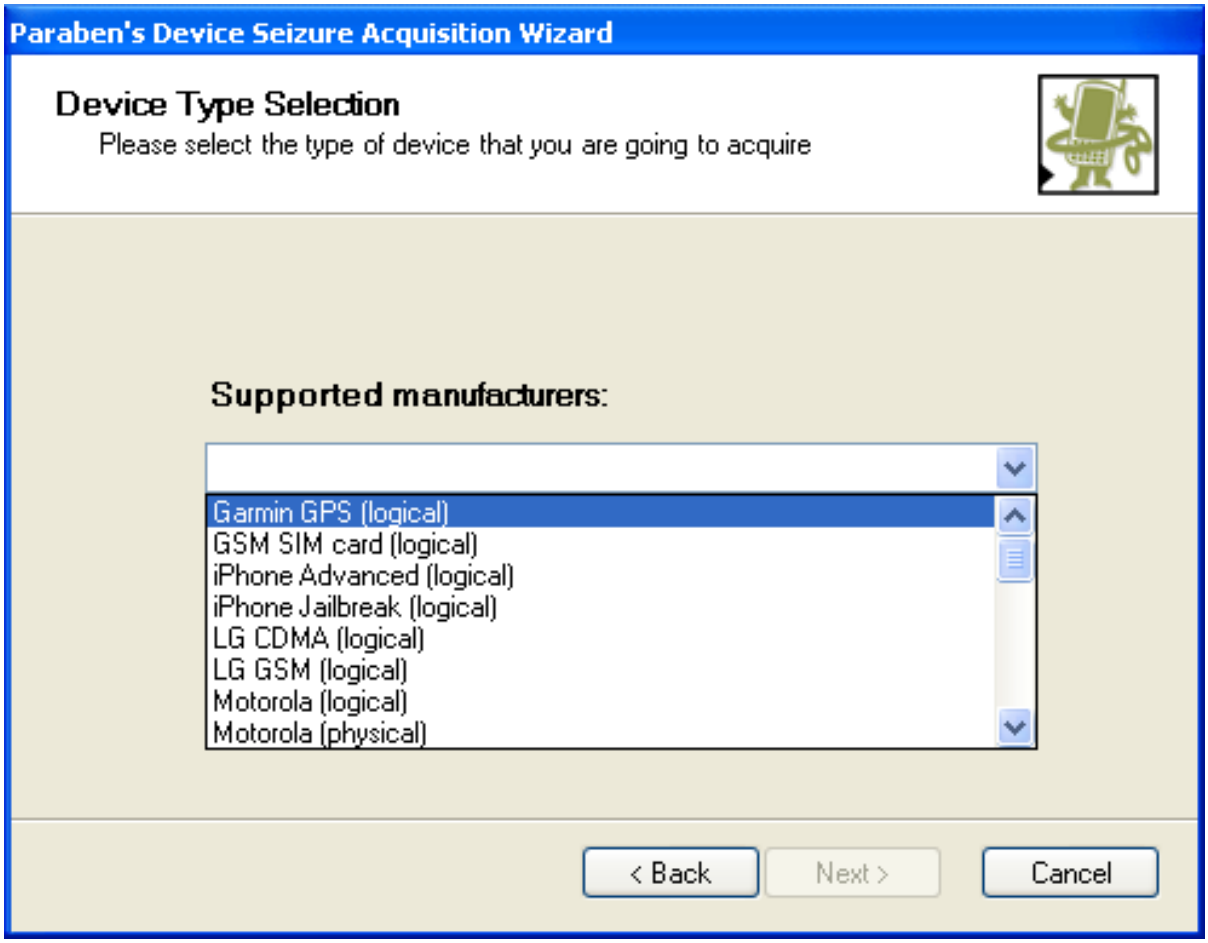


And then run the Acquisition wizard (note, you can also Import from an iPhone backup with the Import Wizard however this failed in 2.2 and I focused on the acquisition in 3.0).

Figure 4.3. Paraben Device Seizure Wizard

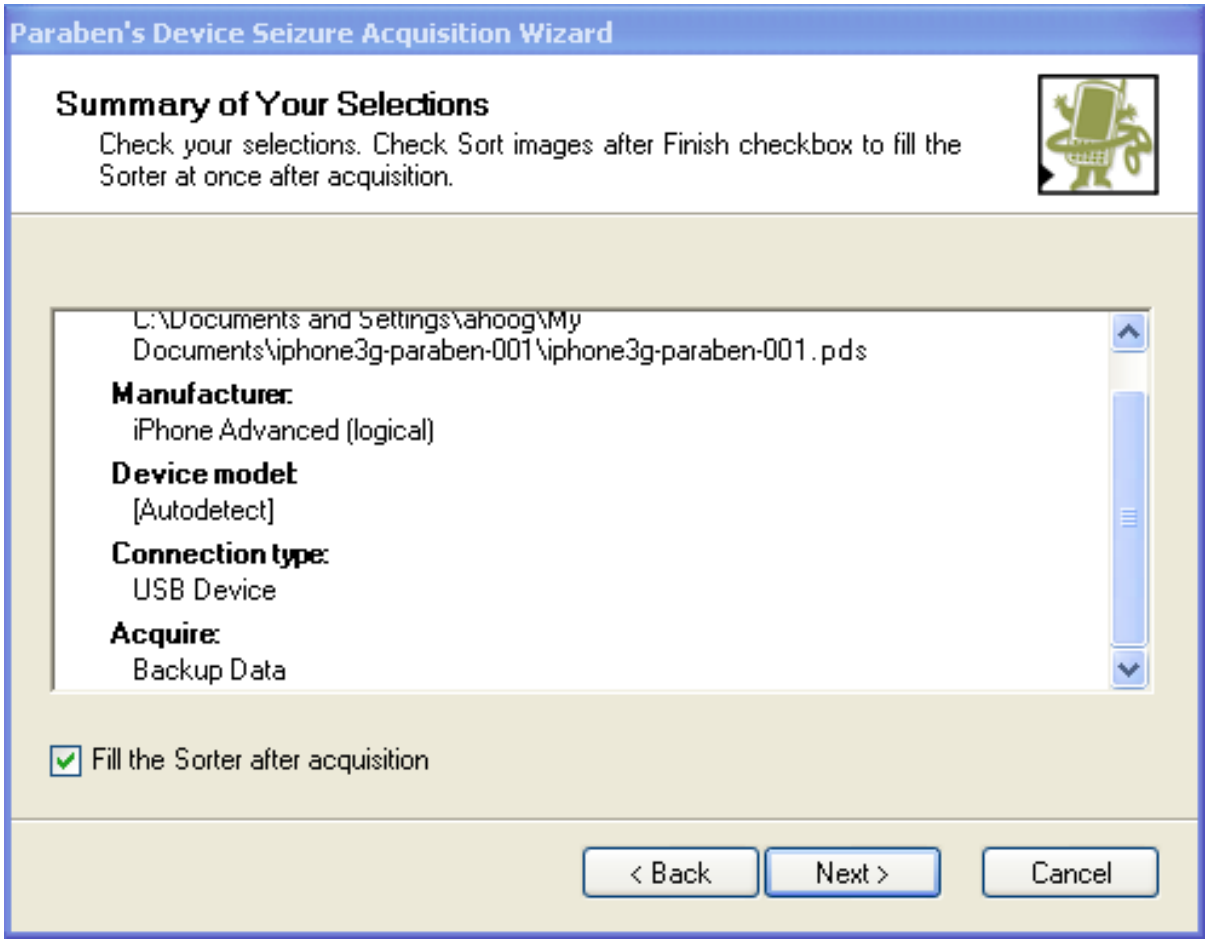
Next, you need to select how you want to acquire the iPhone. This is where the information from Paraben's Technical Support was very helpful. Paraben provides two methods for acquiring data from an iPhone and named them "iPhone Advanced (logical)" and "iPhone Jailbroken Devices Only (logical)". This was very confusing and when I read the Help, I decided to only perform the iPhone Advanced as the phone was not jailbroken. However, Support recommended running both acquisitions against an iPhone and this did yield good results.

Figure 4.4. Paraben Device Selection



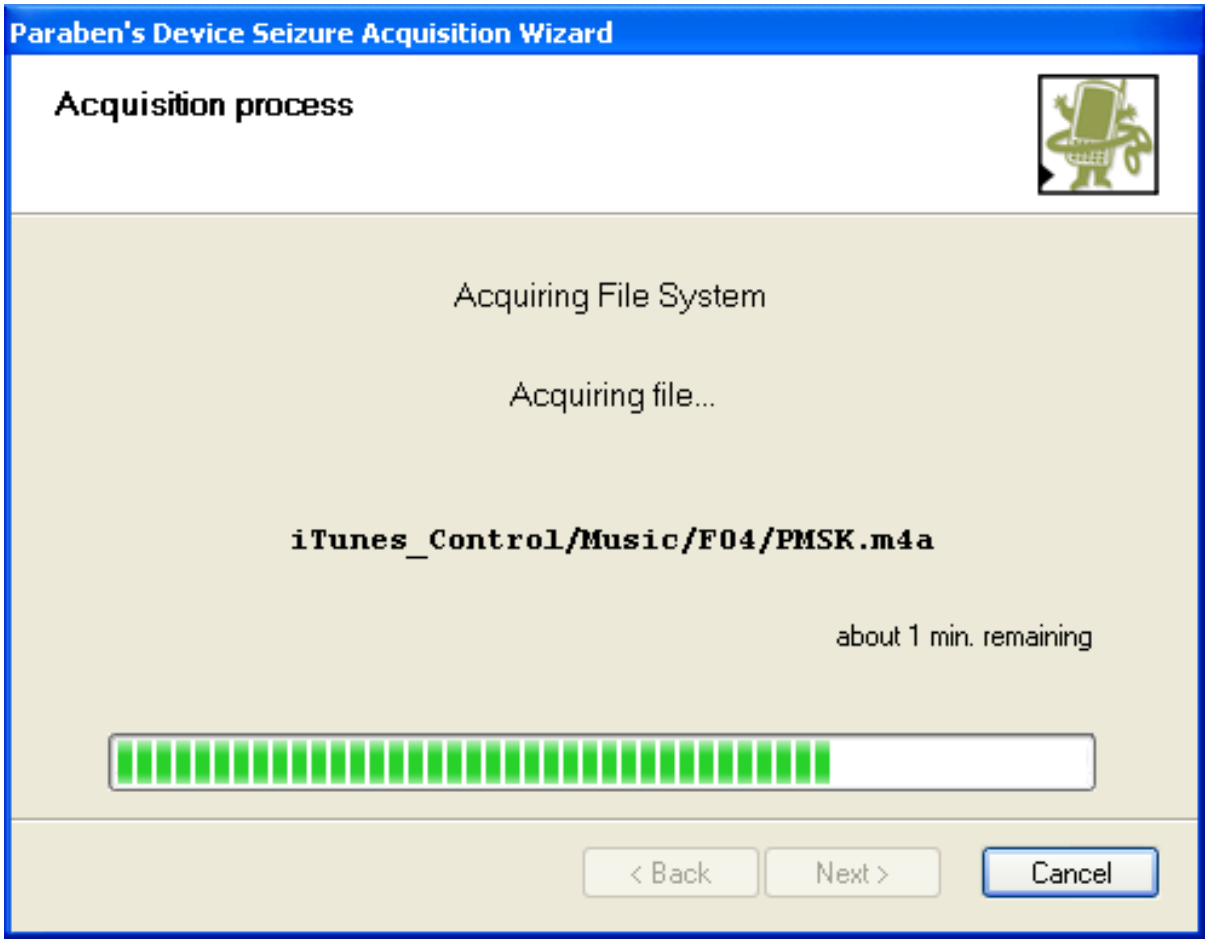
DS detects the device and you are ready to start.

Figure 4.5. Paraben Summary of Selections



Both the iPhone Advanced and iPhone Jailbroken Devices Only (called iPhone Jailbreak in 2.2) methods were quite fast lasting only a few minutes each.

Figure 4.6. Paraben Acquisition Process

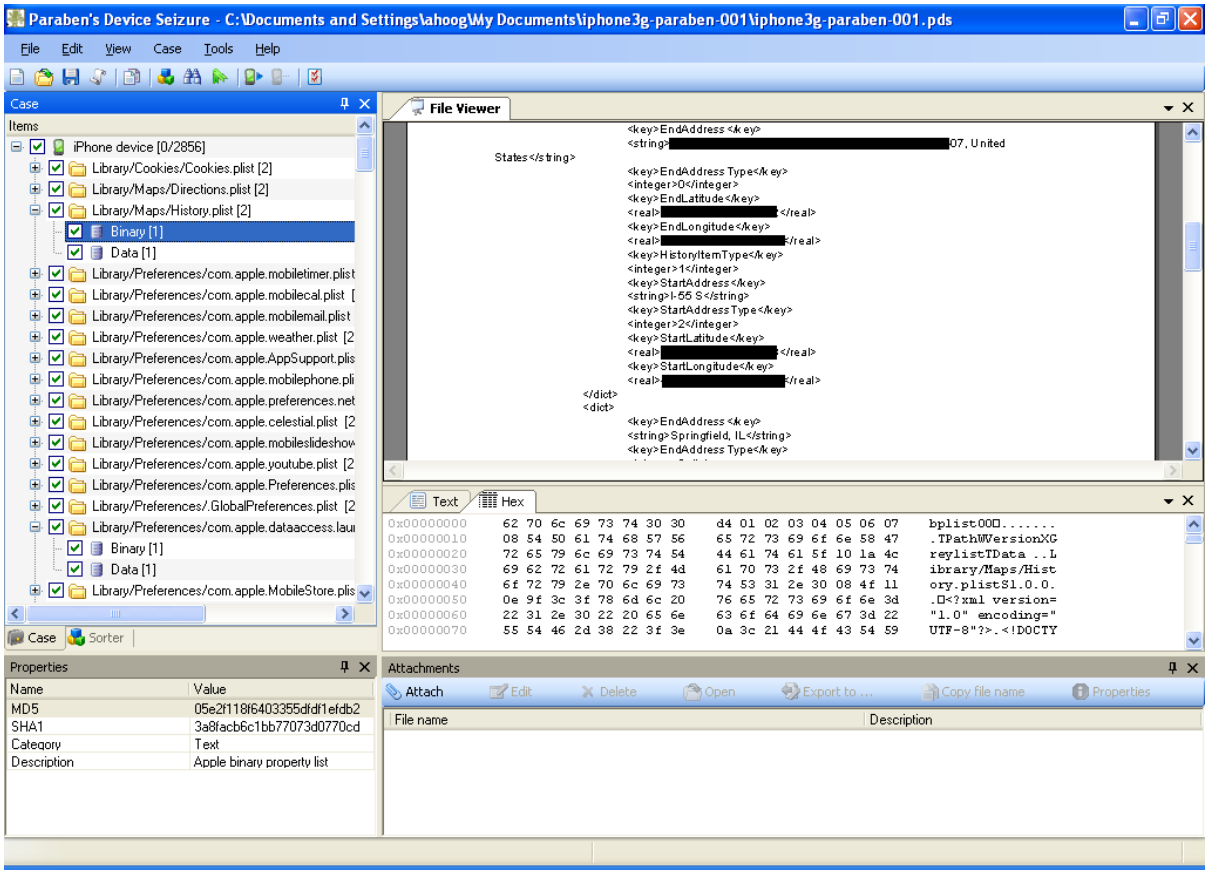


4. Results and Reporting

Paraben did a very good job extracting data from the iPhone using both the iPhone Advanced and iPhone Jailbroken Devices only plug-in (even though the iPhone was not Jailbroken). The Advanced plug-in extracted 2,856 items and the Jailbroken plug-in extracted 169. The Jailbroken recovered items such as the Music files which the Advanced plug-in was unable to extract.

When the acquisition is complete, DS presents the investigator with a user interface consisting of the case and acquired elements in a left pane and a window for the properties of the acquired data (MD5 and SHA1 hashes, Category and Description). There is a large pane for viewing the contents of a data elements and running the appropriate viewer. For instance, when viewing most SQLite database (although not all for some reason), the viewer windows display the data in a grid format. There are viewers for text, binary Plists, pictures, SQLite and more file types. Below the viewer, when appropriate, there are tabs to view the data in Text or hex.

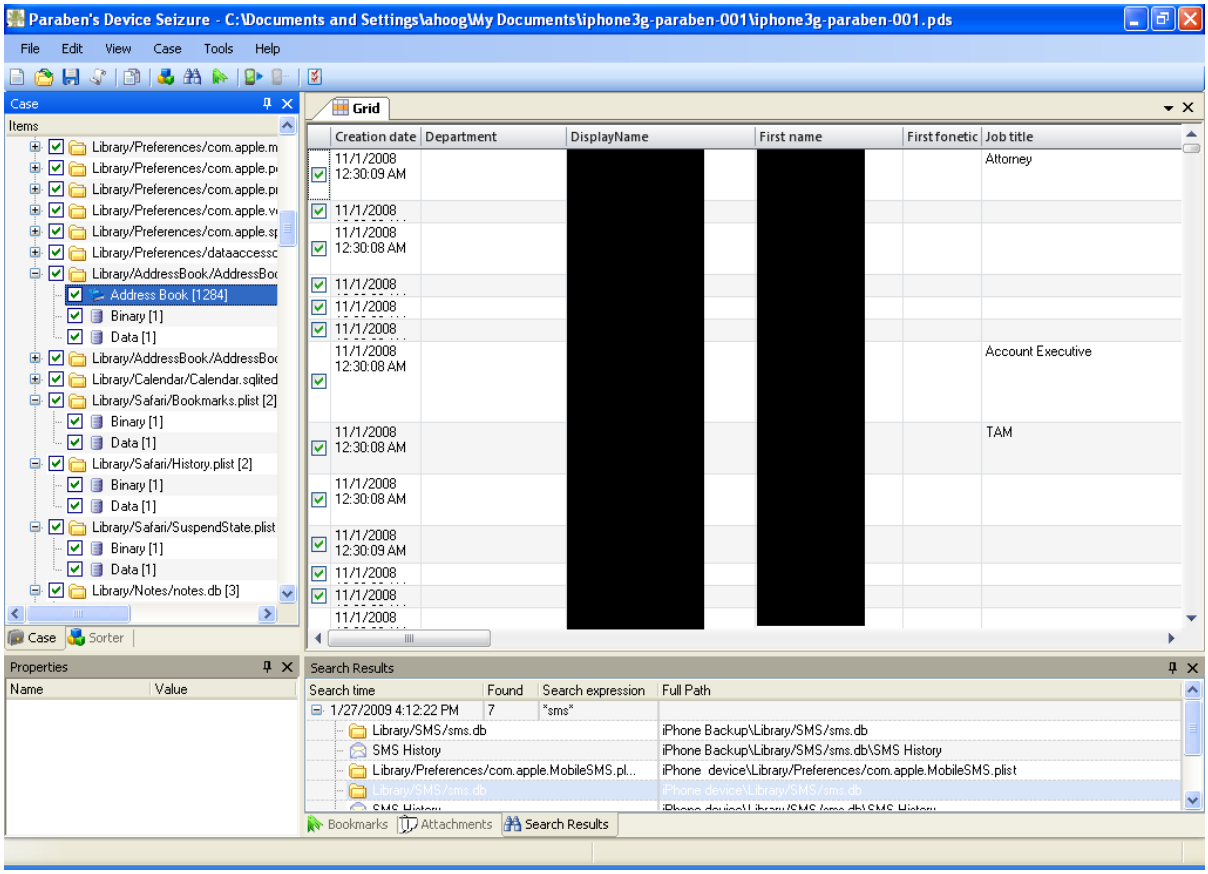
Figure 4.7. Paraben History.plist



In the example above, you can see by extracting and exposing the History.plist for Google Maps that the analyst can view the start address, end address and more information presented to the user during their interaction with the application.

The bottom window will also show search results, bookmarks and attachments. Below is an example of the data grid displaying the Contacts SQLite data.

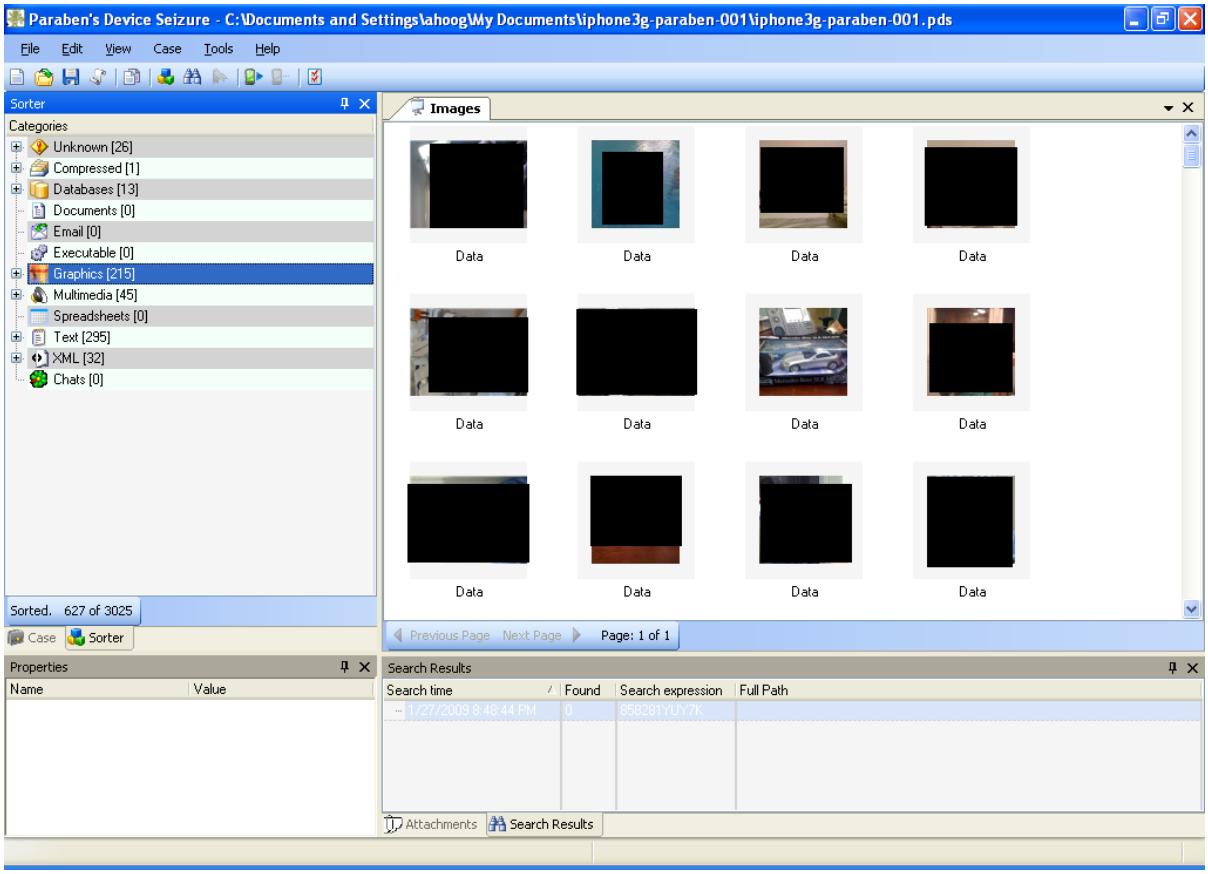
Figure 4.8. Paraben Address Book



If you want to export the SQLite database for further analysis, you can click on the Data node and select export. The file as well as a hash file are exported to a directory of your choosing and can be further analyzed as needed.

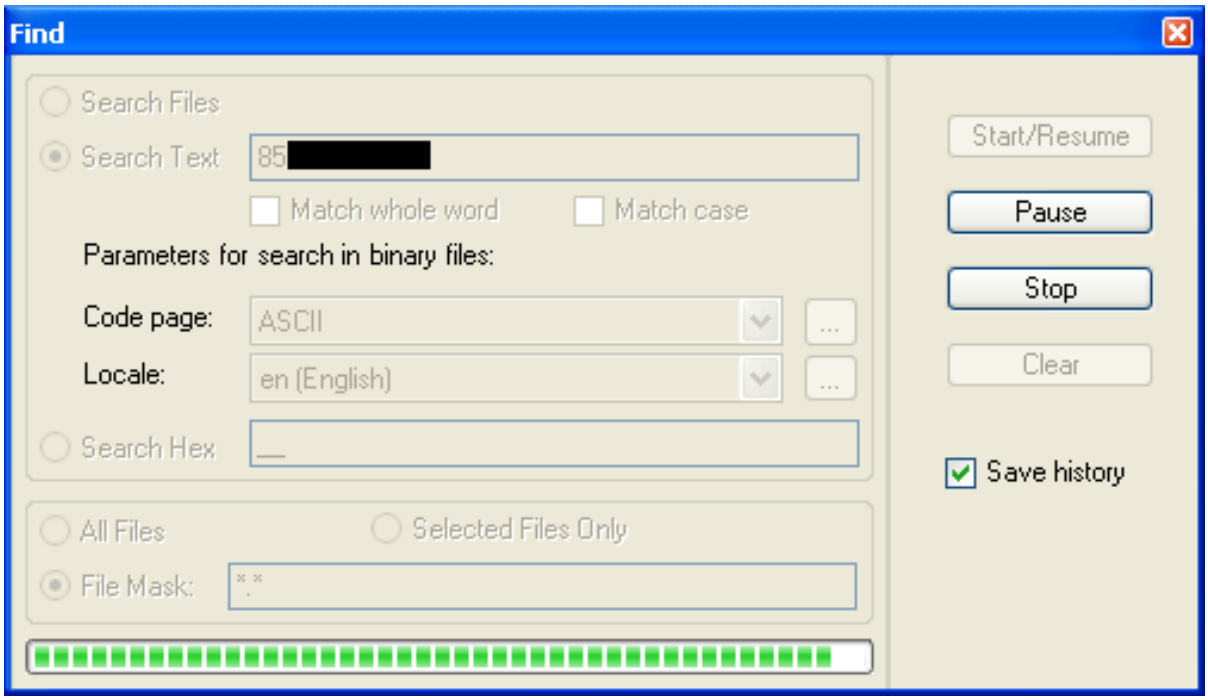
Another nice feature is the Sorter which will sort the acquired data by file type for quick review. The Sorter is a nice feature and worked fairly well but did miss some files that can be found by (tediously) going through the full list of acquired data in the case view.

Figure 4.9. Paraben Images



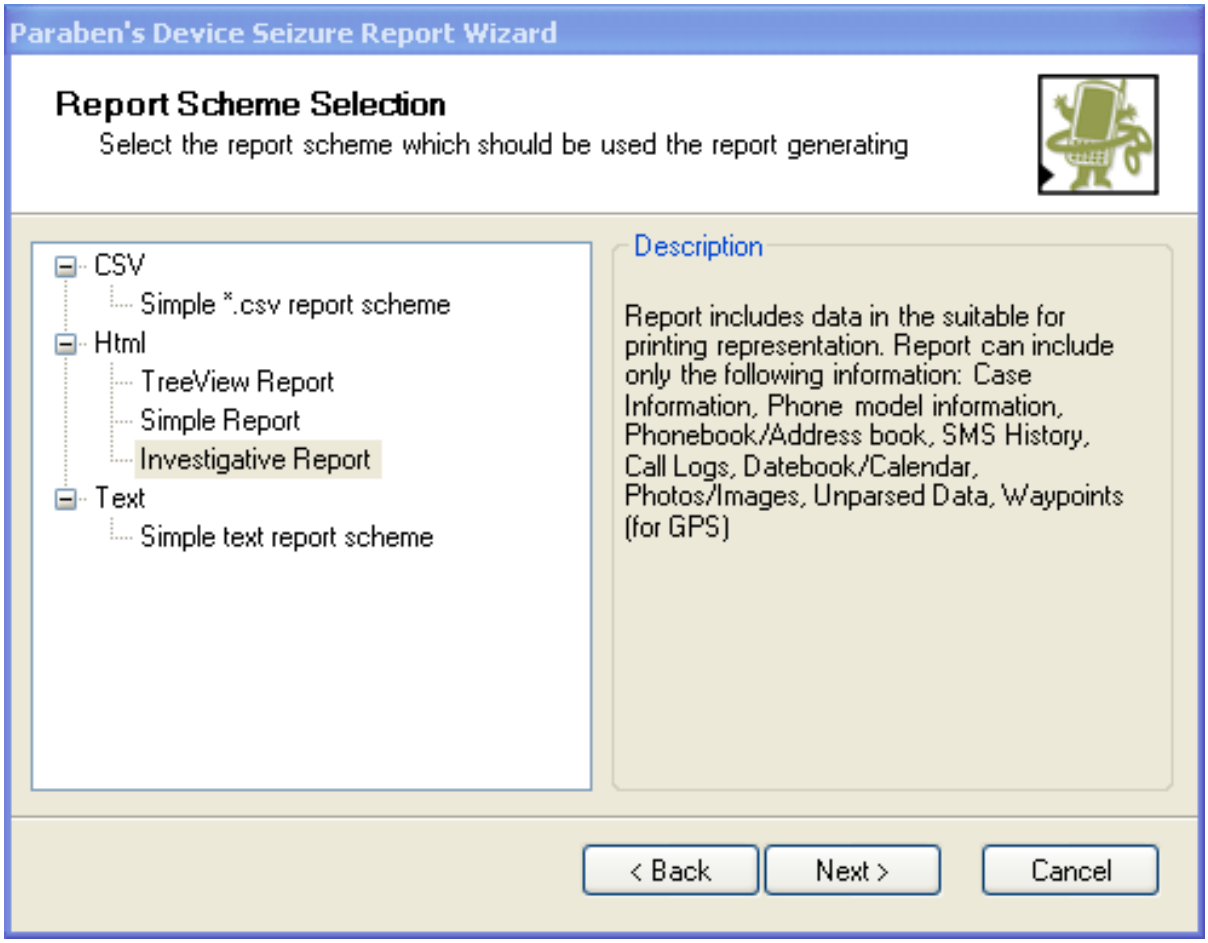
Searching is built in and fairly sophisticated including options to search text, hex, filename and Boolean variables. Searches were simple to execute and results were easy to examine.

Figure 4.10. Paraben Find



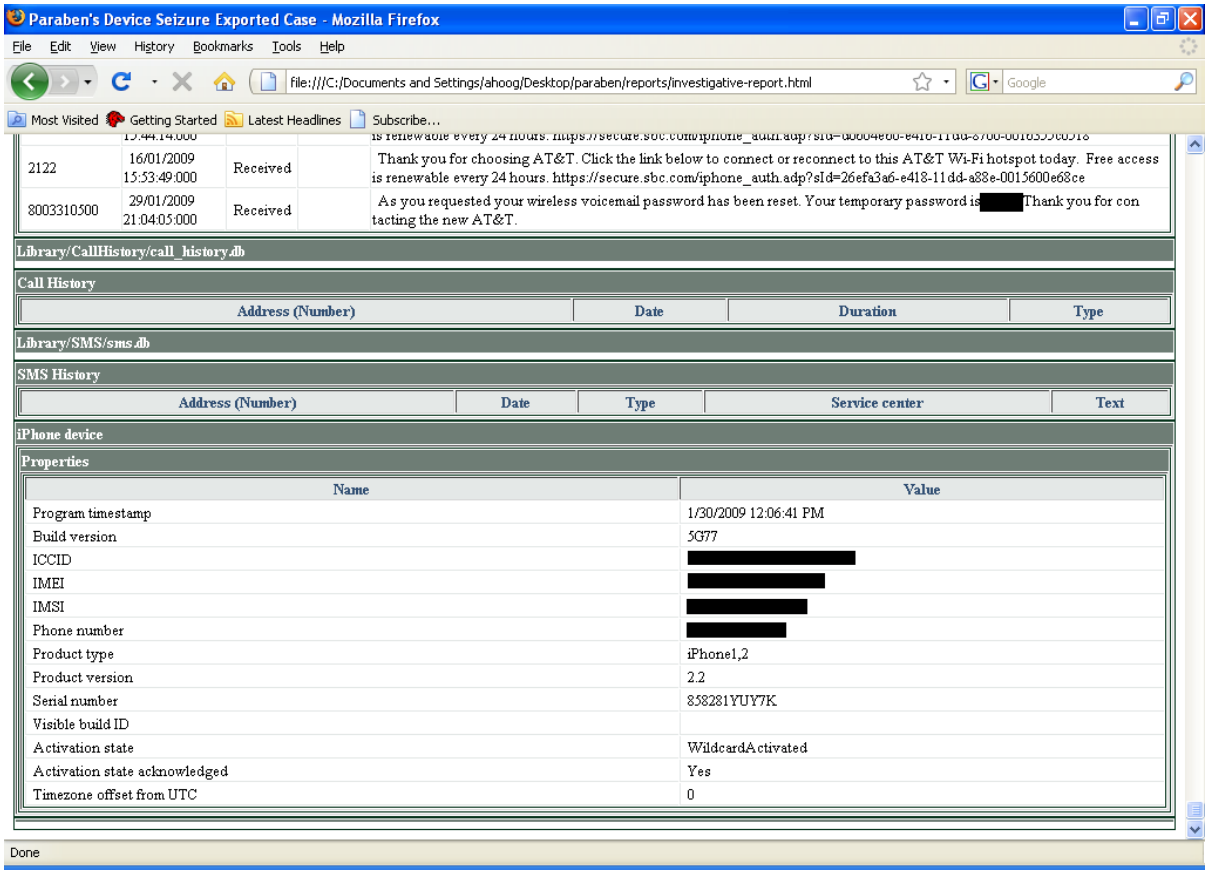
You have many reporting options, including various report types, custom selected files, bookmarks, etc.

Figure 4.11. Paraben Report Wizard



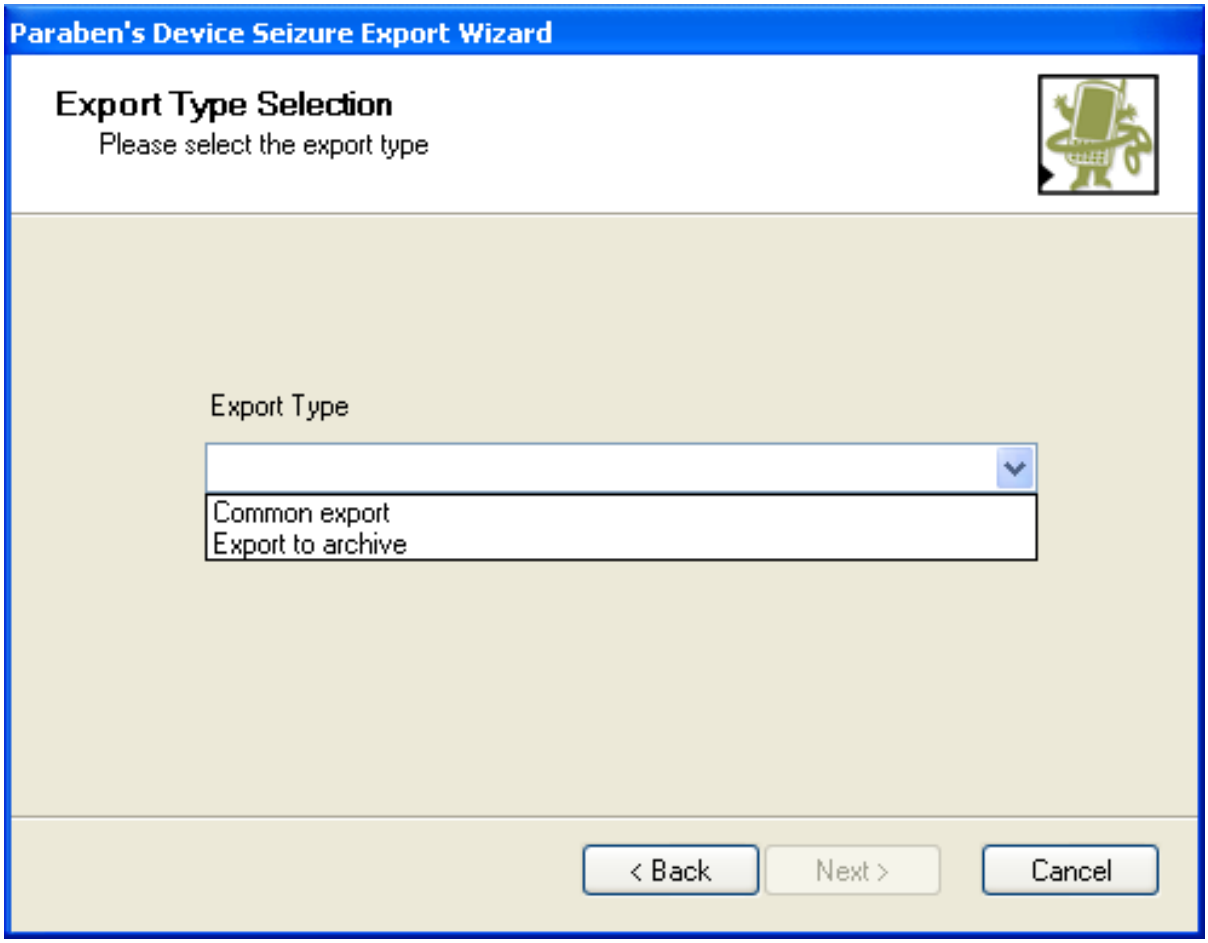
I chose the Investigative report and the following HTML report was generated.

Figure 4.12. Paraben Exported Case



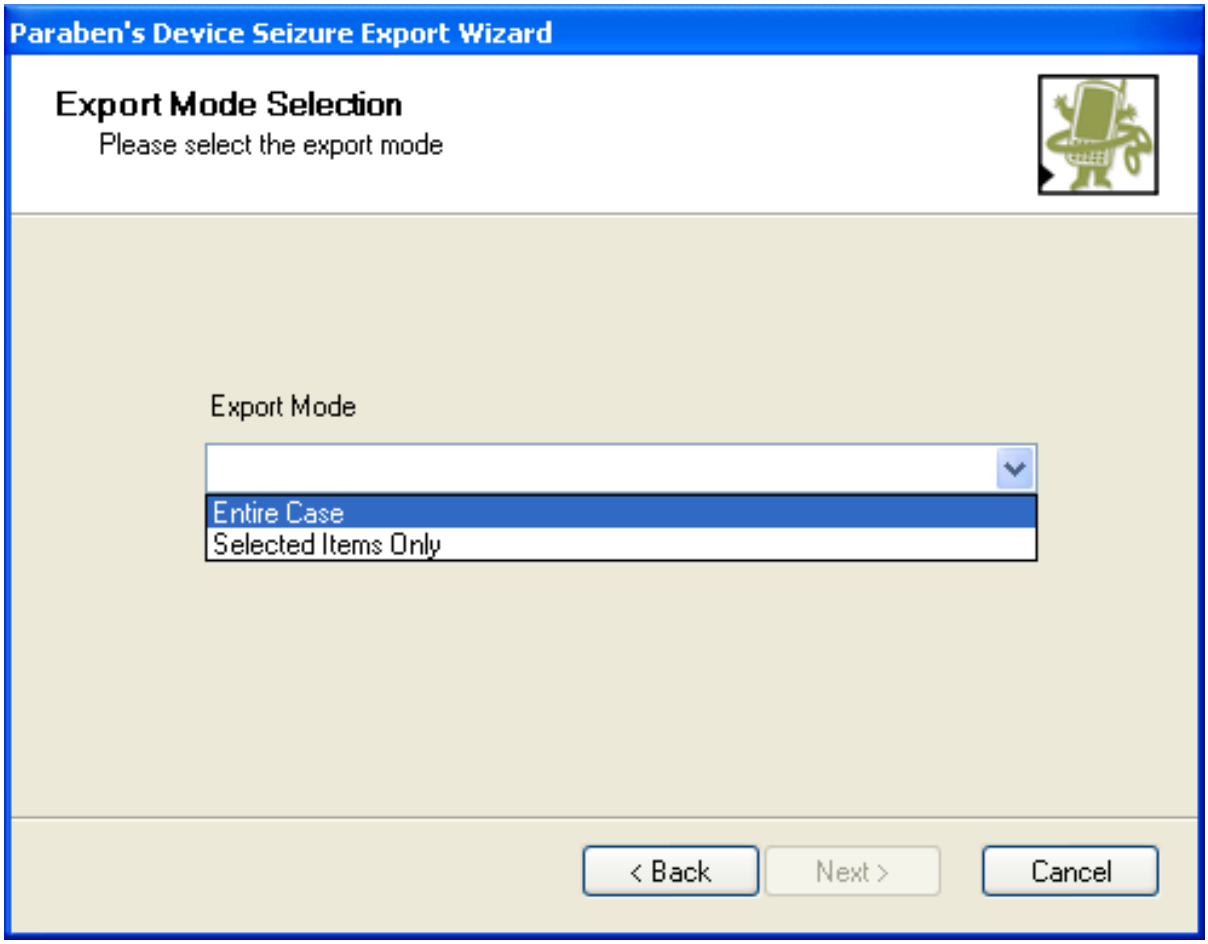
Finally, you can (and likely should) export the acquired data (or items you selected individually or bookmarked) to the file system. This will allow for direct examination of SQLite files and other forensic techniques. To export, I chose a “Common Export” which exported the files directly to the file system instead of within a compressed (ZIP) archive.

Figure 4.13. Paraben Export Type



You can then select the items you want to export (Entire Case or Selected Items Only).

Figure 4.14. Paraben Export Mode



The export was quick and did not exhibit any problems. The results include an export of both the Binary and Data elements in the item, a properties file which details the MD5, SHA1, Category and Description of the data and an XML hash file which also contains the MD5 and SHA1 hashes.

5. Matrix of Results

The following are the results from the Paraben tests.

Table 4.1. Paraben Matrix of Results

Scenario	Device Seizure	Ranking	Results
Call Logs	100	3	Meet
SMS	262	3	Meet
Contacts	1282	3	Meet
Email	some account info, folder info, etc.	1	Miss
Calendar	3070	3	Meet
Notes	2 (1 recovered in SQLite db)	5	Above
Pictures	178 (lots of icons, no synced pictures)	4	Above
Songs	44	3	Meet

Web History	2	3	Meet
Bookmarks	5	3	Meet
Cookies	29	5	Above
App Info	Yes	3	Meet
Google Maps	5 histories	3	Meet
Voicemail	0	0	Miss
Password	7	3	Meet
Plists/XML	73	2	Below
Phone Info	Yes	3	Meet
Video	1	3	Meet
Podcasts	1	3	Meet
Speed Dials	4	3	Meet
VPN	Yes	3	Meet
Bluetooth	Yes	3	Meet
GPS	Yes	3	Meet
File Hashes	Yes	3	Meet
You Tube	50 URLs	4	Above
HTML	0	0	Miss
Office Docs	0	0	Miss

6. Conclusions

Device Seizure 3.0 is a significant improvement over version 2.2 for acquisition of the Apple iPhone. After installation is complete, the acquisition and reporting processes are fast and thorough. Aside from a physical (dd) acquisition, Paraben returned more data from the acquisition stage than any other product. The user interface for subsequent analysis is also quite mature and provides many features other tools lack.

The following ranking establishes Device Seizure's overall rating of 2.9 on the four criteria established at the beginning of this white paper.

Table 4.2. Paraben Rankings

Area	Weight	Rank
Installation	0.1	2.0
Acquisition	0.2	4.0
Reporting	0.3	3.0
Accuracy	0.4	2.8
TOTAL		2.9

Chapter 5. MacLock Pick (1.4/5.0)

1. Summary (from company information)

MacLockPick II (MLP) by SubRosaSoft (makers of MacForensicLab) takes a unique approach to forensic acquisition. The goal of MLP is to provide a cross platform forensic solution that performs a live acquisition of a suspect machine after inserting the USB device. The information is stored on the USB device and software is provided to analyze the results. The solution support plug-ins for many acquisition types however only the iPhone plug-ins were tested.

MLP does not work directly on the iPhone and instead targets the backup directory where the iPhone stored most files as MDBACKUP files. The following data is recoverable:

- Call History (Received, Dialed, Missed)
- Text Messages (SMS)
- Phonebook
- Notes
- Photos
- Mail Accounts setup for synchronization.
- International Roaming Edge Status
- Favorites - Speed dial entries
- Safari – State, History and Bookmarks
- Phone Details (IMEI / ESN, phone number, TMSI, IMSI)

2. Installation

After familiarizing myself with MLP’s approach to forensic acquisition, the setup was quite simple. Initially, you must license the device by running a program to generate a key file and sending to SubRosaSoft. They will then send you an .inf file which must be placed on the root of the USB key. I had to follow this process at a later time (after I updated the MLP executables) but the device initially arrived licensed.

After you insert the device on a computer, you can explore the drive and run various programs. To configure the acquisition process, I ran MacLockPick Setup (OS X) on my Mac 10.5.6 computer. The program read the device configuration and allowed me to select what type of data I wanted to acquire from the target device. I chose Apple Mobile and Apple Mobile Pictures.

Figure 5.1. MacLockPick Setup

After quitting this application (which saved the settings but that was not initially from the interface) the device was ready to acquire data from target computers.

I did receive a few application updates from SubRosaSoft during the testing and, as mentioned above, I eventually had to complete the licensing process by running MacLockPick Authenticator and following the steps. This process was intuitive. However, afterwards I had trouble running the updated software on the original Windows XP target and eventually switched to a different computer. I also had several problems updating the software and finally moved the entire contents of the device to an “old” folder and extracted the contents of the updated .zip file from a Mac instead of a Window XP computer. This resolved some of the issues and eventually I was able to proceed running the updated software.

3. Forensic Acquisition

With the device licensed and software updated, the acquisition process was quite simple. After inserting the device into the target Windows XP computer and running the Autoplay, my entire screen was taken over the MLP.

Figure 5.2. MacLockPick Acquisition

The acquisition only took a few minutes and I removed the drive and placed it into an analysis computer (Mac) for review the results.

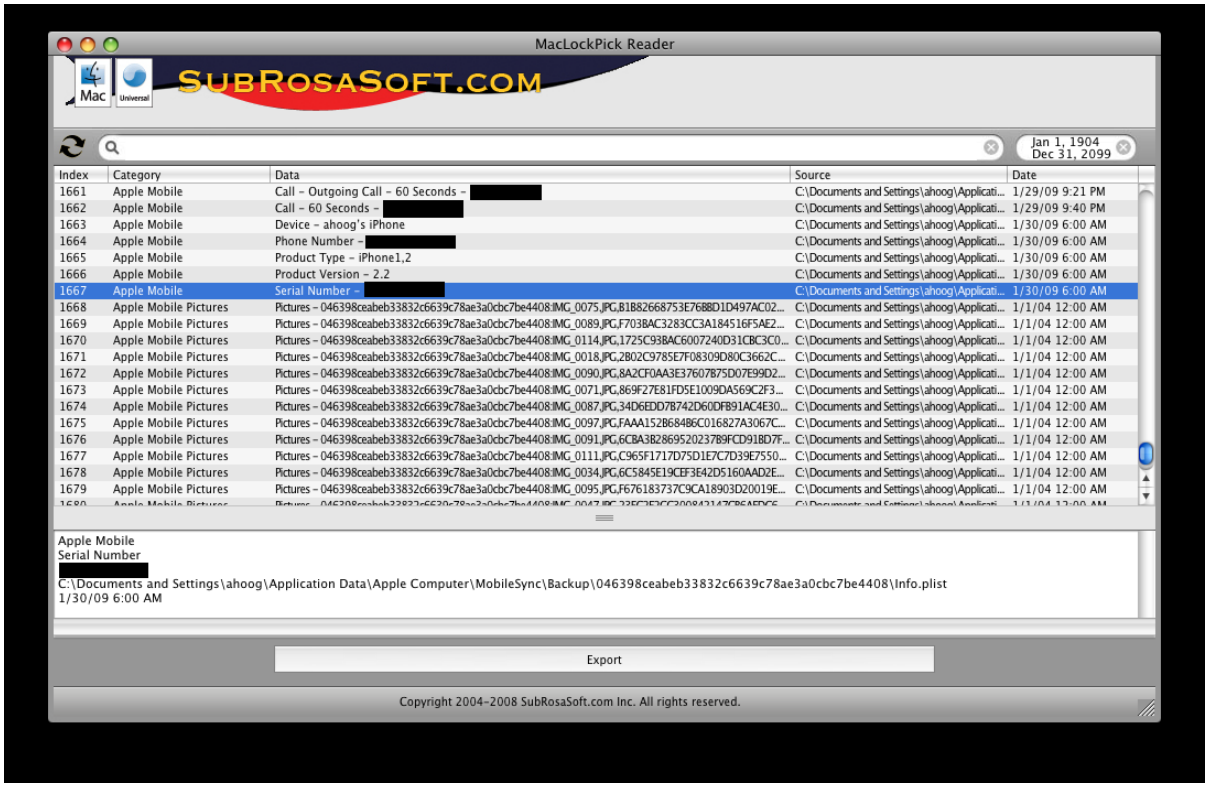
4. Results and Reporting

I ran the MacLockPick Reader on my Mac where I was prompted to open a keylog file.

Figure 5.3. MacLockPick Reader

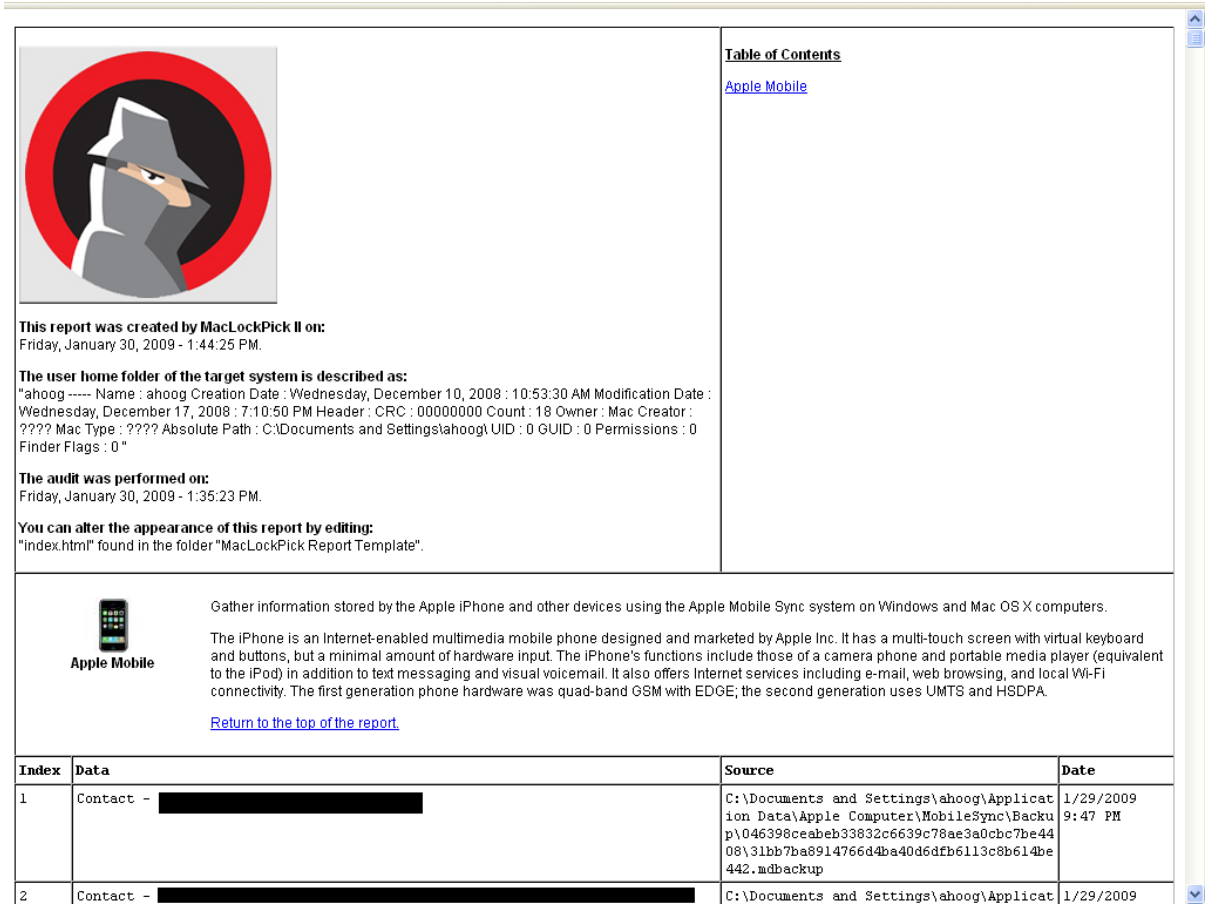
After selecting the file from the MLP device output directory, I was presented with the main MLP window which allowed me to examine the results, search and export/report on the data.

Figure 5.4. MacLockPick User Interface



The user interface is simple and allows for quick searches but I found that I wanted access to more information such as the raw files recovered. The reports generated quickly but were basically text file representations of the analysis window. There was an option to export to HTML which I could tell would contain much more data if a full computer acquisition was performed.

Figure 5.5. MacLockPick Table of Contents



This report was created by MacLockPick II on:
Friday, January 30, 2009 - 1:44:25 PM.

The user home folder of the target system is described as:
"ahoog ----- Name : ahoog Creation Date : Wednesday, December 10, 2008 : 10:53:30 AM Modification Date : Wednesday, December 17, 2008 : 7:10:50 PM Header : CRC : 00000000 Count : 18 Owner : Mac Creator : ???? Mac Type : ???? Absolute Path : C:\Documents and Settings\ahoog\ UID : 0 GUID : 0 Permissions : 0 Finder Flags : 0 "

The audit was performed on:
Friday, January 30, 2009 - 1:35:23 PM.

You can alter the appearance of this report by editing:
"index.html" found in the folder "MacLockPick Report Template".

Table of Contents

Apple Mobile

Apple Mobile

Gather information stored by the Apple iPhone and other devices using the Apple Mobile Sync system on Windows and Mac OS X computers.

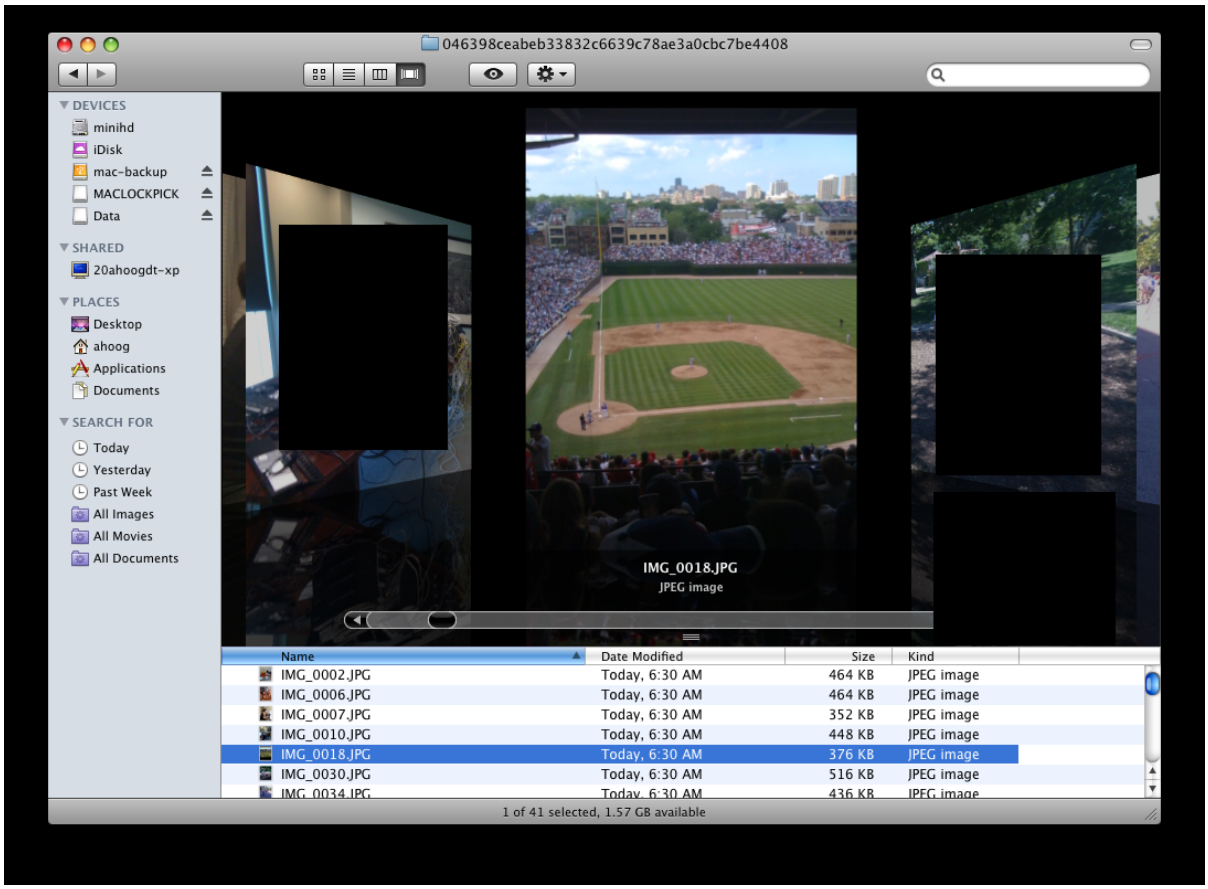
The iPhone is an Internet-enabled multimedia mobile phone designed and marketed by Apple Inc. It has a multi-touch screen with virtual keyboard and buttons, but a minimal amount of hardware input. The iPhone's functions include those of a camera phone and portable media player (equivalent to the iPod) in addition to text messaging and visual voicemail. It also offers Internet services including e-mail, web browsing, and local Wi-Fi connectivity. The first generation phone hardware was quad-band GSM with EDGE; the second generation uses UMTS and HSDPA.

Return to the top of the report.

Index	Data	Source	Date
1	Contact - [redacted]	C:\Documents and Settings\ahoog\Application Data\Apple Computer\MobileSync\Backup\046398ceabeb33832c6639c78ae3a0cbc7be4408\31bb7ba8914766d4ba40d6dfb6113c8b614be442.mdbackup	1/29/2009 9:47 PM
2	Contact - [redacted]	C:\Documents and Settings\ahoog\Applicat	1/29/2009

Finally, the latest version extracted the pictures found in the backup directory and placed them in a subdirectory for direct review. Below is a screenshot of the folder using Apple's Cover Flow view in Finder.

Figure 5.6. MacLockPick Picture Viewer



5. Matrix of Results

The following are the results from the MacLockPick tests.

Table 5.1. MacLockPick Matrix of Results

Scenario	MacLockPick Results	Ranking	Results
Call Logs	100	3	Meet
SMS	262	3	Meet
Contacts	1282	3	Meet
Email	0	0	Miss
Calendar	0	0	Miss
Notes	1	3	Meet
Pictures	41	3	Meet
Songs	0	0	Miss
Web History	0	0	Miss
Bookmarks	0	0	Miss
Cookies	0	0	Miss
App Info	0	0	Miss
Google Maps	0	0	Miss
Voicemail	0	0	Miss

Password	7	3	Meet
Plists/XML	0	0	Miss
Phone Info	Yes	3	Meet
Video	0	0	Miss
Podcasts	0	0	Miss
Speed Dials	0	0	Miss
VPN	Some	2	Below
Bluetooth	Some	2	Below
GPS	0	0	Miss
File Hashes	From backup files	1	Below
You Tube	0	0	Miss
HTML	0	0	Miss
Office Docs	0	0	Miss

6. Conclusions

MacLockPick is a “triage” forensic tool targeted at first responders on-site at the scene of a crime or incident. It performs a fast, efficient acquisition of the target computer and is packaged as an easy to transport USB key device. However, without support for direct acquisition of the iPhone, much of the data is missing. The reporting interface is also designed for a fast analysis of the data and as such does not contain the sophistication found in other tools.

The following ranking establishes MacLockPick’s overall rating of 1.4 on the four criteria established at the beginning of this white paper.

Table 5.2. MacLockPick Rankings

Area	Weight	Rank
Installation	0.1	3.0
Acquisition	0.2	3.0
Reporting	0.3	1.0
Accuracy	0.4	1.0
TOTAL		1.4

Chapter 6. MDBackup Extract (2.2/5.0)

1. Summary (from company information)

MDBackup Extract is a Mac-only forensic tool from BlackBag Technologies (makers of Macintosh Forensic Suite and MacQuisition Boot Disk) that analyzes data from the iTunes mobile sync backup directory. The tool is currently in Beta and production information is limited. Since this is a Mac-only utility, you must copy the backup directory from a Windows computer to a Mac for analysis.

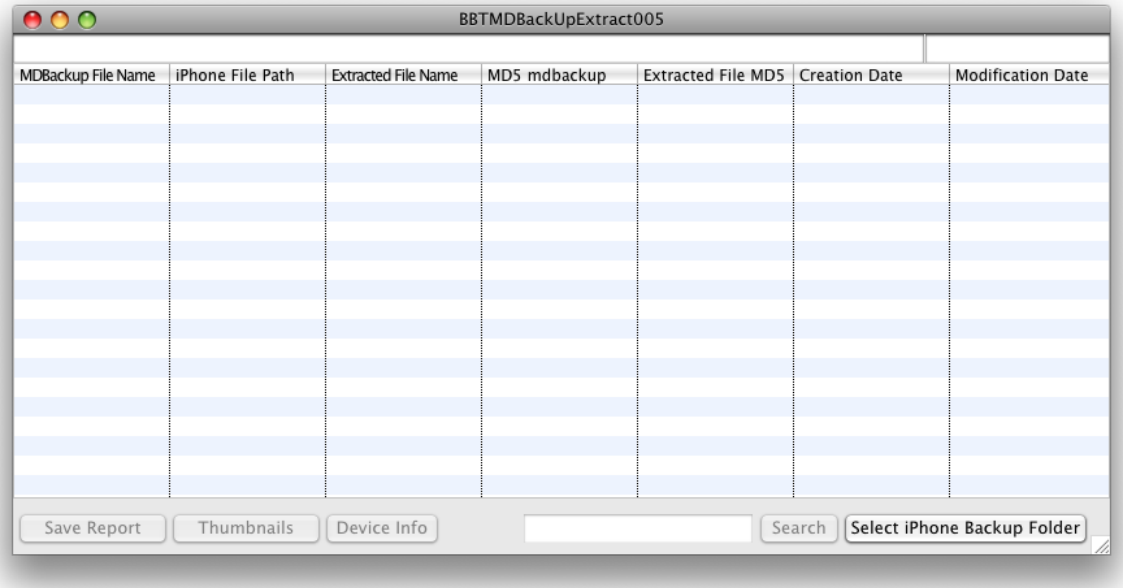
2. Installation

The program was delivered to me via email and by simply downloading it, Mac OS X recognized it as an application, confirmed I wished to run an downloaded from the Internet and then was up and running. Presumably the first release of this product will have some activation component.

3. Forensic Acquisition

When the application starts, you must select the iPhone backup folder from your computer.

Figure 6.1. MDBackup No Results



The first time I selected the backup folder, I miss-clicked and did not select the correct folder. The application accepted this folder but did not produce any results. After realizing my mistake, I selected the full backup folder and then was prompted to select an Extraction folder for the results.

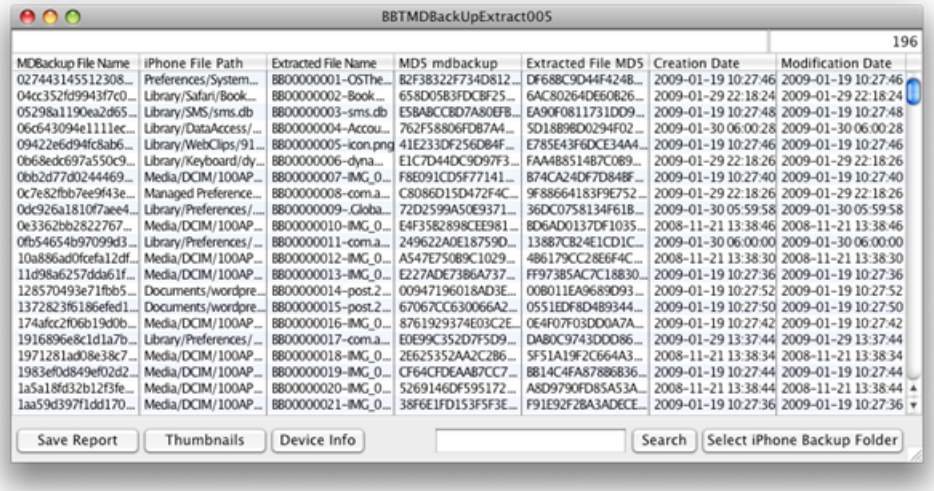
Figure 6.2. MDBackup Extraction Folder

Less than a minute later, the files were extracted and ready for analysis.

4. Results and Reporting

The main application window shows the results of the extraction and allows you to analyze the information.

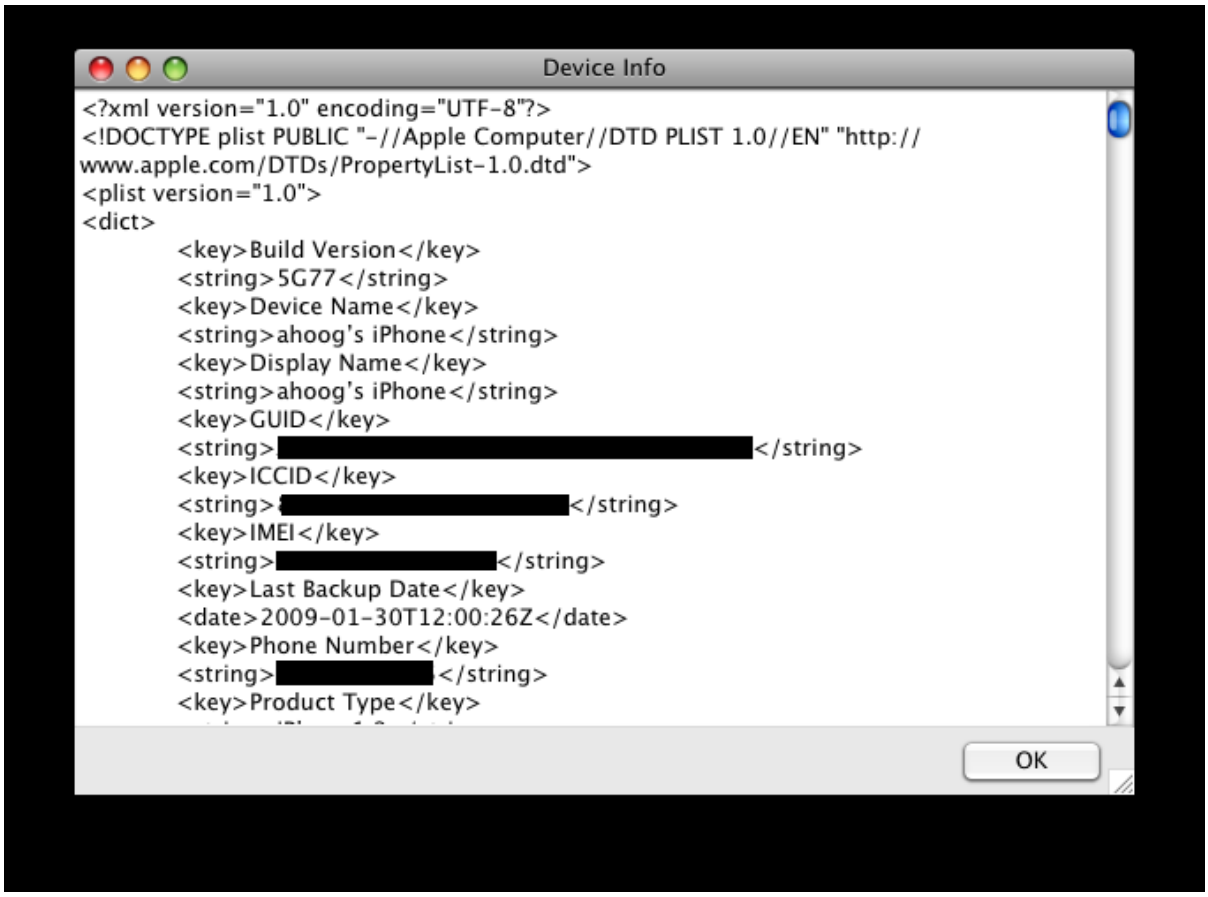
Figure 6.3. MDBackup Extraction Folder



If you look at the resulting folder on the Mac, you will find the application stores each extracted file in the Extraction directory and creates a subfolder called Original_Files which allow you to open/analyze the extracted files and still retain an original copy.

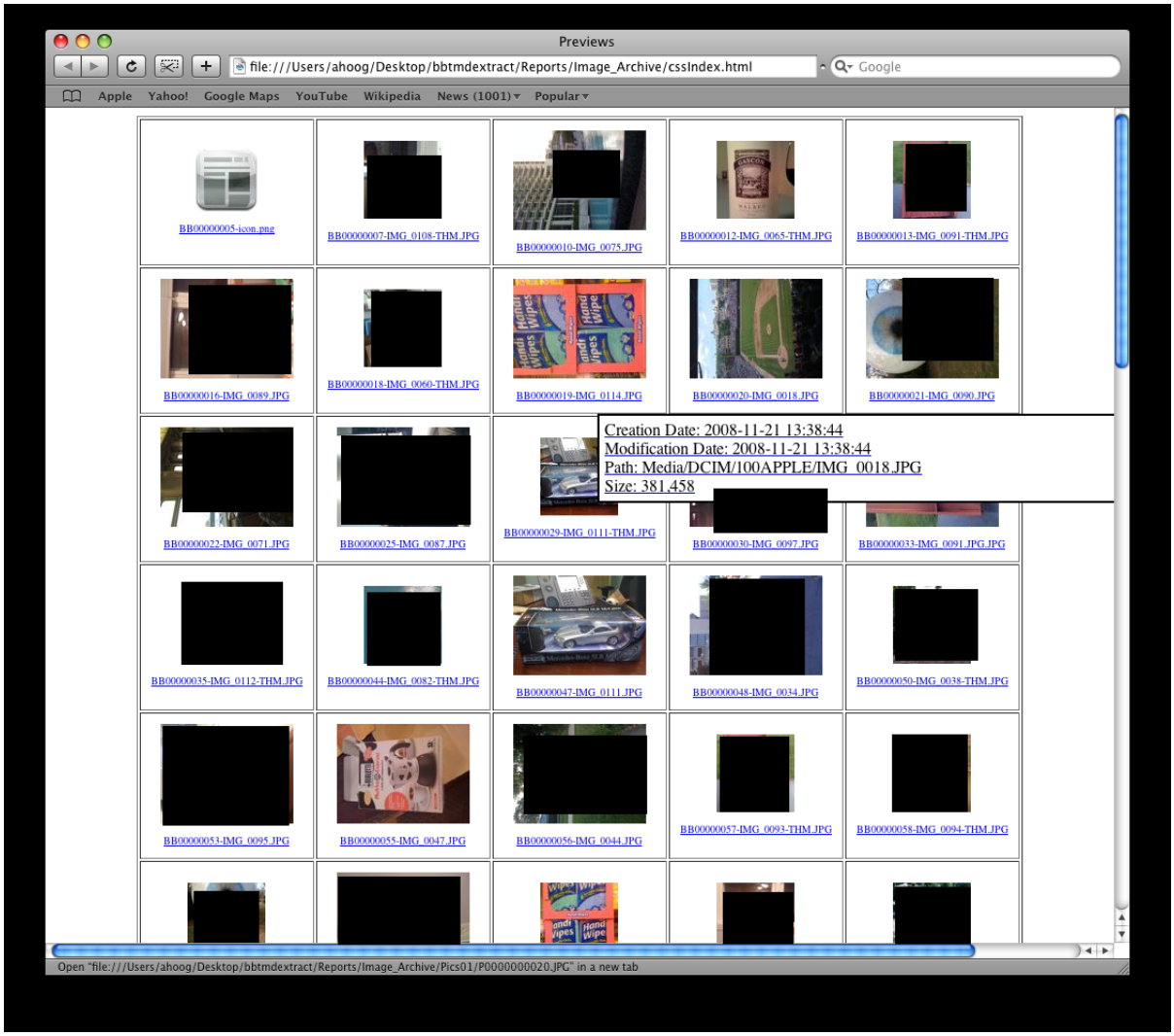
If you click on Device Info, the main Info.plist file with the core phone information is presented.

Figure 6.4. MDBackup Device Info



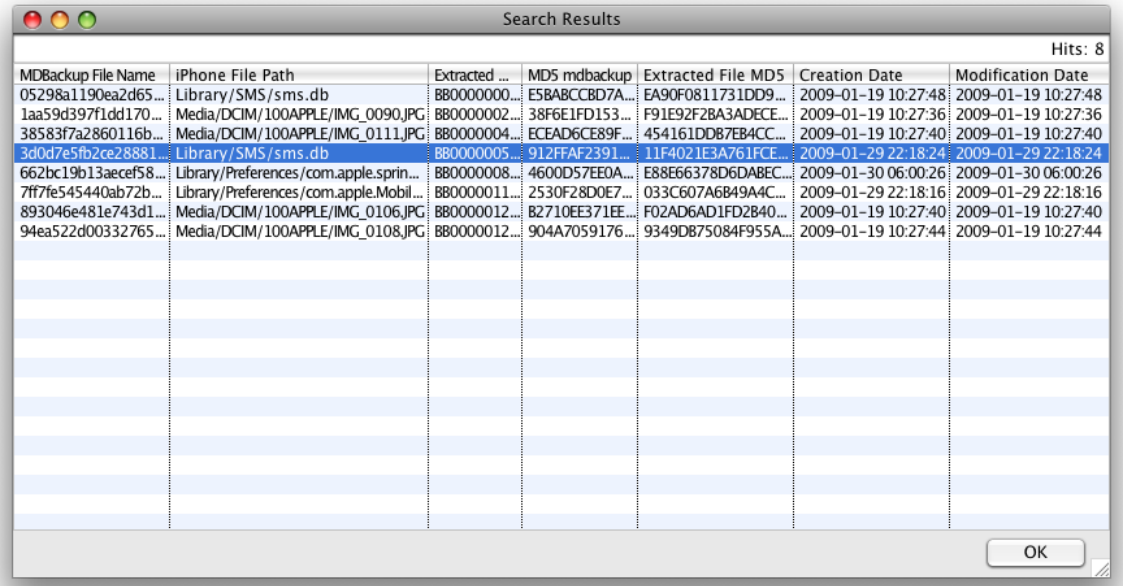
You can click on the Thumbnails button and the application will make a copy of all image files, create thumbnails and then write an HTML file (and open in Preview) which will allow you to quickly scan the images recovered. More information is provided when you hover over a picture and if you click on it, you can see the full picture.

Figure 6.5. MDBackup Pictures



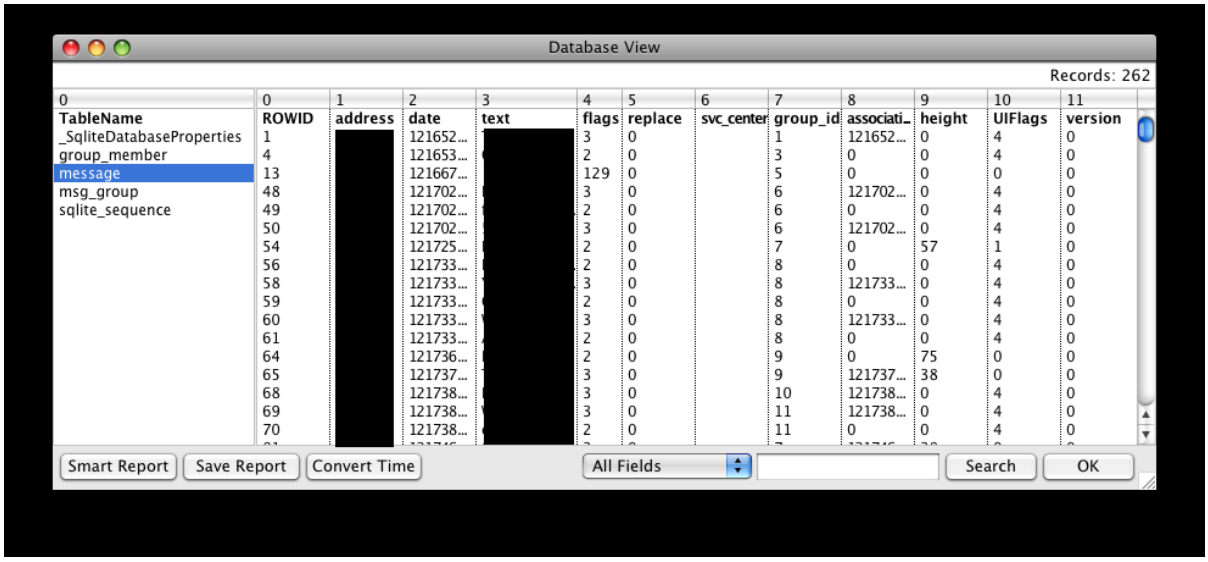
You can easily search for any keyword and click the Search button (unfortunately you cannot simply hit Return) and it pops up in a new window.

Figure 6.6. MDBackup Search Results



The application is file type aware and will run the appropriate viewer when you double-click on a row. I found the built-in SQLite viewer a very nice touch.

Figure 6.7. MDBackup SQLite Viewer



For certain data types (SMS, Call Log, Address Book, Address Book Images, Notes and Calendar), you can click on the Smart Report button and the application will save the important fields to a file and convert the time to GMT. While the formatting of the text report makes it a little difficult to follow in most text editors, this is still a nice feature.

Since the utility extracted the files to the file system, it allowed for a thorough analysis of the SQLite, Plist and other files resulting it a fair amount of information being extracted. However, all media files were missed (songs, video, podcasts, etc.). Nearly 200 files were extracted including over 150 Plist and XML configuration files.

There are some usability issues I came across. For instance, I did not find any way to open an existing acquisition which meant each time I wanted to verify something, I had to re-run the acquisition process. While this process was very fast it still was a bit cumbersome to run multiple times.

5. Matrix of Results

The following are the results from the MDBackupExtract tests.

Table 6.1. MDBackup Matrix of Results

Scenario	MDBackup Results	Ranking	Results
Call Logs	100	3	Meet
SMS	262	3	Meet
Contacts	1282 (14 w/images)	3	Meet
Email	0	0	Miss
Calendar	3070	3	Meet
Notes	1	3	Meet
Pictures	84	4	Above
Songs	0	0	Miss
Web History	2	3	Meet
Bookmarks	5	3	Meet
Cookies	29	5	Above
App Info	Yes	3	Meet
Google Maps	5 histories	3	Meet
Voicemail	0	0	Miss
Password	7	3	Meet
Plists/XML	108	3	Meet
Phone Info	Yes	3	Meet
Video	0	0	Miss
Podcasts	0	0	Miss
Speed Dials	4	3	Meet
VPN	Yes	3	Meet
Bluetooth	1	3	Meet
GPS	Yes	3	Meet
File Hashes	Some	2	Below
You Tube	50 URLs	4	Above
HTML	0	0	Miss
Office Docs	0	0	Miss

6. Conclusions

MDBackup Extract shows promise as an iPhone forensic tool for analyzing the backup directory. The native file and data viewers are fast and the search is effective. With some additional usability tweaks, the application could be a strong tool for consideration.

The following ranking establishes MDBackup Extract's overall rating of 2.2 on the four criteria established at the beginning of this white paper.

Table 6.2. MDBackup Extract Rankings

Area	Weight	Rank

Installation	0.1	3.0
Acquisition	0.2	3.0
Reporting	0.3	1.0
Accuracy	0.4	2.3
TOTAL		2.2

Chapter 7. Zdziarski Technique (3.3/5.0)

1. Summary (from company material)

Jonathan Zdziarski is a Research Scientist for McAfee, Inc., and well known outside of work in the iPhone community as "NerveGas", who has contributed significantly to research into the iPhone and iPod touch.

He has authored many utilities, and devised many of the methods used to open the iPhone's platform to the open source community. Zdziarski has written three books pertaining to the iPhone platform: iPhone Open Application Development, iPhone SDK Applications, and iPhone Forensics: Recovering Evidence, Personal Data, and Corporate Assets.

Prior to publishing iPhone Forensics, Zdziarski maintained an unofficial forensics guide for the iPhone distributed exclusively to law enforcement.

The “holy grail” of any forensic acquisition is performing a bit-by-bit copy of the original media and retaining proof that the two are identical by comparing the unique cryptographic signatures of the original and the copy to ensure they match. While the processes and procedures for this are well established in traditional hard drive based computer forensics, the process for mobile forensic devices proves quite challenging. Unlike hard drives which can be removed or computers that can start in special read-only forensic environments, cell phones comparatively have a very limited operating system, cannot boot into a forensic environment and the memory is typically non-removable. Logical copies have been the only method of performing mobile forensics outside of imaging removed memory cards or reading SIM card data.

The uniqueness of Zdziarski’s method is that it is the only approach that allows the examiner to perform a bit-by-bit copy of the iPhone’s user partition and can provide an MD5 sum to prove the copy was authentic. However, this ability does not exist in the standard iPhone operating system and requires modifying a read-only system partition to allow for this technique. Fortunately, this partition remains completely isolated from the partition containing user data and is intended to remain in a factory state throughout the life of the iPhone. This makes it an ideal and forensically sound location to perform the necessary payload installation, without violating user data.

Zdziarski recognized the technique has obstacles to overcome to be fully understood and accepted and addressed various concerns beginning on the second page of his book. Zdziarski explains that since his technique modifies the system partition only and preserves the user partition, the process is not only valid but more reliable and complete than other "triage" approaches, because it provides access to the raw disk images and allows the examiner to bypass any security added onto the iPhone, such as a user pass code.

It is important to point out that during the normal operation of the iPhone, the system partition remains in a factory state and is only modified during a firmware upgrade. As such, the likelihood of any user data or evidence existing in this system partition in highly unlikely, so modifying this system partition is not much different than modifying a desktop's boot sequence to boot a USB keychain or CD-ROM.

Another important point to note is the difference between Zdziarski's methods and popular jailbreaking methods, and the forensic superiority of the former over the latter. The term jailbreaking refers to a hacking process by which the iPhone firmware is overwritten in order to install third party application bundles or perform baseband unlocking. The jailbreaking process makes many modifications to the user data partition (as well as the baseband radio) to accomplish this, making it forensically unsound. Zdziarski's procedures, on the other hand, are more custom-tailored to forensic recovery and only operate on the read-only system partition. Unlike jailbreaking, they do not install any additional software or modify the user data partition in any way.

Zdziarski's procedure involves creating a custom forensic recovery RAM disk that is booted as if it were a firmware restore. Rather than actually restoring the iPhone, this bundle installs a recovery payload onto the iPhone's read-only system partition granting the examiner SSH access to the device, and bypassing any pass code security that

might exist. The payload can also be optionally modified to disconnect the iPhone's user data from the system keychain, preventing the iPhone from logging into a suspect's email accounts after it has been seized.

In speaking with the author, he stated that the concerns attorneys, law enforcement officers, judges or juries might have are quickly alleviated when a simple explanation and overview of his technique is presented. He stressed the importance of explaining the difference between the jailbreaking hacking techniques and his own recovery techniques, and explained that some misconceptions about his procedures may have led some agencies to quickly discount his procedures based on incorrect assumptions.

2. Installation

Since Zdziarski's technique is not an application but rather the procedures to follow to install a forensic toolkit and acquire a bit-by-bit copy (dd image), there is no single application to install. Rather, you must possess the tools and technical knowledge to follow his procedures.

I consider myself very technical. I've taught assembly language as an Adjunct Professor, programmed numerous system and in many languages, lead large teams of developers on enterprise implementations, performed complicated Cisco networking and phone installs, function as the CIO of a large, multi-national corporation and I am an experience computer/mobile forensic analyst. I have been an avid user of Linux since 1993 (pre-1.x kernel) and am comfortable navigating in many operating systems including version of VMS, Unix, Linux, Mac and Windows.

I mention the above to provide some background as to my technical knowledge. I was confident at the outset that I could follow Zdziarski's technique and easily acquire an iPhone. However, my experience was quite different. I feel certain that without 15+ years of highly technical experience, I would have likely failed or would have certainly taken much longer to succeed (and perhaps sacrifice the iPhone data a few times along the way). In the end, I succeeded in performing Zdziarski's technique and acquired a physical image of the iPhone user data partition. For completeness, I performed the technique a second time on a different phone. The method certainly works and while very technical, the benefits are, are you will see, quite significant.

Briefly, the following four steps provide an overview of the procedures you must follow to prepare an iPhone for a physical dd acquisition (if you want the full details I suggest purchasing a copy of his book):

1. 1.Use Pwnage Tool to create a custom firmware package (which you then modify in steps 2 and 3) and prepare the boot ROM to accept the custom (unsigned) images.

2. 2.Use xpwntool (from Xpwn) to create a "Stage 1" custom firmware package that will updated the NOR (kernel cache) and not destroy the live user data.

3. 3.Use xpwntool (from Xpwn) to create a "Stage 2" custom firmware package that will install the forensic recovery toolkit used to create a dd image and MD5 signature of the user partition.

4. 4.Use iTunes and install the Stage 1 and then Stage 2 firmware, achieved by placing the iPhone in DFU (Device Firmware Update) mode.

I will preface these steps by stating that I used a combination of Mac OS X 10.4, 10.5, Debian Linux (2.6.18) and Windows XP Professional to perform various tasks. In his book, Zdziarski points out both the Mac and Windows XP versions of the packages needed however to perform his technique however he executes all steps using Mac OS X 10.5.

Installing Pwnage Tool and creating your base custom firmware package is quite simple. I chose to run Pwnage Tool 2.2.1 for the Mac. Basically, you follow the prompts. First, select your phone type (important to not get this part wrong).

Figure 7.1. Pwnage Tool

And then select the firmware (which you can download from several archive sites if needed) that the phone is running.

Figure 7.2. Pwnage Tool Selection

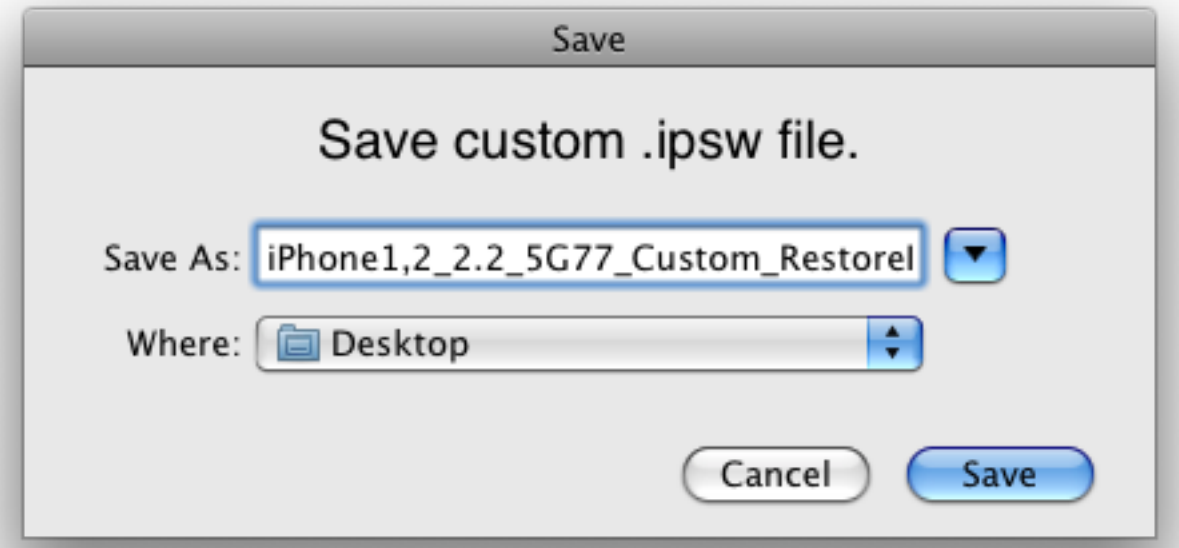
You then answer a series of question which I will not step you through in detail at this time.

Figure 7.3. Pwnage Tool General

But, basically you are using Pwnage Tool to build a custom firmware bundle (IPSW) which you will then modify to suit your needs.

After the questions are answered, Pwnage Tool will prompt you to name the custom firmware bundle.

Figure 7.4. Pwnage Tool Save

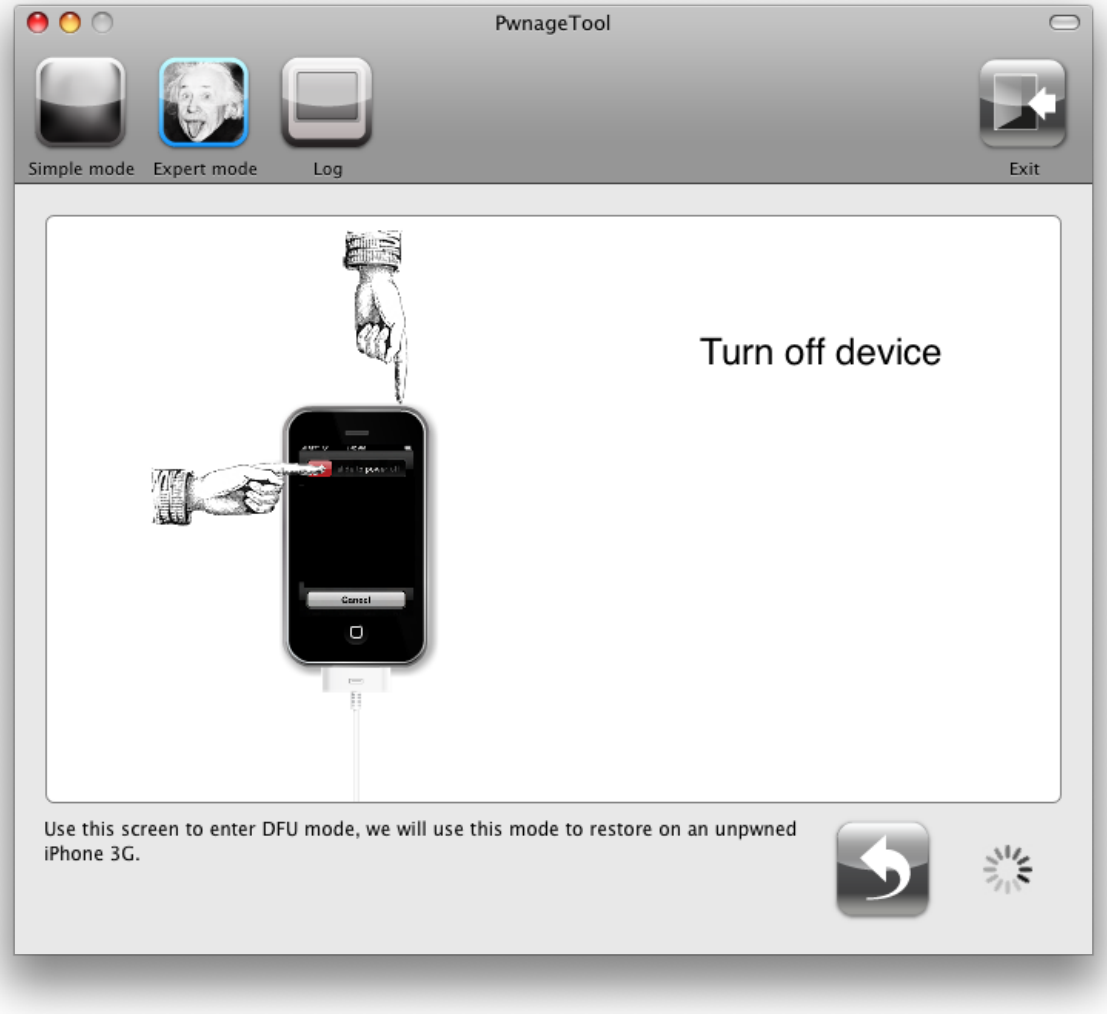


And will then build it for you.

Figure 7.5. Pwnage Tool Building

At this point, Pwnage Tool asks if your phone has ever been Pwned before. If not, if steps you through setting the iPhone into DFU mode.

Figure 7.6. Setting to DFU Mode



This prepares the iPhone to accept the custom firmware update into the boot ROM.

Figure 7.7. DFU Mode Successful

Things get a bit more interesting when you have to create the stage1.ipsw and stage2.ipsw firmware bundles. If you are following the book step by step, make sure you check the book's Errata (http://oreilly.com/catalog/9780596153588/) and the sample code section (http://www.zdziarski.com/iphone-forensics/) as there have been some corrections and updates since it was published.

When I first installed Xpwn on OS X, I noticed the firmware bundles needed to grab the key and initialization vector needed to unpack and decrypt the Apple firmware were not included. I did not want to switch to Windows XP so I decided to use Linux for these steps. The binaries I found for Xpwn on Linux did not work on my workstation as they required GLIBC 2.4. So, I downloaded and compiled the Xpwn on Linux and used Linux for steps 2 and 3 (full directions at http://chicago-ediscovery.com/iphone-forensic-howtos/howto-compile-xpwn-debian-etch.html).

While these steps were more technical, it is the final step that can be the most difficult and frustrating. Putting the iPhone in DFU mode is quite easy and you get better over time. But what is very frustrating are the series of errors you may experience from iTunes. You will eventually dread the various errors you may receive such an error 6, 1600, 1601, 1602 or 1604.

Figure 7.8. iTunes Errors

Here are a few things I did to (eventually) get around these errors.

1. Make sure the .ipsw image is not corrupt. While transferring the file between my Linux workstation and my Mac via a USB drive, I failed to complete a good copy of stage2.ispw and spent some time diagnosing the 6 and 1435 errors from iTunes. Those issues went away after re-transferring the firmware bundle.

2. There is much on Google about using the correct versions of iTunes with Pwnage Tool. To play it safe, I removed and added the version Zdziarski used to minimize errors. I used iTunes 7.7.8.43. Of course, if you are using you main/only computer, downgrading iTunes may be quite daunting. Apple significant changes to how iTune’s communicates with the iPhone in iTunes 8.x but I think at this time, that version is now supported.

3. If you are using Mac OS X (10.5), make sure you use a powered USB hub instead of connecting directly to the Mac. In between my first and second successful Pwn, I upgraded my Mac and this problem caused me some issues.

4. Make sure your iPhone cable is in pristine shape. I think this is the issue that caused me the most problems. As I continued to run into more iTunes errors and Google searches lead me further into uncharted advice from the web, I eventually came across one article that pointed out how a faulty cable caused their issues. So I grabbed a spare and everything worked.

5. Make sure you have a Device Support folder and it does not have an .ipsw file from a previous restore attempt. On the Mac, it is located at /Users/<username>/Library/iTunes/Device Support and on Windows XP it is at c:\documents and settings\<username>\Application Data\Apple Computer\iTunes\Device Support.

While creating the stage1.ipsw and stage2.ipsw can be a bit tedious, iTunes errors are the most frustrating because you really don’t know what is going on and there are way too many posts on the web from people you should probably not take advice from. If you are determined to get around these problems, follow Zdziarski’s steps closely, make sure your equipment and software are up to date and be tenacious.

3. Forensic Acquisition

After you successfully apply the Stage 1 and Stage 2 custom firmware bundles, things get a bit easier. Zdziarski details several ways to acquire the dd image from the iPhone but I chose to create an Ad-Hoc network with my Mac and configure the iPhone with a static IP. If you choose to use a WAP, then you may consider more secure options such as using encryption or tunneling over SSH.

1. Join iPhone to Wi-Fi network, set static IP if needed

2. Disable Auto-Lock under Settings -> General -> Auto-Lock by setting to Never

3. Open 2 ssh sessions as root/alpine to the iPhone from your host computer. In the first session, start a continuous ping back to the host computer.

4. In the second ssh session, you un-mount (umount) the user directory and re-mount read-only.

5. You then can take an MD5 hash of the user partition.

6. On the host computer, you will start a netcat session to accept the dd image from the iPhone over the Wi-Fi network

7. Finally, on the iPhone, you will send the dd image of /dev/rdisk0s2 over the network (using netcat) to the host computer.

My 16GB iPhone took about 5hrs and 15mins to upload the resulting dd image. The MD5 hash of the drive also took several hours however everything matched in the end. Below is output from the second session on the iPhone during the acquisition:

andrew-hoogs-mac:~ ahoog$ ssh -l root 192.168.0.2

The authenticity of host '192.168.0.2 (192.168.0.2)' can't be established.

RSA key fingerprint is ba:d0:4f:d9:5c:2f:a1:5d:7c:16:79:44:9e:5b:6e:53.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '192.168.0.2' (RSA) to the list of known hosts.

root@192.168.0.2's password:

-sh-3.2# cd / -sh-3.2# umount -f /private/var

-sh-3.2# mount -o ro /private/var

-sh-3.2# /bin/dd if=/dev/rdisk0s2 bs=4096 | nc 192.168.0.1 7000

3836826+0 records in

3836826+0 records out

15715639296 bytes (16 GB) copied, 19224.3 s, 817 kB/s

And the corresponding session on the Mac:

andrew-hoogs-mac:Desktop ahoog$ nc -l 7000 | dd of=./rdisk0s2 bs=4096

23349+22444701 records in

23349+22444701 records out

15715639296 bytes transferred in 19484.749921 secs (806561 bytes/sec)

Both the iPhone and the Mac reported the same MD5 has for /dev/rdisk0s2 and the dd image: 543f2ce0f118d5e1f3a6bd29c575b75f.

4. Results and Reporting

Unlike other tools which then guide the examiner through analysis of the acquired data, Zdziarski's method produces a raw dd image and can thus be imported into many forensic tools or analyzed command line.

However, the iPhone uses the HFS/X file system (fifth generation HFS) and as such many forensic tools do not yet recognize the file system. A work around is to modify the dd image a 0x0400 and change the HX to H+. This will allow any forensic tool that understands HSF+ to process the image.

ahoog@wintermute:~/iphone-img$ file rdisk0s2.img

rdisk0s2.img: Macintosh HFS Extended version 5 data last mounted by: '10.0', created: Sat Jan 3 04:43:38 2009, last modified: Sat Jan 3 19:49:38 2009, last checked: Fri Jan 2 22:43:38 2009, block size: 4096, number of blocks: 3836826, free blocks: 3829592

This is another key area where you must be able to explain what you changed, why you changed it and how it did not impact the authenticity of the image.

I chose not to import the image into any commercial forensic tools, instead opting for command line analysis on my Linux forensic workstation. I did have a problem where Linux could not fully traverse the HFS file system mounted natively and so I preformed part of the analysis in Linux but mounted the disk image file system on a Mac.

Like any standard dd image analysis, you likely want to carve files from allocated and unallocated space as well as extract all strings. Zdziarski provides a scalpel configuration file in the sample code section of his website that is tailored to recovering important iPhone files such as images, XML files, SQLite database, Plist files and more. When I ran scalpel on my Linux workstation, it processed the image in just over 8 minutes and carved 30,213 files. Below is the output from the command.

ahoog@wintermute:~/iPhone-results$ time /home/ahoog/src/scalpel-1.60/scalpel -c /home/ahoog/scalpel.conf rdisk0s2.dd Scalpel version 1.60 Written by Golden G. Richard III, based on Foremost 0.69.

Opening target "/home/ahoog/iPhone-results/rdisk0s2.dd"

Image file pass 1/2.

rdisk0s2.dd: 100.0% |**********************************| 14.6 GB 00:00 ETA

Allocating work queues...

Work queues allocation complete. Building carve lists...

Carve lists built. Workload:

gif with header "\x47\x49\x46\x38\x37\x61" and footer "\x00\x3b" --> 1 files

gif with header "\x47\x49\x46\x38\x39\x61" and footer "\x00\x3b" --> 78 files

jpg with header "\xff\xd8\xff\xe0\x00\x10" and footer "\xff\xd9" --> 1022 files

jpg with header "\xff\xd8\xff\xe1" and footer "\x7f\xff\xd9" --> 53 files

png with header "\x50\x4e\x47\x3f" and footer "\xff\xfc\xfd\xfe" --> 17 files

png with header "\x89\x50\x4e\x47" and footer "" --> 983 files

dat with header "\x44\x79\x6e\x61\x6d\x69\x63\x44\x69\x63\x74\x69\x6f\x6e\x61\x72\x79\x2d" and footer "" --> 2 files

plist with header "\x3c\x70\x6c\x69\x73\x74" and footer "\x3c\x2f\x70\x6c\x69\x73\x74" --> 9240 files

plist with header "\x62\x70\x6c\x69\x73\x74\x30\x30" and footer "" --> 11873 files

sqlitedb with header "\x53\x51\x4c\x69\x74\x65\x20\x66\x6f\x72\x6d\x61\x74" and footer "" --> 915 files

email with header "\x46\x72\x6f\x6d\x3a" and footer "" --> 5469 files

doc with header "\xd0\xcf\x11\xe0\xa1\xb1" and footer "" --> 4 files

htm with header "\x3c\x68\x74\x6d\x6c" and footer "\x3c\x2f\x68\x74\x6d\x6c\x3e" --> 545 files

pdf with header "\x25\x50\x44\x46" and footer "\x25\x45\x4f\x46\x0d" --> 0 files

pdf with header "\x25\x50\x44\x46" and footer "\x25\x45\x4f\x46\x0a" --> 1 files

amr with header "\x23\x21\x41\x4d\x52" and footer "" --> 10 files

Carving files from image.

Image file pass 2/2.

rdisk0s2.dd: 100.0% |**********| 14.6 GB 00:00 ETAProcessing of image file complete. Cleaning up... Done.

Scalpel is done, files carved = 30213, elapsed = 509 seconds.

real 8m28.869s

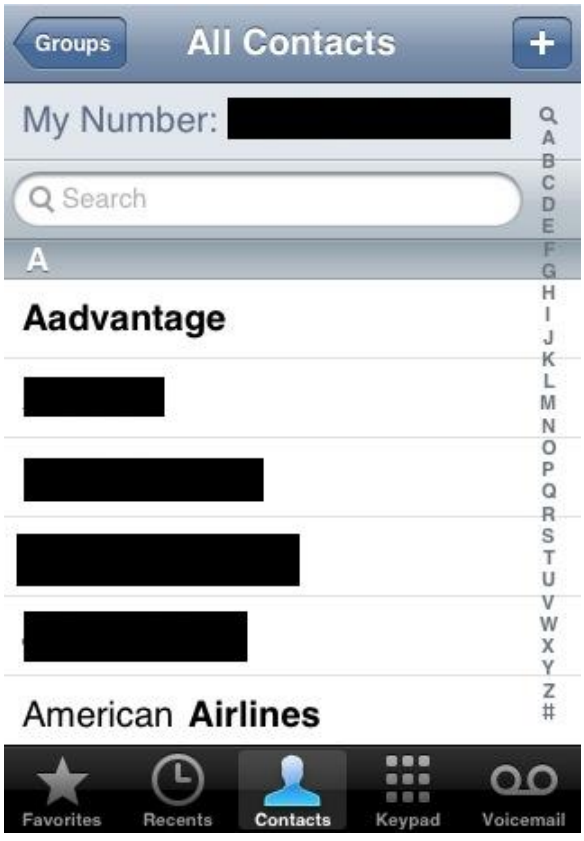
user 5m37.189s

sys 0m20.561s

As you can see, this process was fast and produced many carved files but the drawback is the potential for many false positives. However, using standard techniques such as the file command, scripts, image viewers and more, you can whittle away at the extraneous files quickly. The results were quite effective and provided insight into iPhone usage that no other tool could provide

Of the 2,000+ images exported, many gave detailed information about the phone and the owner. One feature of the iPhone is that screens zoom in and out as you switch between applications. To achieve this, the iPhone takes a screenshot of the iPhone just prior to changing screen and then applies the transition. Because of this, the iPhone is full of images that show what the user was viewing when they switched screens. Below are examples of some images recovered from the iPhone.

Table 7.1. Screen Images Extracted

Multiple images of Contacts were recovered allowing the examiner to possibly piece together contacts that may have been deleted and were unrecoverable.	

Screen displayed a note that was later deleted. While the note was discovered by analyzing deleted records in the SQLite database, sometimes that approach fails. This screenshot could provide a key piece of evidence in a case.	
Map image that was displayed on the iPhone	

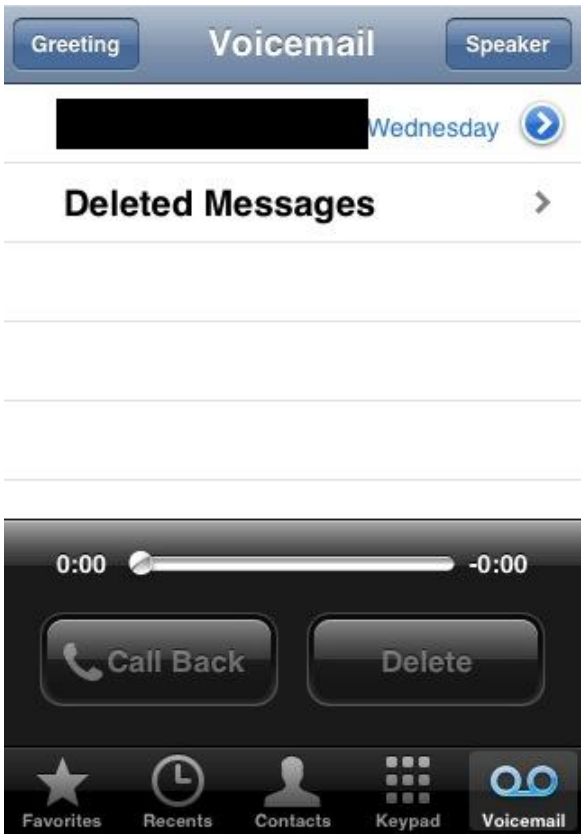
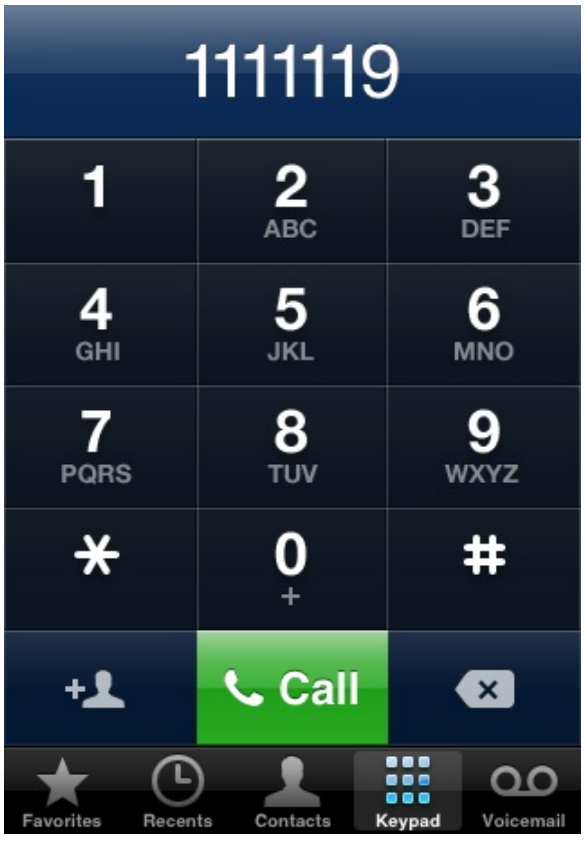
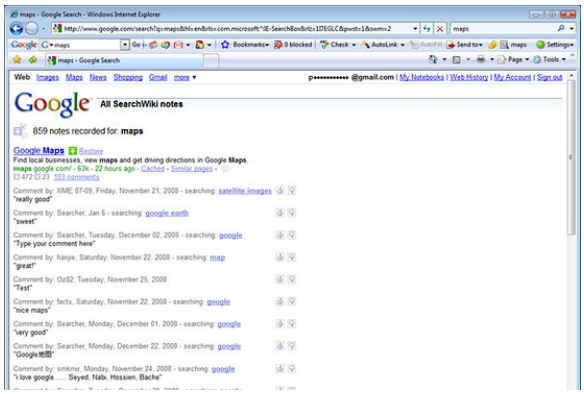
Voicemail page displaying undeleted message(s)	
Dial pad displaying phone numbers dialed	

Image likely downloaded from email (Internet Explorer)	
Email message with To/From/CC and full message	

Another useful technique is to run strings on the dd image. Below is the output from that command:

wintermute:/home/ahoog/iPhone-results# time strings -t d rdisk0s2.dd > rdisk0s2.dd.str

real 4m29.952s

user 3m54.267s

sys 0m18.137s

The resulting file had 14,400 lines and could be easily searched to located important information.

As with other tools, analysis of the SQLite databases directly (using a hex editor, strings and other techniques) revealed some deleted rows that were not yet purged from the file. Quite notably, Zdziarski's technique was the only approach that found the 1,000+ email messages on the system. This highlights the drawback to techniques that do not perform a bit-by-bit copy. In scenarios where a logical copy is made, the technique can only produce information that is presented by the operating system, transfer protocols or known by the forensic vendors. I suspect every other product missed the emails because they were either stored in non-standards areas since they were downloaded via Microsoft's Active Sync or because the iPhone transfer protocol does not expose these files.

If you assume some criminals are technically savvy and have some financial support, it would be quite simple to develop an application for the iPhone that stores information they want in non-standard ways, thus thwarting any logical data extractions.

Zdziarski's technique was also the only approach to recover deleted and undeleted voicemails (11 in total). This can be a key aspect of any investigation. Similarly, his technique also recovered various office documents including an Excel expense report, 1 PowerPoint presentation, 546 HTML documents and two court documents that were stored as a Word document and PDF. Given the iPhone's agility in handling these types of documents and their proliferation in email messages, the inability to recover them is a real deficit for any forensic tool. Finally, Zdziarski's technique also recovered over 21,000 XML and Plist files. Undoubtedly there are some repeats and false-positives however I believe a more thorough investigation of these files will results in even more detailed information being discovered.

5. Matrix of Results

The following are the results from the Zdziarski tests.

Table 7.2. Zdziarski Matrix of Results

Scenario	Zdziarski Results	Ranking	Results
Call Logs	100 (129 from deleted rows SQLite db)	4	Above
SMS	262 (400+ from deleted rows in SQLite db)	4	Above
Contacts	1282 (found 2 deleted in dd image and strings, 14 w/ images)	4	Above
Email	296 in SQLite message table, 472 message_data, 990 sqlite_sequence, 4474 From:)	5	Above
Calendar	3073	4	Above
Notes	2 (1 deleted)	5	Above
Pictures	1002 PNGs, 81 GIFs, 1078 JPGs. Includes many deleted images, images from Apps, web, etc.	5	Above
Songs	46	3	Meet
Web History	2 (plus most/all deleted on file system)	5	Above
Bookmarks	5	3	Meet
Cookies	15	5	Above
App Info	Deleted and Undeleted	5	Above
Google Maps	5 histories	3	Meet
Voicemail	11	5	Above
Password	7	3	Meet
Plists/XML	21113	5	Above
Phone Info	Yes	3	Meet
Video	1	3	Meet
Podcasts	1	3	Meet

Speed Dials	Yes	3	Meet
VPN	Yes	3	Meet
Bluetooth	Yes	3	Meet
GPS	Yes	3	Meet
File Hashes	Yes	3	Meet
You Tube	70	5	Above
HTML	546 (includes some emails)	5	Above
Office Docs	1 spreadsheet, 1 PowerPoint, 1 WordDoc, 1 PDF	5	Above

6. Conclusions

The forensic technique outlined by Jonathan Zdziarski provides the most sought after asset in any computer forensic investigation, a bit-by-bit copy of the original media. While there are questions that might arise about how the technique is performed, what it might change and the degree of difficultness, it is unarguably the most accurate of any tool tested. By analyzing the resulting image, an examiner can discover a wealth of information other tools simply cannot provide and they can use the forensic analysis tools and approaches they utilize in standard hard drive based forensic investigations.

The following ranking establishes Zdziarski's technique's overall rating of 3.3 on the four criteria established at the beginning of this white paper.

Table 7.3. Zdziarski Rankings

Area	Weight	Rank
Installation	0.1	2.0
Acquisition	0.2	3.5
Reporting	0.3	2.0
Accuracy	0.4	4.0
TOTAL		3.3

Chapter 8. .XRY (2.6/5.0)

by MicroSystemation

1. Summary (from company information)

The .XRY/.XACT 4.1 is a mobile forensic system that performs logical data acquisition as well as physical dumps. The systerm is enclosed in a small brief case and contains: the USB 2.0 communication unit, a license key dongle, all current cables in cable holders, SIM card reader, rewritable SIM id-Cloner examination cards, a read only memory card reader and a CD with all software. All new cables that are released during the license period are included in the license fee as well as all software updates.

Both systems use the .XRY report format. The .XRY reader is available for free. This reader insures the forensic report is secure.

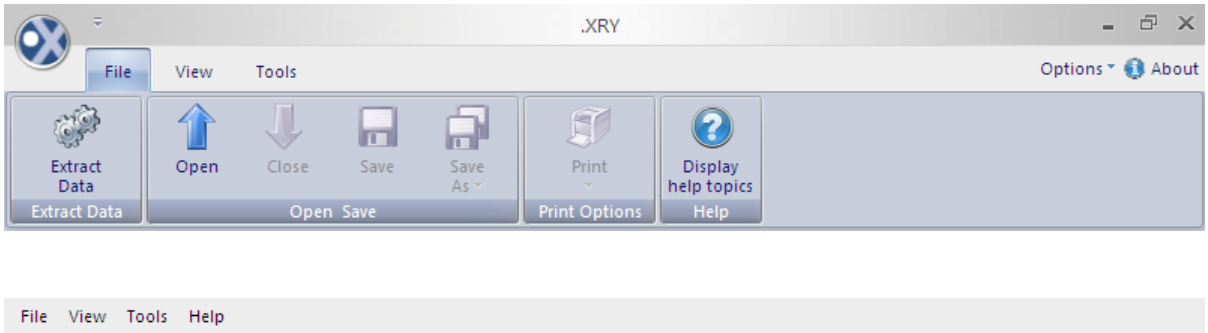
2. Installation

The .XRY System came in its own hard case containing the software on a CD, hardware unit and cords, SIM readers, and 46 different cables to connect different mobile devices. Once the CD was inserted the software loaded using the standard Windows InstallShield Wizard. Before using the device a protection key must be installed on the computer from an enclosed dongle.

Our protection key expired before our test and requesting a new one was necessary. To do this Micro Systemation provides directions in the Help section of the software. Since Micro Systemation notified us of our expired protection key they took the initiative to send us this information in a .pdf. The directions described how to do update the protection key on both computers that do and do not have Code Meter installed. Code Meter was installed on the computer used and the process was straightforward. We emailed the request for a new protection key to Micro Systemation over the weekend and new one was sent to us on Monday.

We tested version 4.10. The user interface(UI) was familiar and easy to understand. The UI had both Ribbons of large shortcut buttons organized under tabs as well as traditional drop down menus.

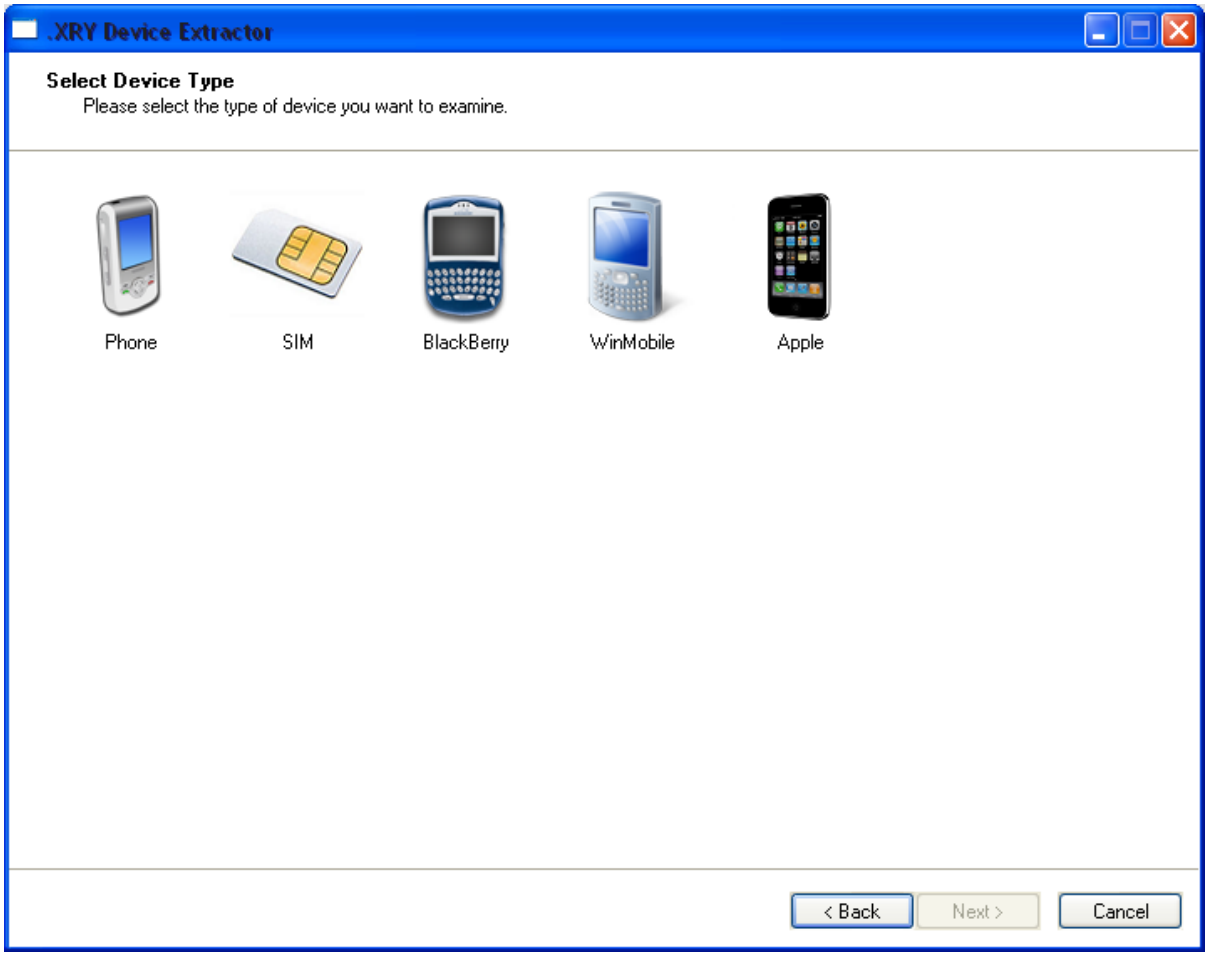
Figure 8.1. .XRY Ribbons UI



3. Forensic Acquisition

The acquisition process was simple and fast. The program first asks you which type of device you are extracting information from.

Figure 8.2. Supported Devices



At the end of the extraction process a summary of the results was displayed. The results of our test revealed that the extraction process finished with errors. The errors can be found in the system log. The error log reflected that the errors were a result of the device being locked.

After selecting the iPhone the following information screen for the iPhone appeared. It is important to note that the information screen states Email messages will not be retrieved unless the phone is unlocked. The iPhone tested was not unlocked.

Figure 8.3. .XRY iPhone Information

The next couple of screens allows the user to select only certain items to be extracted and to choose where the results are saved. The extraction process then begins and the status is shown on the screen.

Figure 8.4. .XRY Extraction Options

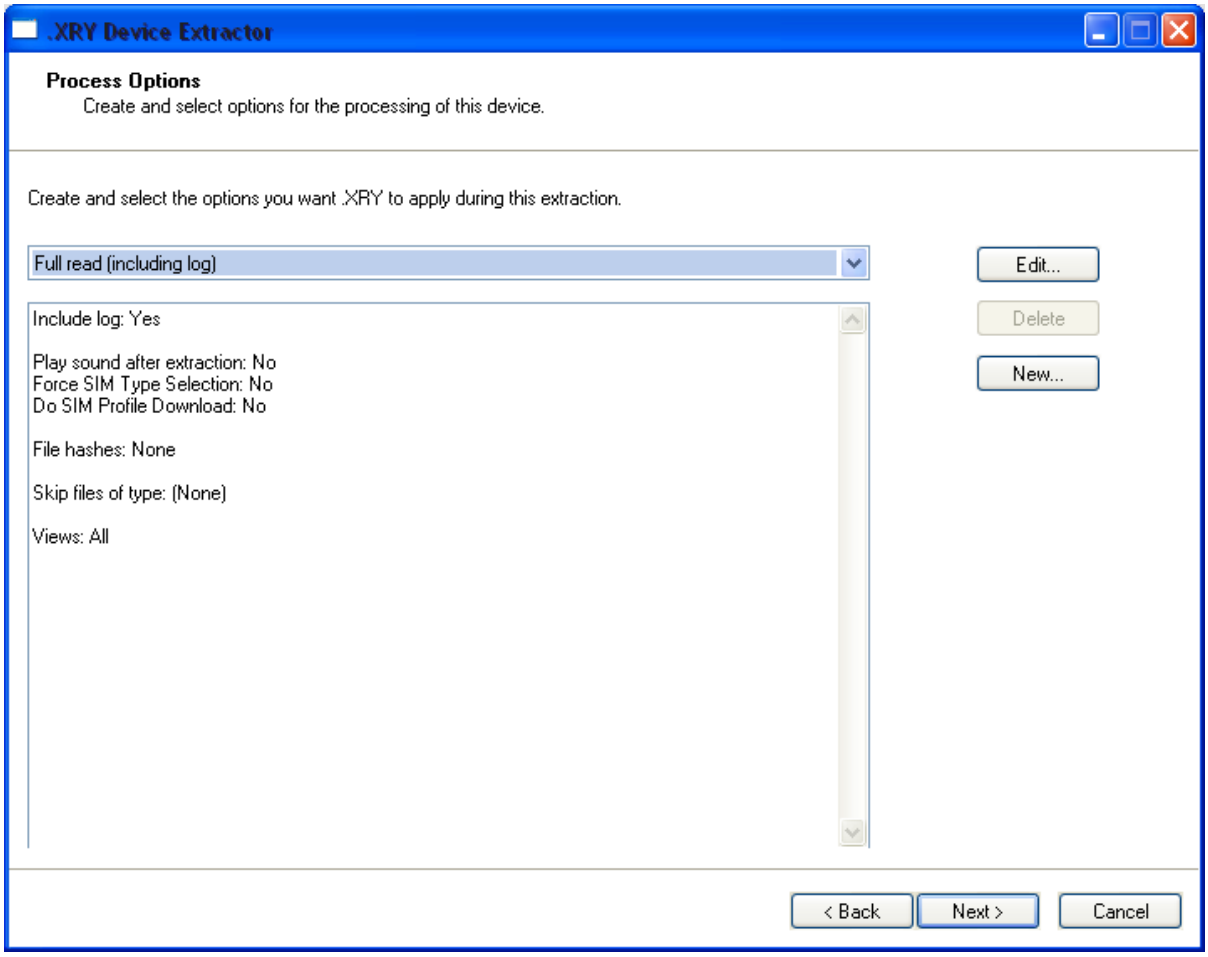
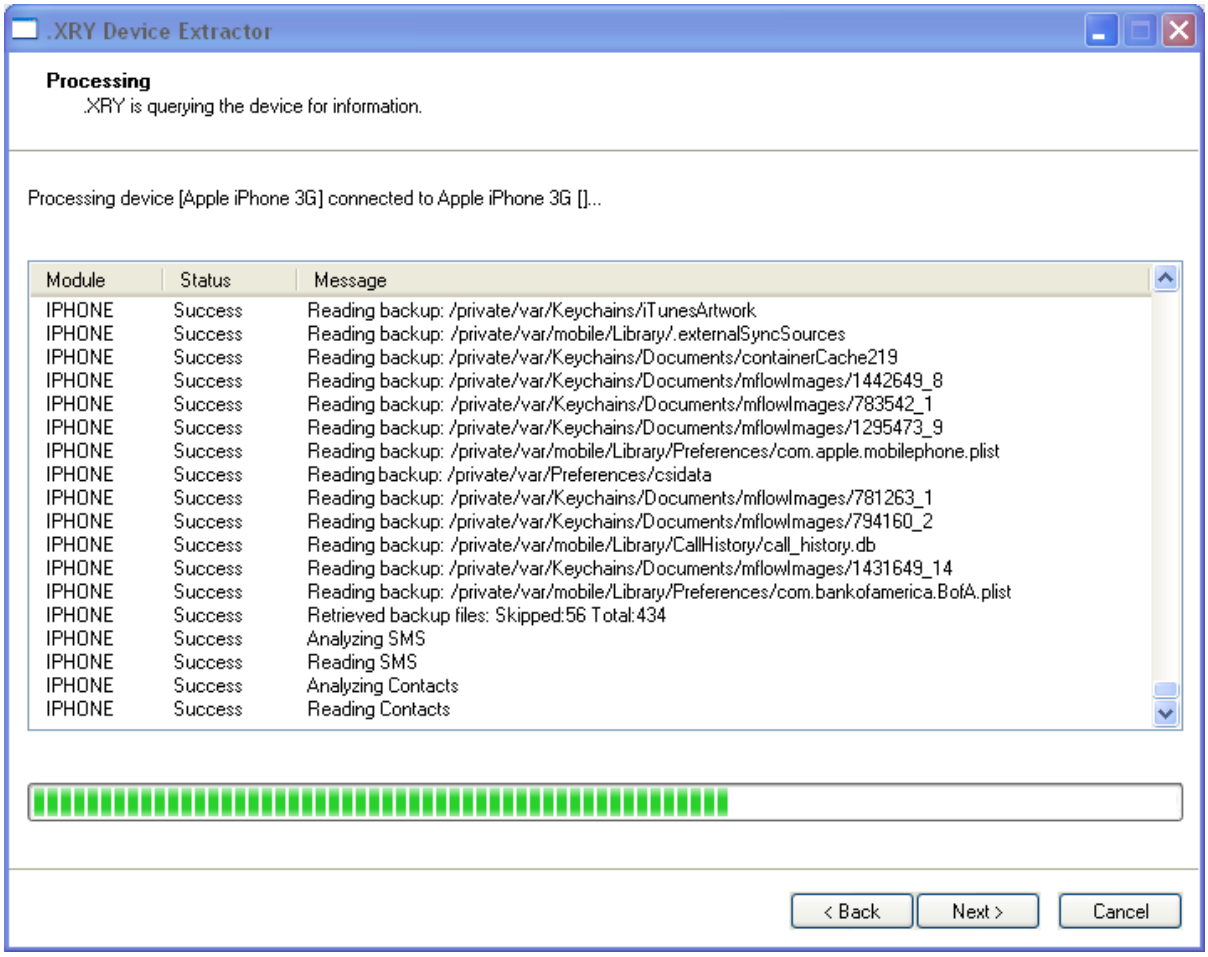
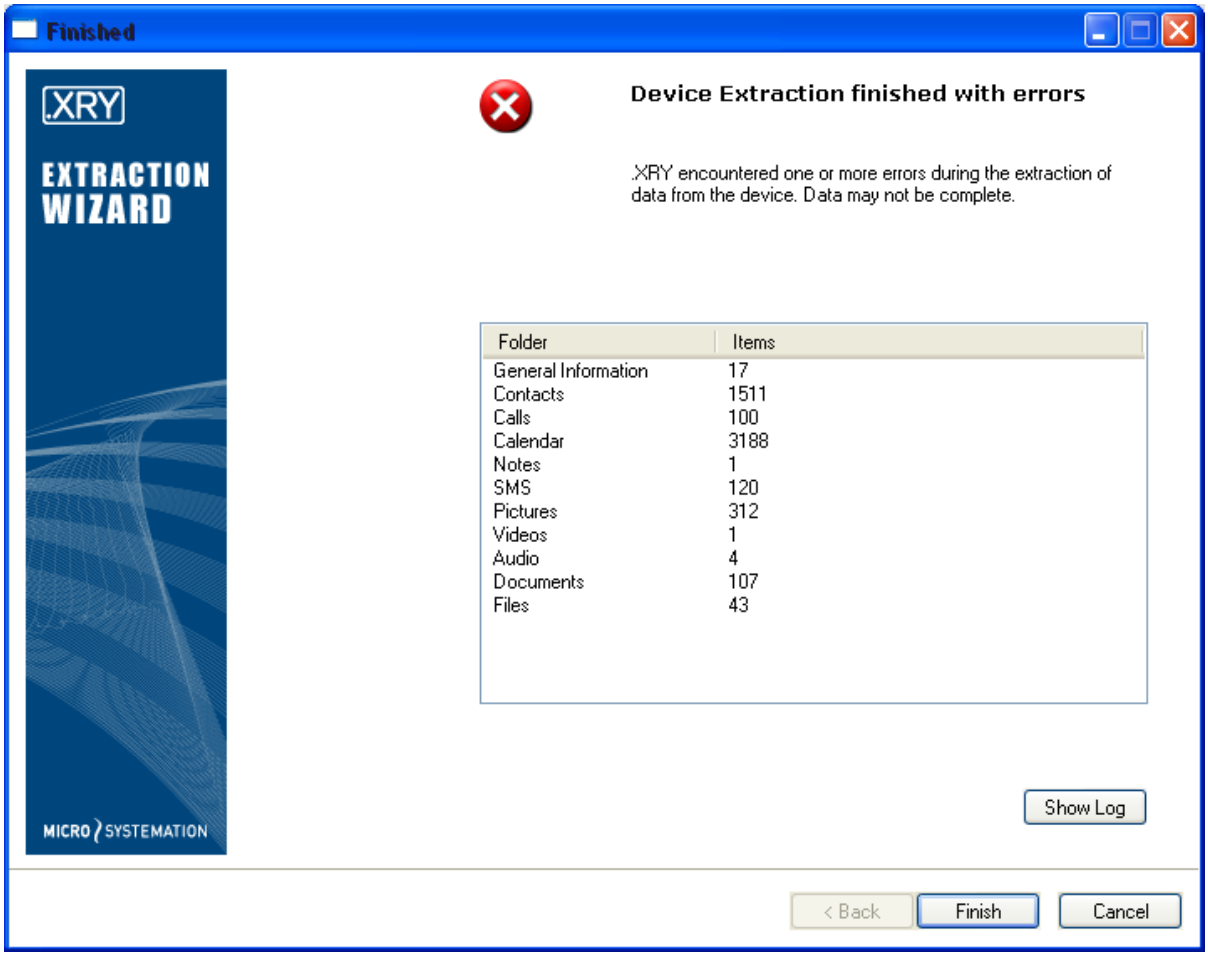


Figure 8.5. .XRY Extraction Progress



At the end of the extraction process a summary of the results was displayed. The results of our test revealed that the extraction process finished with errors. The errors can be found in the system log. The error log reflected that the errors were a result of the device being locked.

Figure 8.6. .XRY Extraction Finished



4. Results and Reporting

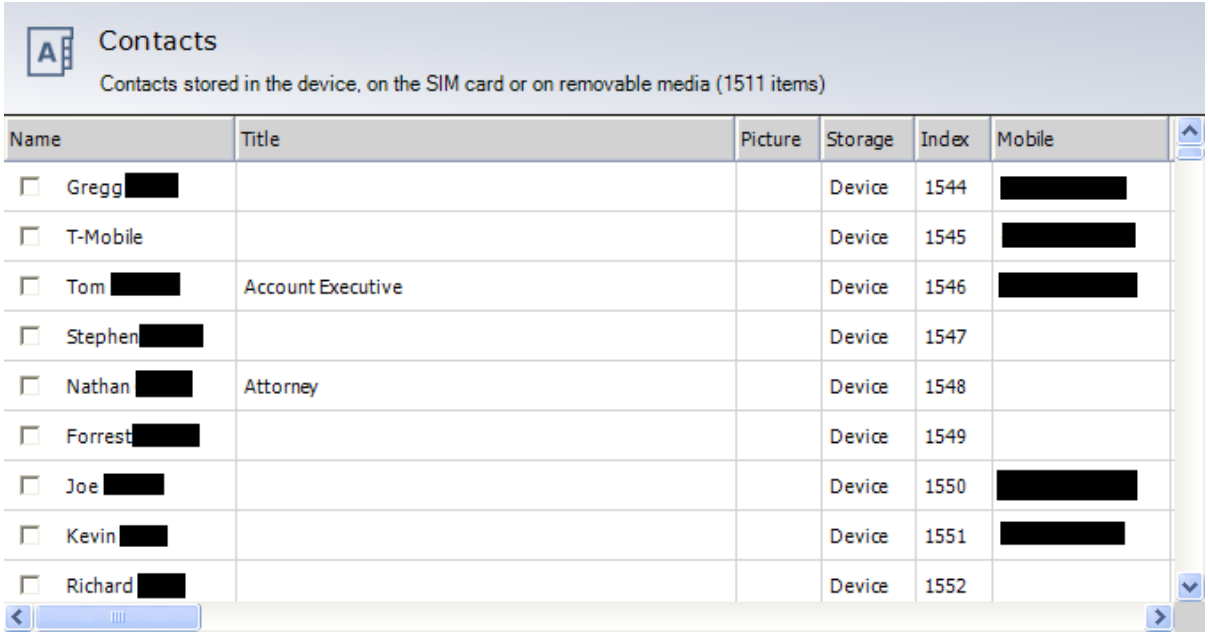
.XRY pulled some very valuable and useful data from the iPhone such as Contacts, SMS messages and images with GPS locations. However, because the iPhone was not 'jailbroken' the emails were not retrieved.

.XRY displayed the extracted information in a simple to use report. The pane on the left side of the screen lists the different categories of items retrieved from the mobile device and the main pane has the detailed information for the selected category.

The Case Data and Summary categories contained basic information about when the extraction was completed, what version of software was used. The Case Data section also contained an area for making notes. The General Information section contained information regarding the phone and carrier, i.e. phone and sim ID numbers.

The Contacts section was easy to read and contained the name, multiple number fields, address fields, email, and notes for each contact.

Figure 8.7. .XRY Contacts Display

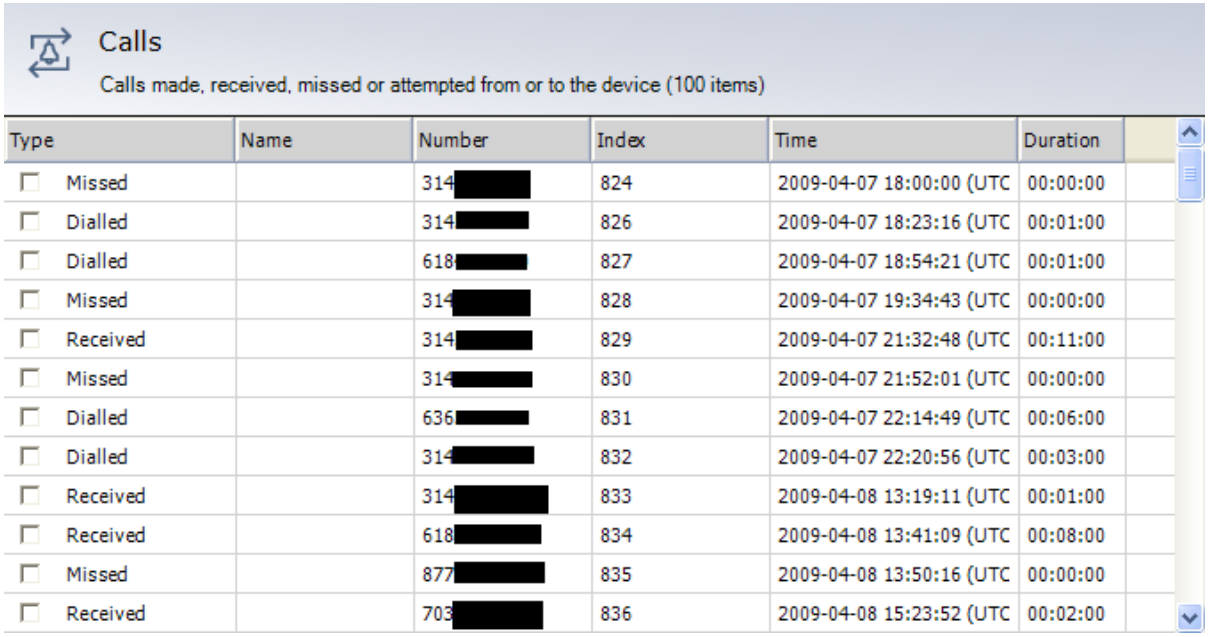


Contacts

Contacts stored in the device, on the SIM card or on removable media (1511 items)

Name	Title	Picture	Storage	Index	Mobile
Gregg [redacted]			Device	1544	[redacted]
T-Mobile			Device	1545	[redacted]
Tom [redacted]	Account Executive		Device	1546	[redacted]
Stephen [redacted]			Device	1547	
Nathan [redacted]	Attorney		Device	1548	
Forrest [redacted]			Device	1549	
Joe [redacted]			Device	1550	[redacted]
Kevin [redacted]			Device	1551	[redacted]
Richard [redacted]			Device	1552	

Figure 8.8. .XRY Call Logs



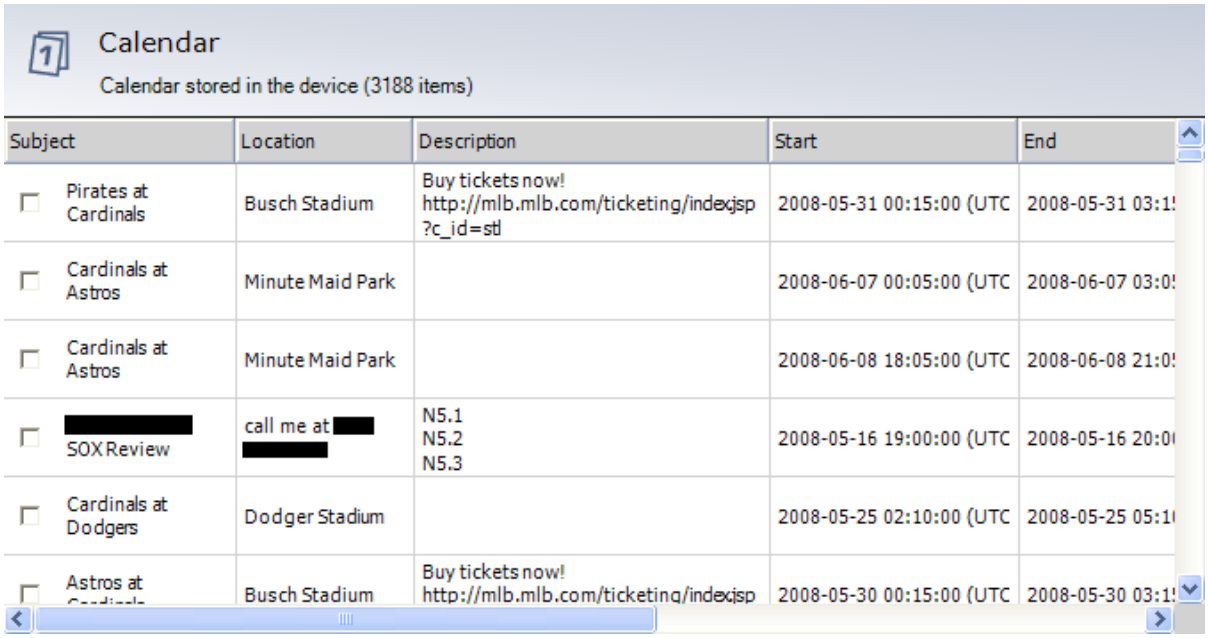
Calls

Calls made, received, missed or attempted from or to the device (100 items)

Type	Name	Number	Index	Time	Duration
Missed		314[redacted]	824	2009-04-07 18:00:00 (UTC	00:00:00
Dialled		314[redacted]	826	2009-04-07 18:23:16 (UTC	00:01:00
Dialled		618[redacted]	827	2009-04-07 18:54:21 (UTC	00:01:00
Missed		314[redacted]	828	2009-04-07 19:34:43 (UTC	00:00:00
Received		314[redacted]	829	2009-04-07 21:32:48 (UTC	00:11:00
Missed		314[redacted]	830	2009-04-07 21:52:01 (UTC	00:00:00
Dialled		636[redacted]	831	2009-04-07 22:14:49 (UTC	00:06:00
Dialled		314[redacted]	832	2009-04-07 22:20:56 (UTC	00:03:00
Received		314[redacted]	833	2009-04-08 13:19:11 (UTC	00:01:00
Received		618[redacted]	834	2009-04-08 13:41:09 (UTC	00:08:00
Missed		877[redacted]	835	2009-04-08 13:50:16 (UTC	00:00:00
Received		703[redacted]	836	2009-04-08 15:23:52 (UTC	00:02:00

The Call Log section displayed information regarding whether the call was dialed, received or missed. The date and number were also displayed. However, only one record displayed the corresponding name from the Contacts. The device itself lists the corresponding name on most of the entries.

Figure 8.9. .XRY Calendar



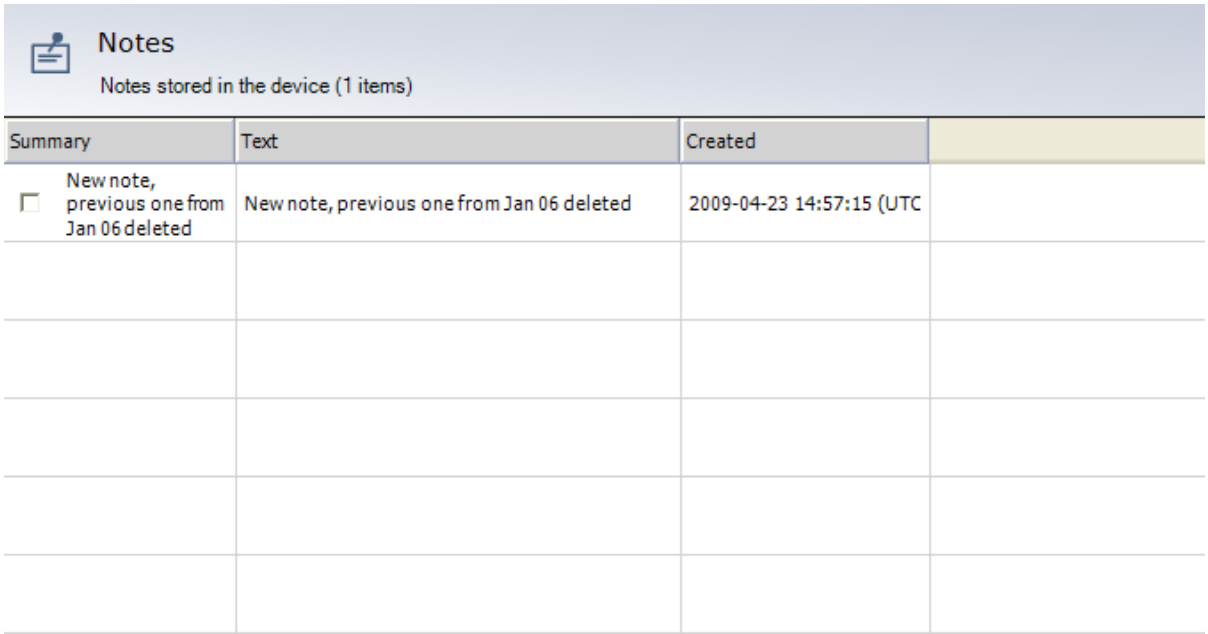
Calendar

Calendar stored in the device (3188 items)

Subject	Location	Description	Start	End
Pirates at Cardinals	Busch Stadium	Buy tickets now! http://mlb.mlb.com/ticketing/index.jsp?c_id=stl	2008-05-31 00:15:00 (UTC	2008-05-31 03:1
Cardinals at Astros	Minute Maid Park		2008-06-07 00:05:00 (UTC	2008-06-07 03:0
Cardinals at Astros	Minute Maid Park		2008-06-08 18:05:00 (UTC	2008-06-08 21:0
[redacted] SOX Review	call me at [redacted]	N5.1 N5.2 N5.3	2008-05-16 19:00:00 (UTC	2008-05-16 20:0
Cardinals at Dodgers	Dodger Stadium		2008-05-25 02:10:00 (UTC	2008-05-25 05:1
Astros at [redacted]	Busch Stadium	Buy tickets now! http://mlb.mlb.com/ticketing/index.jsp	2008-05-30 00:15:00 (UTC	2008-05-30 03:1

The extracted Calendar information was presented in an easy to read format.

Figure 8.10. .XRY Note



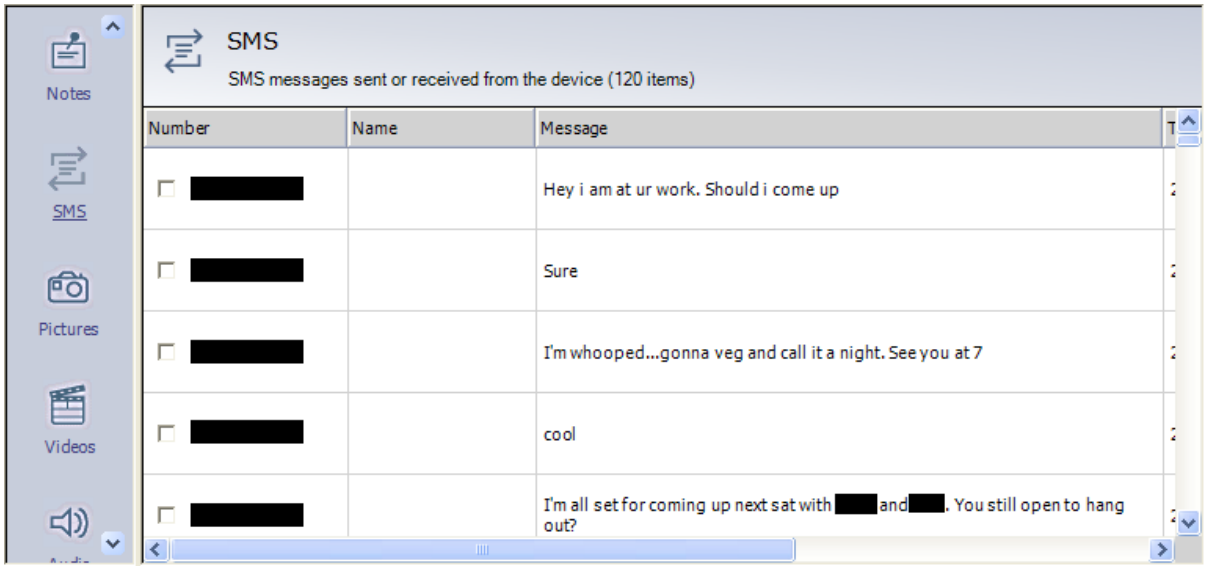
Notes

Notes stored in the device (1 items)

Summary	Text	Created
New note, previous one from Jan 06 deleted	New note, previous one from Jan 06 deleted	2009-04-23 14:57:15 (UTC

The text of the note is displayed on the details pane. There was a note that was created and then deleted, the program did not retrieve the deleted note.

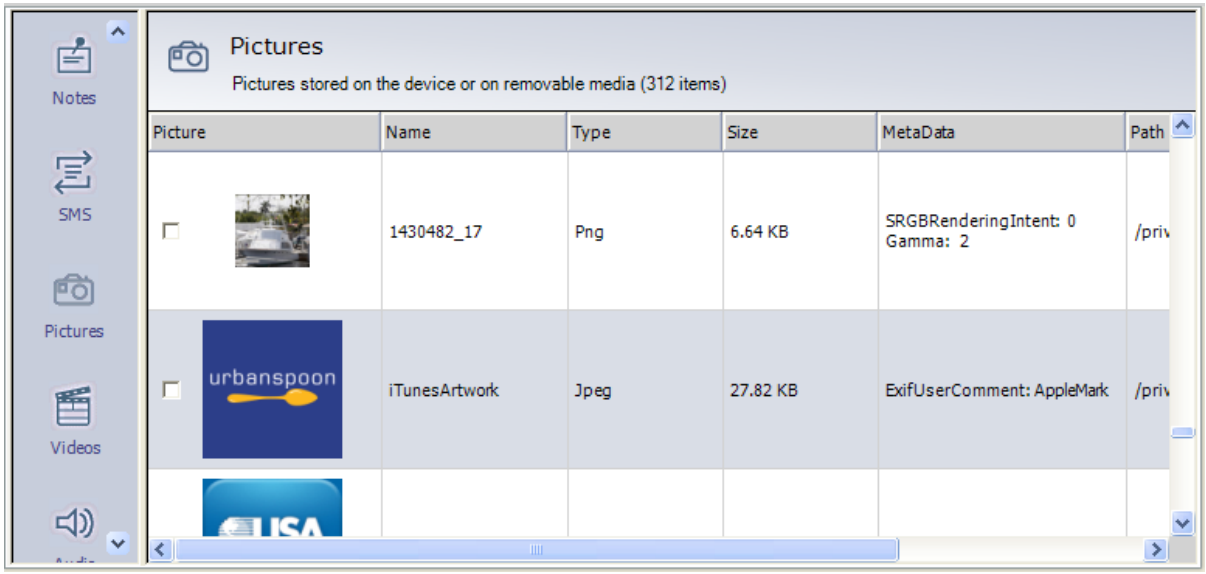
Figure 8.11. .XRY SMS



For this test two SMS messages were deleted. These two deleted messages were not recovered. However, the extracted SMS did contain the number, message and whether the message had been read or not. Again, like the Call Logs, the name from the corresponding Contacts file was not listed.

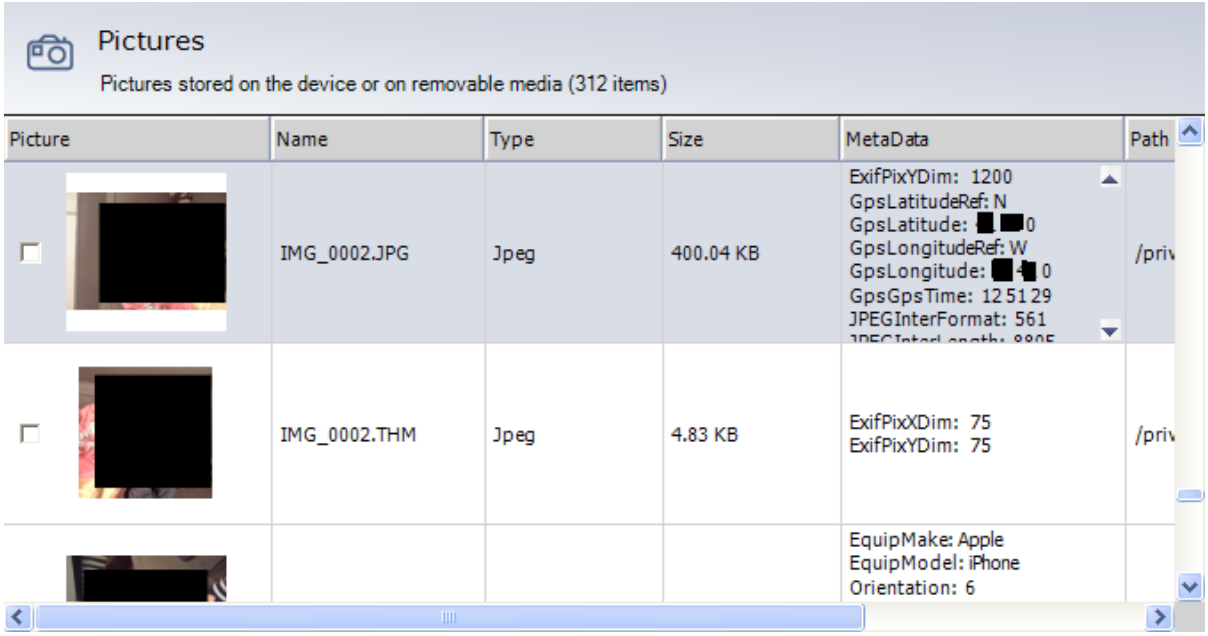
The area that .XRY excelled in was the extraction of photos from the iPhone. Unfortunately, it did not perform perfectly. A photo that was sync'd to the iPhone could not be found in the results.

Figure 8.12. .XRY Pictures



.XRY extracted images from the iPhone that were displayed during web-browsing as well as icons from the iPhone. Probably the most interesting bit of information extracted using .XRY was the GPS coordinates for photos taken using the iPhone. GPS coordinates were included in the MetaData, see below.

Figure 8.13. .XRY Photos with GPS Coordinates



.XRY also extracted audio and video files saved on the iPhone. I was able to immediately play the audio files but was unable to view the video. The audio played using Windows Media player but QuickTime was needed for the video file.

Documents and other files were also extracted. 107 documents and 43 files were extracted from the iPhone. The documents included Cookies and other .plist files. The Files section contained .db files including a search history. From these files were were able to find some of the bookmarks, cookies, a value saved in the memory of the caluclator, web browsing history, recent web searches, the speed dial file, and a location that was searched with a GPS location.

Figure 8.14. .XRY other Documents

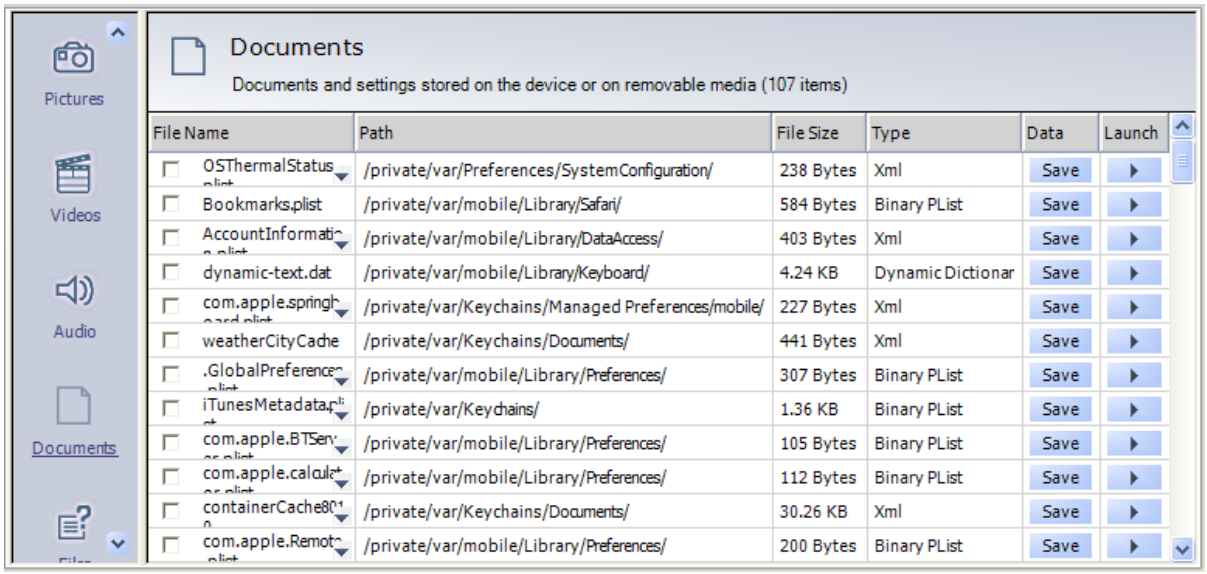
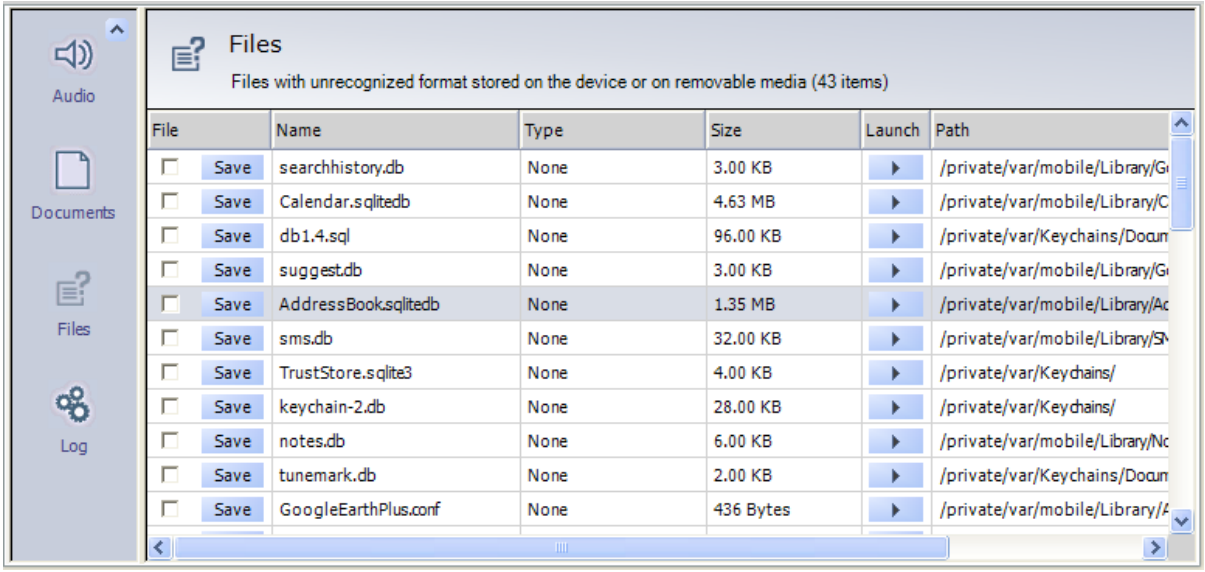


Figure 8.15. .XRY other Files



5. Matrix of Results

The following are the results from the .XRY extraction.

Table 8.1. .XRY Matrix of Results

Scenario	.XRY Results	Ranking	Results
Call Logs	100	3	Meet
SMS	120 (all retrieved, deleted not recovered)	3	Meet
Contacts	1511	3	Meet
Email	0 (an old address not being used was found but not the primary email address or emails themselves)	1	Below
Calendar	3188	3	Meet
Notes	1	3	Meet
Pictures	312 (photos taken with iPhone included GPS coordinates)	4	Above
Songs	none loaded podcasts retrieved	3	Meet
Web History	Yes, 28 were listed. Also listed recent searches.	4	Above
Bookmarks	2 (preloaded bookmarks were not listed)	3	Meet
Cookies	89	3	Meet
App Info	Some apps left evidence	2	Below
Google Maps	1 Address record and GPS location	3	Meet
Voicemail	0	0	Below

Password	None found	0	Below
Plists/XML	Many retrieved	3	Meet
Phone Info	Yes	3	Meet
Video	1	3	Meet
Podcasts	4	3	Meet
Speed Dials	Found programmed speed dial in plist	3	Meet
VPN	0	0	Below
Bluetooth	0	0	Below
GPS	Coordinates found in both images and plist. Specific info from the GPS not pulled.	3	Meet
File Hashes	An available option	3	Meet
You Tube	0	0	Below
HTML	0	0	Below

6. Conclusions

.XRY is a simple to use, effective iPhone extraction device. Installation and extraction is easily completed by following on-screen prompts. Extraction time for .XRY is comparable or better than many of the other products we have tested. The extracted data is presented in an easy to view, organized fashion that includes a search function. Of particular interest is the presence of GPS coordinatates in the metadata of extracted images taken with the iPhone.

The following ranking establishes .XRY's overall rating of 2.5 on the four criteria established at the beginning of this white paper.

Table 8.2. .XRY Rankings

Area	Weight	Rank
Installation	0.1	3.0
Acquisition	0.2	3.0
Reporting	0.3	3.0
Accuracy	0.4	2.3
TOTAL		2.6

Chapter 9. CellDEK® (2.6/5.0)

by Logicube

1. Summary (from company information)

CellDEK® has been developed in cooperation with the UK's Forensic Science Service (FSS®). The portable CellDEK is compatible with over 1600 cell phones, PDA's, and satellite navigation devices. This cell phone data extraction device is a self-contained system with a touch-screen display and allows the user to identify devices by brand, model number, dimensions and/or photographs. When the device type is selected a “smart adapter” feature then illuminates the correct USB adapter. Connectivity by infra-red and Bluetooth are also built-in. Up to 40 adapters may be stored in the system’s built-in rack.

The CellDEK captures all stored data within approximately five minutes. The CellDEK software automatically performs forensic extraction of the following data: Handset Time and Date, Serial Numbers (IMEI, IMSI), Dialed Calls, Received Calls, Phonebook (both handset and SIM), SMS (both handset and SIM), MMS messages (not available from all handsets), Deleted SMS from SIM, Calendar, Memos, To Do Lists, Pictures, Video, and Audio. The CellDEK displays the data on screen, and prompts for downloading to a portable USB device. the CellDEK also includes a Secure Erase feature that allows you to completely remove sensitive data from the laptop hard drive.

2. Installation

CellDEK is a completely self enclosed extraction device therefore installation is minimal. The small suitcase sized device contains all of the necessary connectors and internal computer with a 4.5 x 7.5 inch touch screen. The software is preloaded on the internal computer and starts immediately after signing in and agreeing to the the user agreement. The touch screen includes a pop-up on-screen touch keyboard.

Figure 9.1. CellDEK

However, updates may be necessary as it was in this test. Like most extraction devices, updates to support new mobile devices may be needed from time to time. The update needed for this test was iTunes. To extract information from an iPhone using CellDEK iTunes must be downloaded from Apple. This is done by logging onto the Apple website with a separate computer and saving the iTunes download to a USB flash drive.

After dowloading iTunes to the USB flash drive we inserted the flash drive into the USB port on the CellDEK. Simple on screen prompts guided us through iTunes installation process.

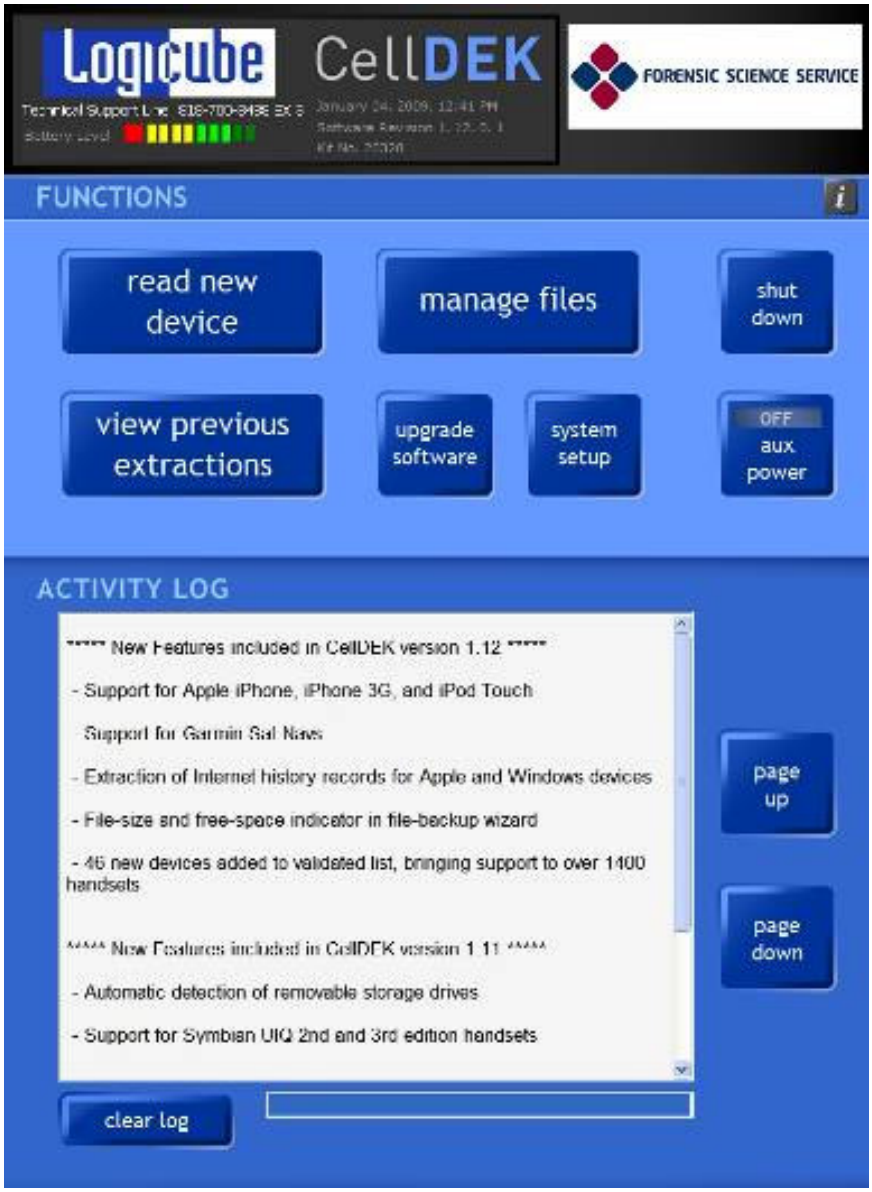
This entire process was eased by the detailed manual that was included with the CellDEK. Many of the images in this review are from the manual because we were unable to take screenshots on the self contained computer of the CellDEK.

3. Forensic Acquisition

After installing iTunes the acquisition process took less than 5 minutes.

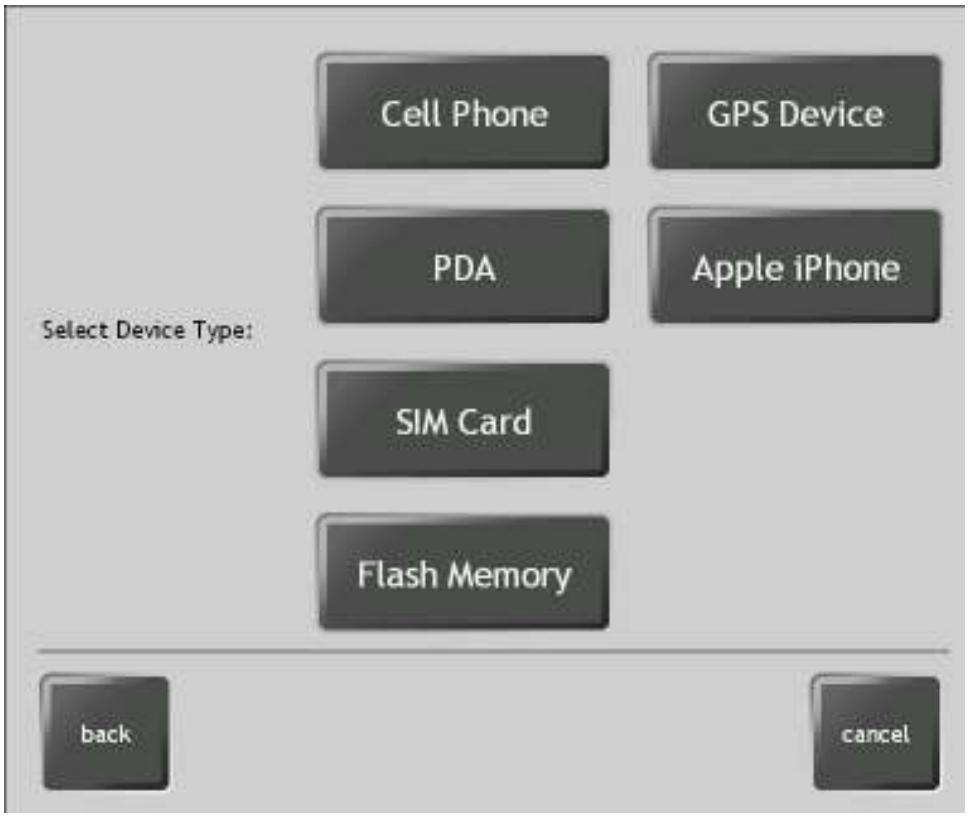
After the user agreement CellDEK loads to its main applications screen. From this screen the user can choose to manage files, view previous extractions, update software, and read a new device.

Figure 9.2. Main Application Screen



After selecting to read a new device the user is asked which kind of device is going to be read, in our test it was an iPhone.

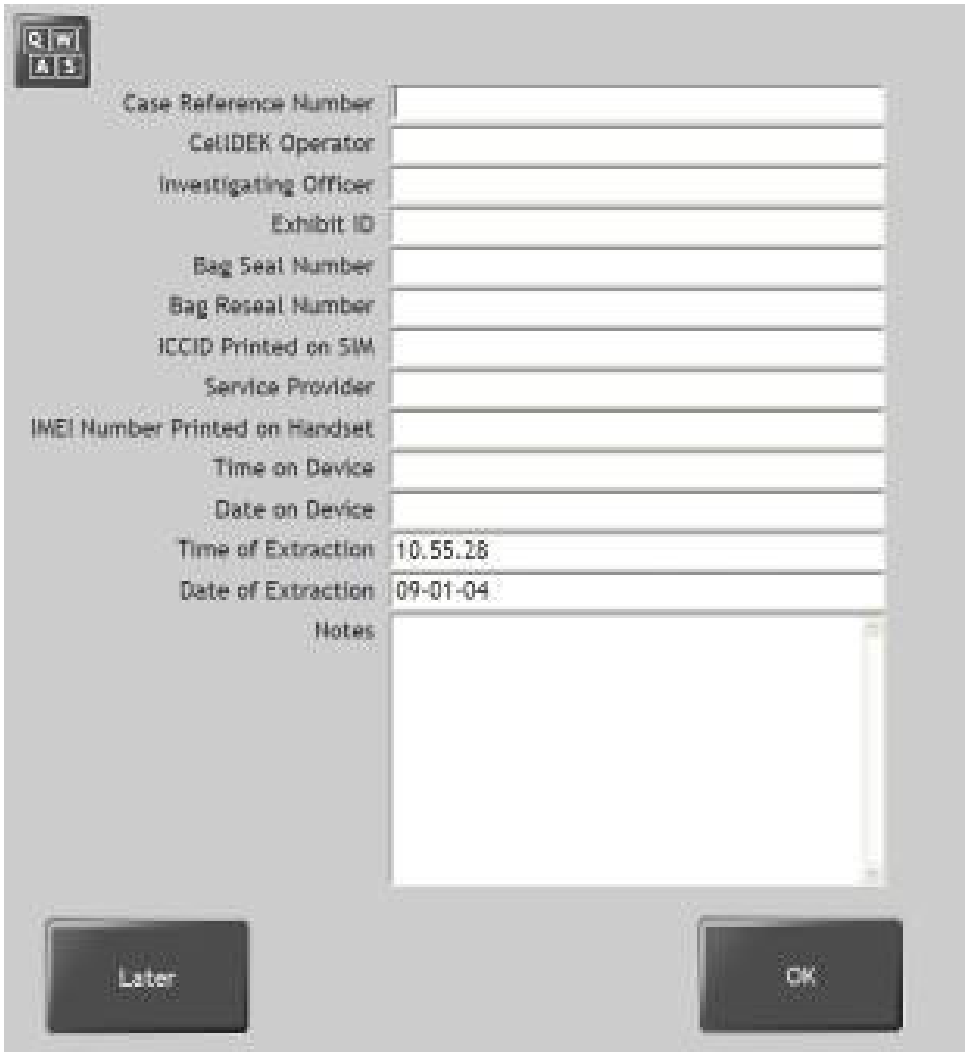
Figure 9.3. Device Type Selection



Next, an on screen prompt tells the user which cable to use by lighting up the cable connector and tells the user to insert this cable into the device bed near the screen. Once successfully connected the next on screen prompt instructs the user to connect the iPhone. After connected, an information screen is displayed listing the useful files that are to be extracted and what information they contain.

The next screen asks the user to input information about the name of the device, user, and other information that can be used for labeling and referencing the extraction report. The extraction process starts immediately following inputting the information about the test.

Figure 9.4. Case Details Input Form



Once the extraction is complete CellDEK creates and displays the report of the extraction process.

4. Results and Reporting

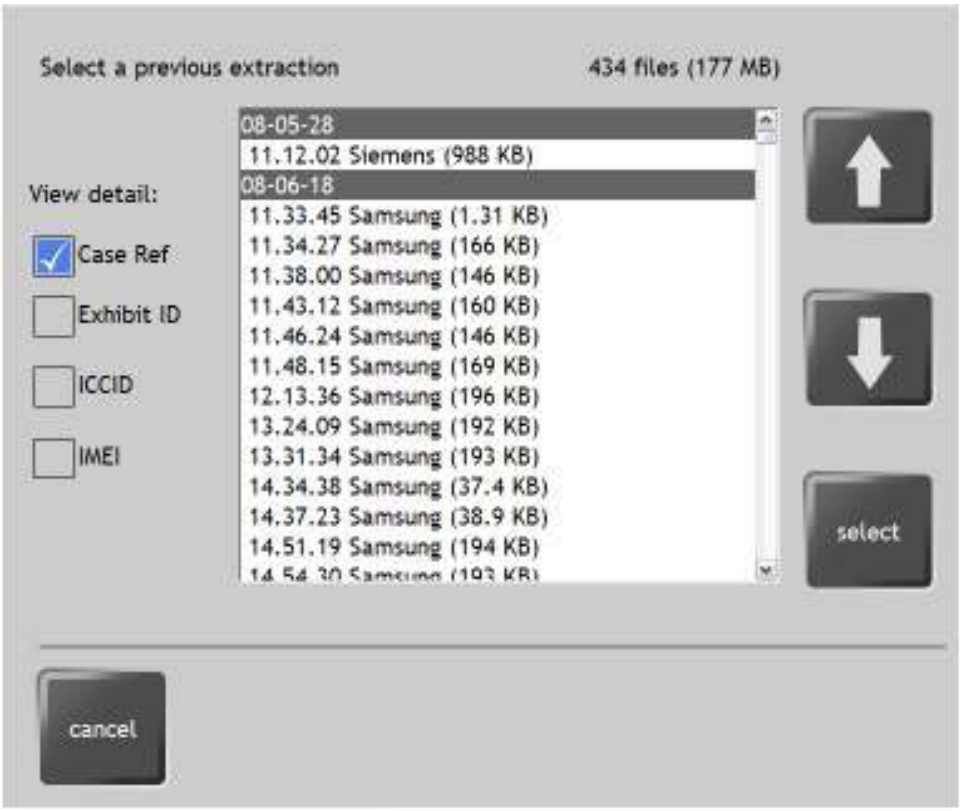
The report from the extraction process is immediately displayed on the built-in screen. The report viewer allows the user to select which area of the report they would like to view, i.e. Contacts, Messages.

Figure 9.5. Standard Data View Screen

The built-in screen and report viewer is very convenient. However the small size of the screen makes looking through large amounts of data tedious. To overcome this CellDEK allows the user to copy the files. The files may be copied to a USB flash drive or can be written to a CD if a CD writer is attached to the CellDEK via the USB connection.

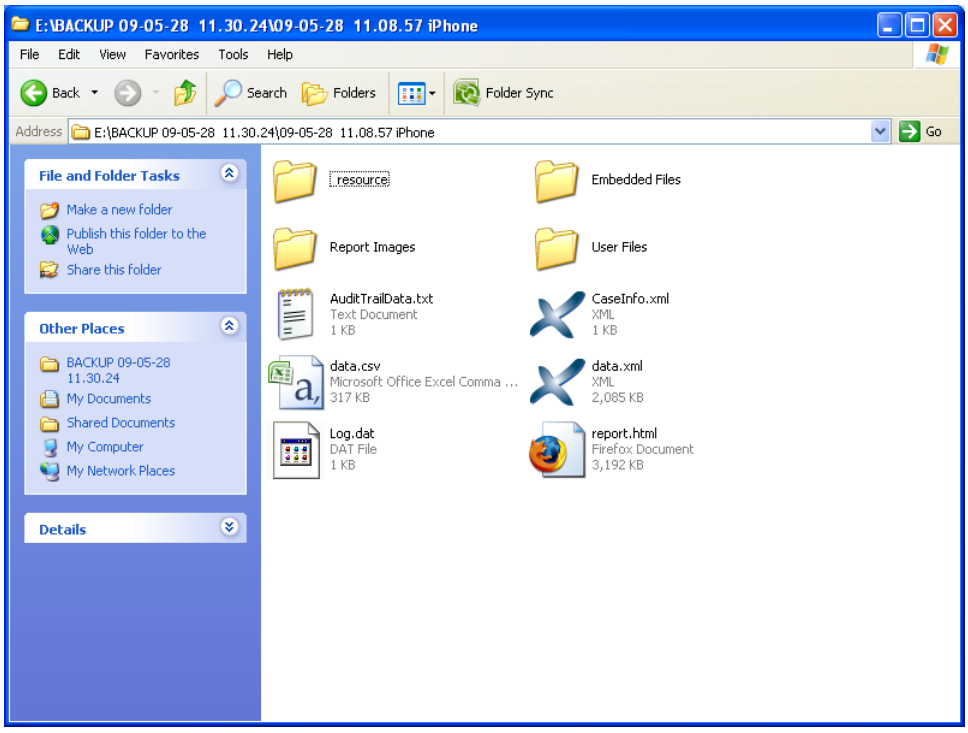
For our test we copied the files to a USB flash drive. After choosing Manage Files from the main application page the user is prompted to choose between copying files or deleting files. The next prompt gives the user the option of copying all or a single file. If copying a single file is chosen a list displaying all reports stored on the CellDEK is displayed organizing the reports by date.

Figure 9.6. Previous Extraction List



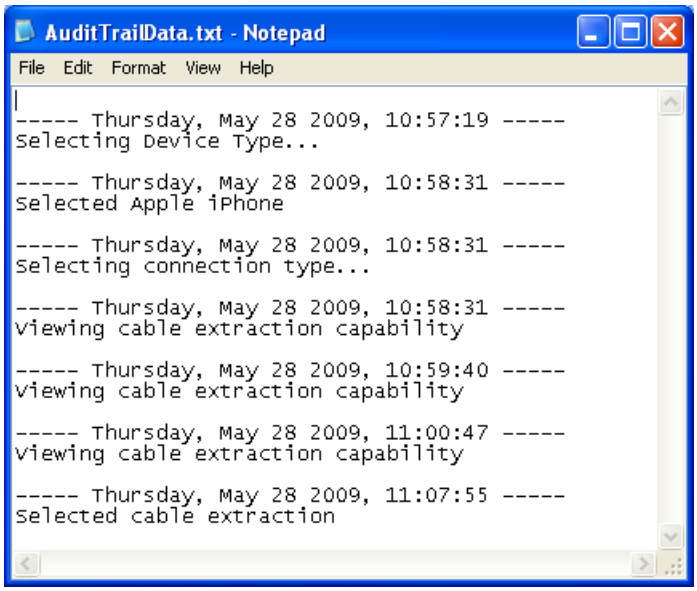
Once the report is saved on a USB flash drive it can be transferred to other computers. Unlike other extraction devices CellDEK does not require that a proprietary software be downloaded to view the extraction report. The extraction report can be viewed as an .html document. The folder containing the transferred files includes .csv and .xml version of the extraction report. The extracted files also include images and a folder of user files.

Figure 9.7. Transferred Files



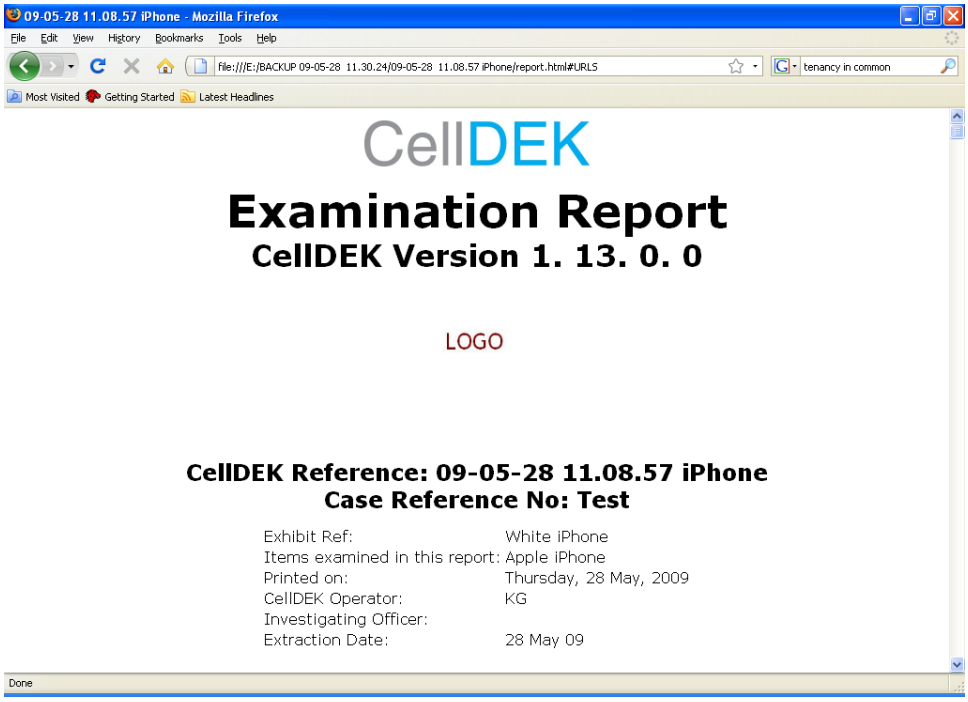
The transferred files also include an Audit Trail text file that shows the exact time that extraction occurred.

Figure 9.8. Audit Trail



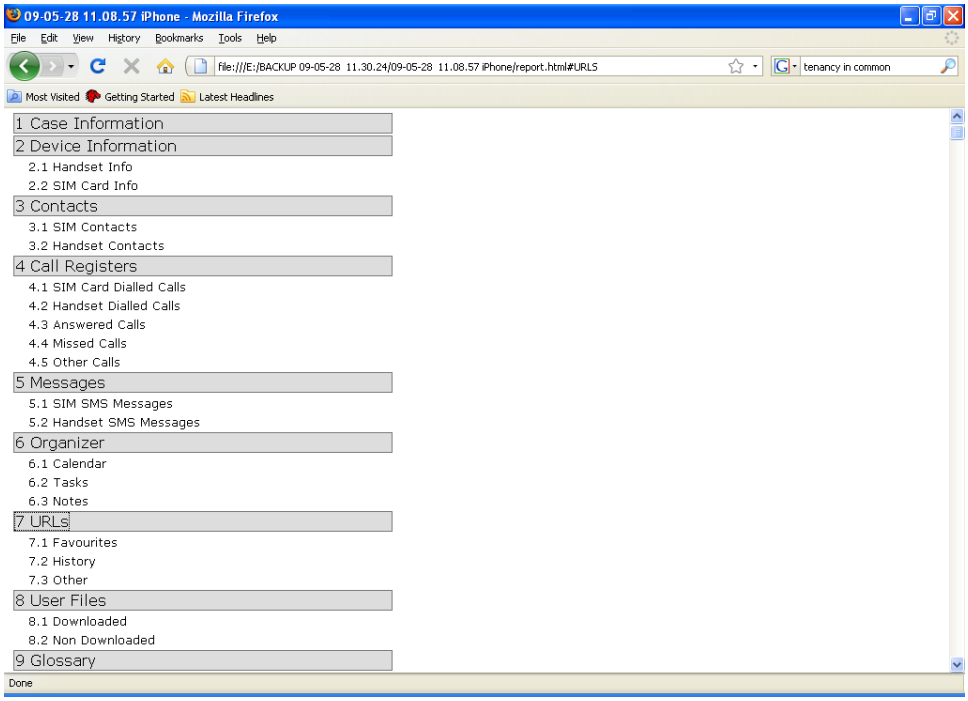
The .html version of the report is similar to the on-screen version.

Figure 9.9. .html version



The .html report does include a table of contents but clicking on the title of the section did not bring us to the section. Therefore when looking for something specific on the report the user must either use the .html browser's find function or scroll down through the large report.

Figure 9.10. .html Table of Contents



Images are available for viewing in both the .html version of the report as well as in a separate Report Images folder.

Figure 9.11. Report Image

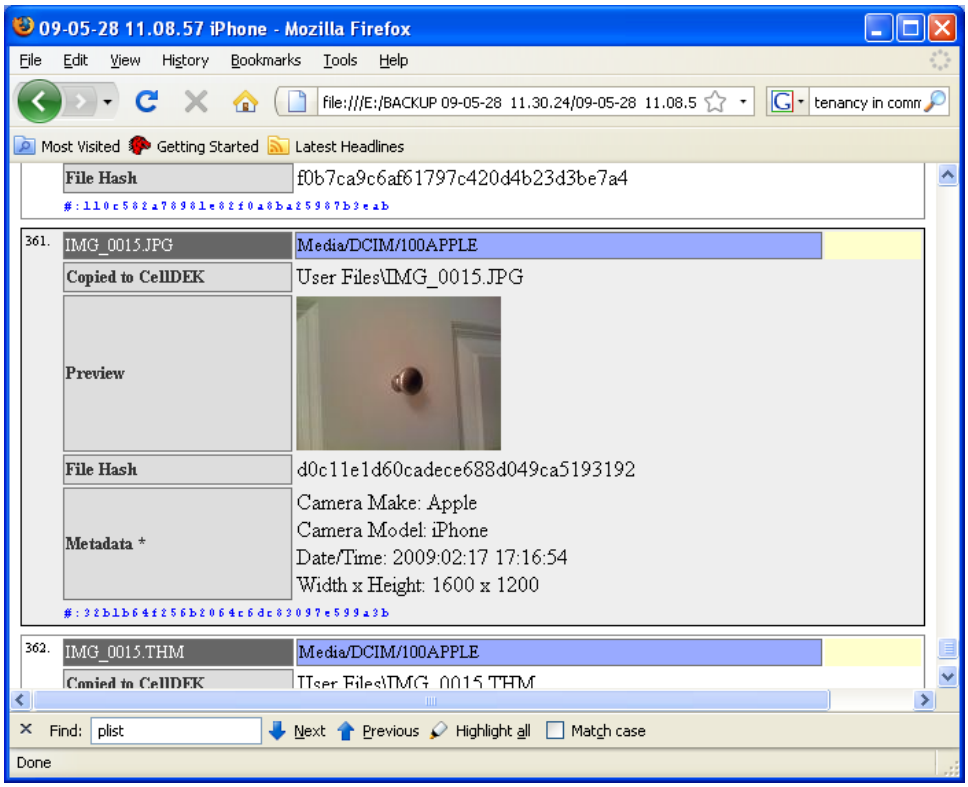
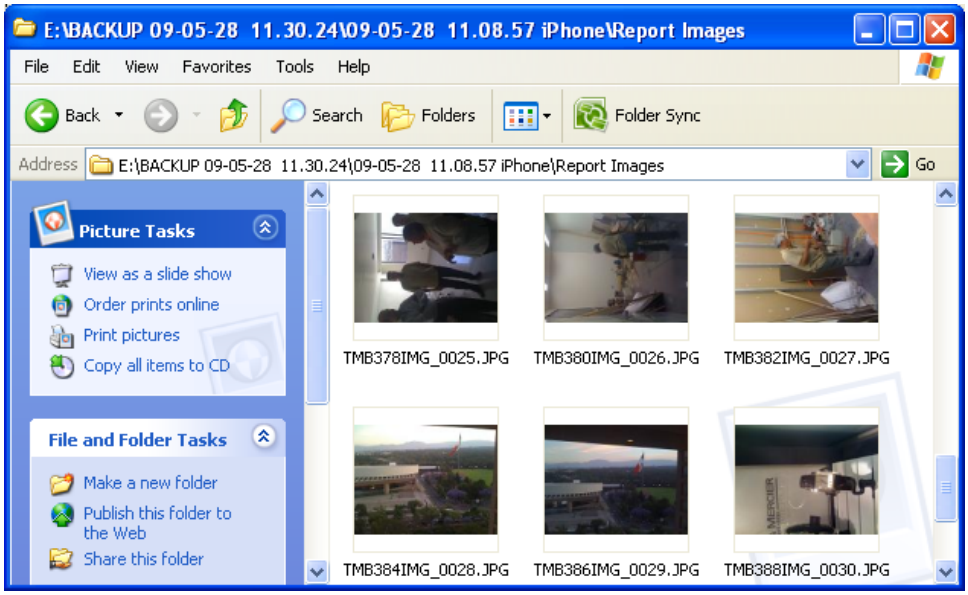
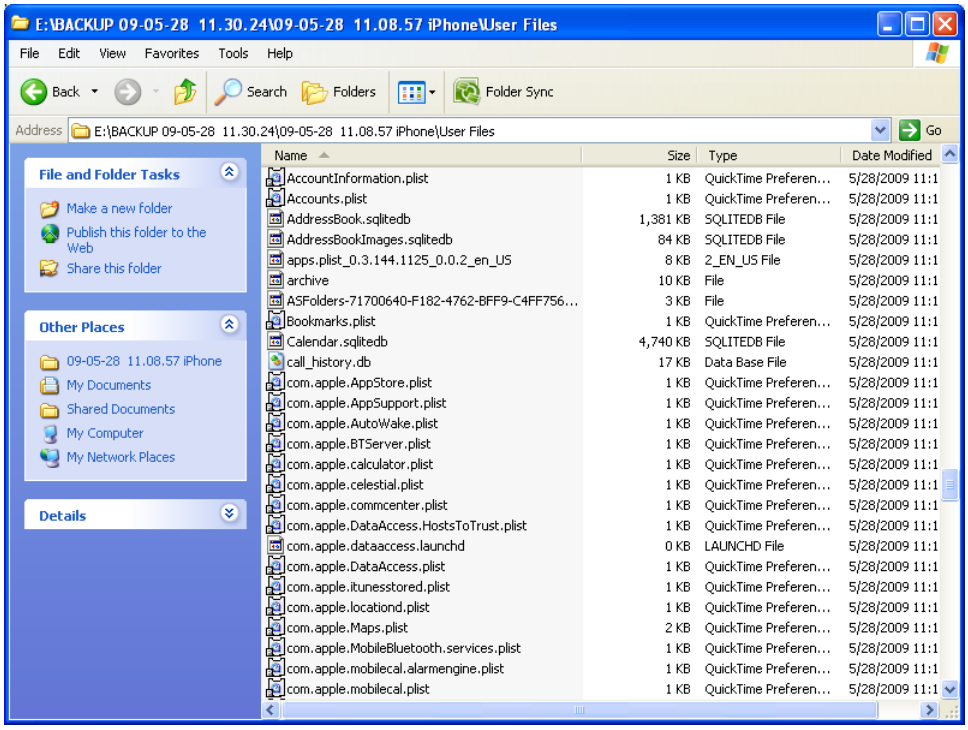


Figure 9.12. Report Images Folder



The .html report lists various .plist and .db files but does not display their contents. To do this the user must use a viewer that can read these files and view them separately. These files can be found in the User Files folder that was transferred from the CellDEK.

Figure 9.13. User Files



5. Matrix of Results

The following are the results from the CellDEK extraction.

Table 9.1. CellDEK Matrix of Results

Scenario	CellDEK Results	Ranking	Results

Call Logs	100 (broken down into Dialed and Missed calls)	3	Meet
SMS	120 (deleted not recovered)	3	Meet
Contacts	1511	3	Meet
Email	0	0	Below
Calendar	3188	3	Meet
Notes	1 (deleted note not recovered)	3	Meet
Pictures	27 (deleted and Syncd photos not retrieved	3	Meet
Songs	audio and video files not recovered	0	Below
Web History	Recent pages and recent search history listed.	3	Meet
Bookmarks	Not all were found.	2	Below
Cookies	Only BOA cookies found	2	Below
App Info	App names listed, only info found	2	Below
Google Maps	Yes, history of recent routes	3	Meet
Voicemail	0	0	Below
Password	None found	0	Below
Plists/XML	Many retrieved	3	Meet
Phone Info	Yes	3	Meet
Video	0	0	Below
Podcasts	0	0	Below
Speed Dials	Found programmed speed dial in plist	3	Meet
VPN	List of routers and wifi networks found	2	Below
Bluetooth	Listed as enabled	2	Below
GPS	Additional coordinates found in .plists	2	Below
File Hashes	Yes	3	Meet
You Tube	Contains info about recently viewed videos.	3	Meet
HTML	Recent weather page info found plus additional Facebook info	3	Meet

6. Conclusions

The self contained nature of the CellDEK gives it portability. This portability is probably the biggest advantage of this extraction device. Since the software came preloaded installation was a breeze, but the device did require us to download an update (iTunes) to a separate computer and transfer it to the CellDEK.

The extraction process was quick and the report was immediately viewable on the built-in screen. Additionally proprietary software was not needed to view the report on a separate computer once transferred.

The following ranking establishes CellDEK's overall rating of 2.6 on the four criteria established at the beginning of this white paper.

Table 9.2. CellDEK Rankings

Area	Weight	Rank
Installation	0.1	4.0
Acquisition	0.2	3.0
Reporting	0.3	3.0
Accuracy	0.4	2.1
TOTAL		2.6

Chapter 10. Report Conclusions

The Apple iPhone is poised to significantly increase market share in 2009 and into the foreseeable future. With over 13 million phones in use today, it has already become a device that forensic experts, or certainly mobile forensics experts, should understand and be capable of analyzing.

The forensic software market is scrambling to add support for the iPhone to their forensic suite or to build a new tool with the specialized purpose of providing iPhone forensics. Each vendor has a unique place in the market. Some are established mobile phone forensic tools which are able to add iPhone support. Others are brand new and built from the ground up while some companies have built on the Apple expertise to provide a solution. And one technique seeks to provide the import bit-by-bit copy of the original user media.

With such scrambling and positioning, it is inevitable that some tools will not perform consistent with the standards forensic expert require. However, every vendor I spoke with is actively developing and improving on their product so expect frequent updates.

It is important that you test and have confidence in the forensic tool you choose. Between the frequent firmware updates from Apple, availability of Application downloads from the App Store, personal user's habits and the state of the iPhones host computer, it is difficult to find a common state from which a vendor can base their tool. While the vendors must be diligent, as forensic analysts we too have a responsibility to understand, test and explain the tool we select.

Chapter 11. About this white paper

1. About the Authors

Andrew Hoog, Chief Investigative Officer of viaForensics, is a recognized computer scientist and forensic analyst and former chief information officer of a $750 million multinational corporation. He has led investigations, contributed to policy development and lectured at corporations, attorneys' associations and law enforcement agencies about the computer forensic discipline. He maintains a computer forensics and E-discovery glossary, authors computer/ mobile forensic how-to guides and is now writing a book about Android forensics. He is the original author of this ground breaking white paper on iPhone Forensics that has gained recognition throughout the industry.

Kyle Gaffaney is a third year law student at Loyola University of Chicago School of Law. Kyle also has degrees in Accounting and Management Information Systems from the University of Minnesota Carlson School of Management. Prior to law school Kyle served as a staff accountant at a financial management firm.

2. About viaForensics

viaForensics is an innovative computer/mobile forensic and e-discovery company providing expert consulting services to corporations, law firms, law enforcement and government agencies.

Beyond servicing our clients immediate needs, the company focuses on ground breaking research in areas such as mobile forensics, SQLite forensics, data visualization and general education on forensics by regularly posting HOWTOs, glossary terms and the results of our research, accessible at viaforensics.com.

3. Why Outsource?

One key strategy to minimizing this risk is to implement computer forensic techniques. But the question is, why outsource? Often the initial response is that internal IT resources can perform these services in addition to their normal day to day tasks. But the reality is that there are significant burdens including:

- Impartiality: Your case must be credible, unbiased and withstand legal scrutiny; internal investigations present major obstacles in each of these areas.
- Expertise: viaForensics is a qualified expert in the Federal Courts. Expert status is a product of extensive training and a wide range of experience, often a challenge in a single corporate environment.
- Cost: Forensic hardware, software and training are singular in purpose and require major capital investments and recurring expenses.