available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/diin

Digital Investigation

Introduction to Windows Mobile Forensics

Eoghan Casey [a,*], *Michael Bann* [b], *John Doyle* [b]

[a] *cmdLabs, Suite C301, Baltimore, MD 21218, USA*
[b] *Johns Hopkins University, Information Security Institute, Baltimore, MD 21218, USA*

Keywords:
Windows Mobile Forensics
Windows CE forensics, Mobile device forensics
Cell phone forensics, CEDB database
Transaction-safe FAT, TFAT, Mobile spyware
MobileSpy

ABSTRACT

Windows Mobile devices are becoming more widely used and can be a valuable source of evidence in a variety of investigations. These portable devices can contain details about an individual's communications, contacts, calendar, online activities, and whereabouts at specific times. Although forensic analysts can apply their knowledge of other Microsoft operating systems to Windows Mobile devices, there are sufficient differences that require specialized knowledge and tools to locate and interpret digital evidence on these systems. This paper provides an overview of Windows Mobile Forensics, describing various methods of acquiring and examining data on Windows Mobile devices. The locations and data formats of useful information on these systems are described, including text messages, multimedia, e-mail, Web browsing artifacts, and Registry entries. This paper concludes with an illustrative scenario involving MobileSpy monitoring software.

1. Introduction

Windows Mobile devices present a substantial opportunity and challenge for forensic practitioners. These devices are essentially computers that people carry in their pockets, which contain substantial amounts of information that can be useful from a forensic perspective, including communications, multimedia, and location information. These devices can be sources of evidence in a wide range of crimes, including homicide, fraud, and data theft. The personal nature of the information on these devices can provide digital investigators with valuable insights into the *modus operandi* of suspects and activities of victims. In addition, investigators in criminal, corporate, and military contexts must be able to detect the presence of programs that permit remote monitoring of Windows Mobile devices. New acquisition methods have become available that give forensic practitioners access to more information on these devices, including deleted data.

At the same time, Windows Mobile devices are relatively new and the data formats are unfamiliar to most forensic practitioners, such as volume files and embedded databases. Tools for interpreting and analyzing data on Windows Mobile devices are struggling to keep pace with advancements in the technology. Forensic analysts need to understand the underlying technologies and formats that exist, prior to using a variety of tools to extract useful information.

This paper covers various methods for acquiring and analyzing data on Windows Mobile devices using both commercial and open source tools. Details regarding the test devices used for this paper are provided in Table 1.

To enable forensic practitioners to obtain useful evidence from Windows Mobile devices this paper begins with an overview of Windows Mobile, covering current effective practices for acquiring data from these systems. The remainder of this paper describes where useful information is stored and how to examine these important data sources. This paper concludes with a scenario involving MobileSpy monitoring software. Common hurdles are discussed to help practitioners navigate issues such as data translation errors.

[*] *Corresponding author.*
E-mail address: ecasey@cmdlabs.com (E. Casey).

1742-2876/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.diin.2010.01.004

Table 1 – Summary of test device characteristics.

Manufacturer/model	OS version	OS build	Radio version
HTC S620 (Dash)	Windows Mobile 6 Standard, 5.2.1236	17741.0.2.1	4.1.13.61_03.21.90
Motorola Q	Windows Mobile 5.0, 5.1.195	14960.2.4.0	Q2-BP_C_06.0B.11P, Q2 Portable
Samsung i607 (Blackjack)	Windows Mobile 5.0 with Messaging and Security Feature Pack, 5.1.342	15100.3.0.2	

This paper deals with Windows Mobile devices that are not password protected. Be aware that a password protected device may be configured to wipe all user-created data after a set number of failed logon attempts. More advanced acquisition methods like chip extraction may be able to bypass password protection and, due to the use of Flash memory and wear leveling on these devices, can enable forensic analysts to access more deleted data than the methods detailed in this paper (van der Knijff, 2009; Klaver, 2010).

2. Windows Mobile overview

Many of the lessons learned from forensic processing of other Microsoft Windows operating systems can be applied to Windows Mobile, including understanding of FAT file systems and `index.dat` files. As with a desktop or laptop computer, Windows Mobile devices retain substantial information about user activities that can be relevant in a digital investigation like Web browsing, user-created files, and Registry entries. The names of recently connected computers and WiFi access points can also be retained on Windows Mobile devices, which can be useful in some digital investigations. However, there are sufficient differences between Windows Mobile systems and other Windows operating systems to require specialized knowledge and tools to locate and interpret digital evidence.

Windows Mobile uses a variation of the FAT file system called the Transaction-safe FAT (TFAT) file system, which has some recovery features in the event of a sudden device shutdown. As shown in Fig. 1, the file system hierarchy on these devices has similarities with other Microsoft operating systems, which should be familiar to anyone who has performed a forensic examination of Windows computer systems.

The majority of user-created files, including digital photographs and videos taken with the device camera, are stored in the "My Documents" folder. Conversely, Windows Mobile devices retain remnants of user activities in a variety of locations. These usage artifacts include `index.dat` files associated with the use of Internet Explorer and embedded database files ending with a ".vol" extension shown in the right-hand pane of Fig. 1 above. In addition, the Registry on Windows Mobile devices can store information about users and their activities, as demonstrated in the "Examining Registry hives" section of this paper.

2.1. *Locations of usage artifacts on Windows Mobile devices*

Although certain files like `cemail.vol` can be found on all Windows Mobile devices, the location of usage artifacts on

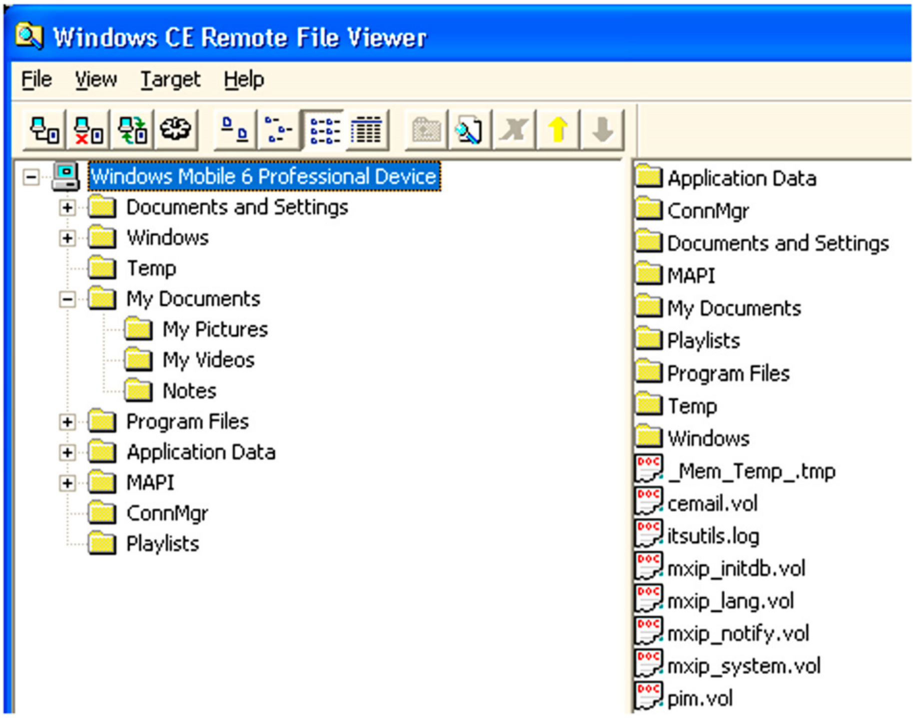


Fig. 1 – File system hierarchy on a Samsung i607 (Blackjack).

Table 2 – Potentially useful sources of evidence on Windows Mobile devices.

File	Description
\cemail.vol	An embedded database that stores information relating to communications, including text messages and portions of e-mails, not including file attachments.
\pim.vol	An embedded database that includes call logs (clog.db), address book information, calendar items, speed dial details (speed.db), and to do tasks.
\ReplStorVol	A file replication database used to synchronize items on the device with data in another location (Microsoft, 2008a).
\My Documents\My Pictures	A repository of photographs taken or downloaded by the user. This is the default download location for pictures.
\My Documents\UAContents	A folder with artifacts of user activities, including portions of MMS in ".dat" files and an MMS log file.
\Documents and Settings\default\user.hv	The User Registry hive.
\Documents and Settings\default.hv OR system.hv[a]	The System Registry hive.
\Windows\Messaging	A repository of viewed SMS and e-mail messages, stored in ".mpb" files.
\Windows\Messaging\Attachments	A repository of downloaded e-mail attachments in ".att" files.
\Windows\Profiles\guest	Contains Internet Explorer history, as well as cache and cookie files, including `index.dat` files.
\Windows\Favorites	Internet Explorer bookmarks.
Windows\eT9Cdb.Cdb and eT9Rudb.Rdb	Custom user T9 dictionary files.

[a] The location of the system Registry hive may vary. The Registry value under HKEY_LOCAL_MACHINE\init\BootVars\SystemHive contains the full path of the system hive.

different mobile device models can vary. Table 2 provides an overview of potentially useful sources of evidence on the Samsung i607 (Blackjack), HTC S62 (Dash), and Motorola Q devices. Many of these locations will also be found on other kinds of Windows Mobile devices. Additional files of potential interest may be found in other locations like the "\Temp" folder.

Further details about the areas listed in Table 2 are provided later in this paper with examples of how the information can be useful from a forensic perspective.

3. Forensic processing of Windows Mobile devices

From a forensic perspective, it can be challenging to acquire and examine the information on Windows Mobile devices. Some of the files that contain usage artifacts on Windows Mobile devices are locked by the operating system, making it more difficult to obtain a forensic duplicate of their contents. For instance, certain forensic acquisition tools that rely on Windows Mobile APIs cannot copy the contents of files that are locked by the operating system like `cemail.vol`, `pim.vol`, and Registry hives. As a result of this and other restrictions on Windows Mobile devices, most widely available acquisition methods do not obtain all data stored on these devices; some tools obtain much more data than others.

Furthermore, many forensic tools have difficulty interpreting data acquired from Windows Mobile devices. These interpretation difficulties can result in misrepresentation of the TFAT file system or exclusion of usable information from important files, requiring forensic practitioners to decipher proprietary file formats. Discrepancies can also exist in the interpretation of data attributes between tools, which is discussed within the "Tools and interpretation" section of this paper.

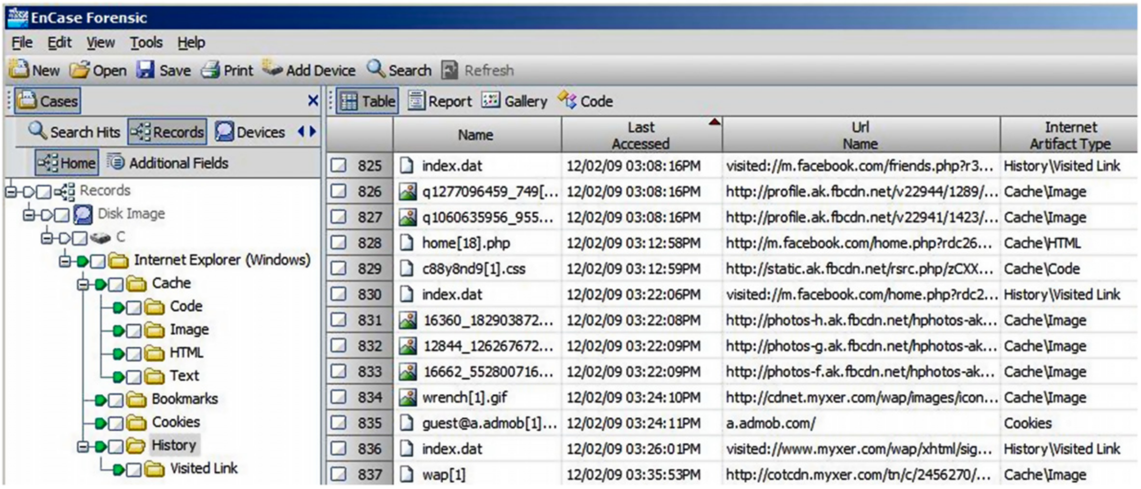
At the same time, certain aspects of Windows Mobile devices will be familiar to forensic analysts and can be examined using the same tools used for examining other Windows operating systems. For instance, the file system and certain files on Windows Mobile devices can be interpreted and examined using file system forensic tools like EnCase, X-Ways, and FTK. Fig. 2 shows EnCase being used to view records in `index.dat` files and associated cached Web content on a Samsung i607 (Blackjack) device.

3.1. *Forensic acquisition*

The forensic acquisition tools that are available to most forensic analysts do not have direct access to flash memory on Windows Mobile devices and are limited to acquiring data through a hardware abstraction layer. As a result, the tools described in this paper can acquire data storage areas on Windows Mobile devices, but do not acquire a complete physical duplicate of flash memory. This limitation prevents access to deleted information that exists in flash memory that no longer resides in an active data storage area.

Furthermore, in order to acquire storage areas on Windows Mobile devices, current tools run a customized software agent on the target device. Some Windows Mobile devices do not permit unsigned programs to run and must be reconfigured to allow the forensic acquisition tools to work properly.

Two tools for acquiring data from Windows Mobile devices are XACT from Microsystemation (http://www.msab.com) and `itsutils` (http://wiki.xda-developers.com/index.php?pagename=XdaUtils). Both of these tools require ActiveSync to be installed on the acquisition system before data can be acquired from a connected Windows Mobile device.


EnCase Forensic
File Edit View Tools Help
New Open Save Print Add Device Search Refresh
Cases
Search Hits Records Devices
Home Additional Fields
Records
Disk Image
C
Internet Explorer (Windows)
Cache
Code
Image
HTML
Text
Bookmarks
Cookies
History
Visited Link
Table Report Gallery Code

	Name	Last Accessed	Url Name	Internet Artifact Type
825	index.dat	12/02/09 03:08:16PM	visited://m.facebook.com/friends.php?r3...	History\Visited Link
826	q1277096459_749[...	12/02/09 03:08:16PM	http://profile.ak.fbcdn.net/v22944/1289/...	Cache\Image
827	q1060635956_955...	12/02/09 03:08:16PM	http://profile.ak.fbcdn.net/v22941/1423/...	Cache\Image
828	home[18].php	12/02/09 03:12:58PM	http://m.facebook.com/home.php?rdc26...	Cache\HTML
829	c88y8nd9[1].css	12/02/09 03:12:59PM	http://static.ak.fbcdn.net/rsrc.php/zCXX...	Cache\Code
830	index.dat	12/02/09 03:22:06PM	visited://m.facebook.com/home.php?rdc2...	History\Visited Link
831	16360_182903872...	12/02/09 03:22:08PM	http://photos-h.ak.fbcdn.net/hphotos-ak...	Cache\Image
832	12844_126267672...	12/02/09 03:22:09PM	http://photos-g.ak.fbcdn.net/hphotos-ak...	Cache\Image
833	16662_552800716...	12/02/09 03:22:09PM	http://photos-f.ak.fbcdn.net/hphotos-ak...	Cache\Image
834	wrench[1].gif	12/02/09 03:24:10PM	http://cdnet.myxer.com/wap/images/icon...	Cache\Image
835	guest@a.admob[1]...	12/02/09 03:24:11PM	a.admob.com/	Cookies
836	index.dat	12/02/09 03:26:01PM	visited://www.myxer.com/wap/xhtml/sig...	History\Visited Link
837	wap[1]	12/02/09 03:35:53PM	http://cotcdn.myxer.com/tn/c/2456270/...	Cache\Image

Fig. 2 – Remnants of Internet Explorer browsing activities on a Samsung i607 (Blackjack) device viewed using EnCase.

Many mobile devices support removable storage media like micro SD cards that can store larger files like digital photographs, videos, and music. Although mobile forensic tools may be able to acquire logical data from such removable media through the device itself, this process may alter data on the store media and will not give forensic analysts access to deleted data. Therefore, unless the card is password protected or encrypted, it is generally advisable to remove such storage media and create a forensic duplicate them using standard computer forensic methods.

The commercial tool XACT can acquire the primary storage areas, the equivalent of partitions, on certain Windows Mobile devices. Fig. 3 shows XACT being used to acquire four storage areas on a Motorola Q device.

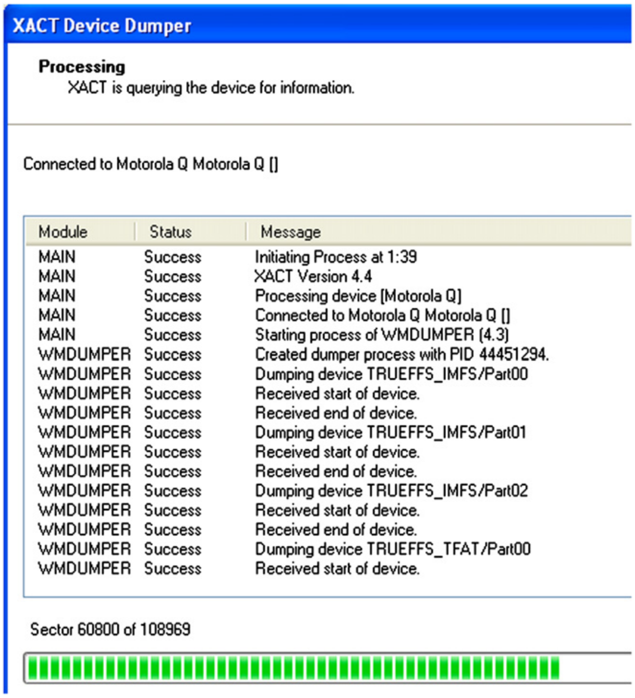
For certain Windows Mobile devices, the XACT software agent can be placed on a memory card and inserted into the device, thus increasing forensic soundness by minimizing the amount of changes caused by running the executable on the subject system.

The `itsutils` package provides an open source option for acquiring storage areas from Windows Mobile devices. Using the `psdread` component of this package with the `-l` option provides a list of storage areas as shown below on a Motorola Q device.

```
D:\itsutils>psdread -l
C: - TOSHIBA MK6021GAS
Drive geometry: 0x1e48 cyls, 240 t/cyl 63 s/t 512 b/s
- 55.89Gbyte
    disknr = 0  fixed disk
D: - SONY DVD+RW DW-P50A
remote disk 1 has 135576 sectors of 512 bytes -
66.20Mbyte
SerialNr: 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
remote disk 2 has 112392 sectors of 512 bytes -
54.88Mbyte
SerialNr: 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

The first two entries in the above list refer to the local hard disk of the acquisition system. The two subsequent entries refer to remote disks on the Windows Mobile device, which are the system and data storage areas, respectively. The following command options acquire the storage area that contains the most relevant data from a forensic perspective (remote disk 2 on this device shown in bold above), starting at 0 and copying 57544704 bytes (112392 sectors × 512 bytes).

```
D:\itsutils>psdread -2 0 57544704 E:\Samsung-
i607.bin
remote disk 2 has 112392 sectors of 512 bytes -
54.88Mbyte
SerialNr: 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```



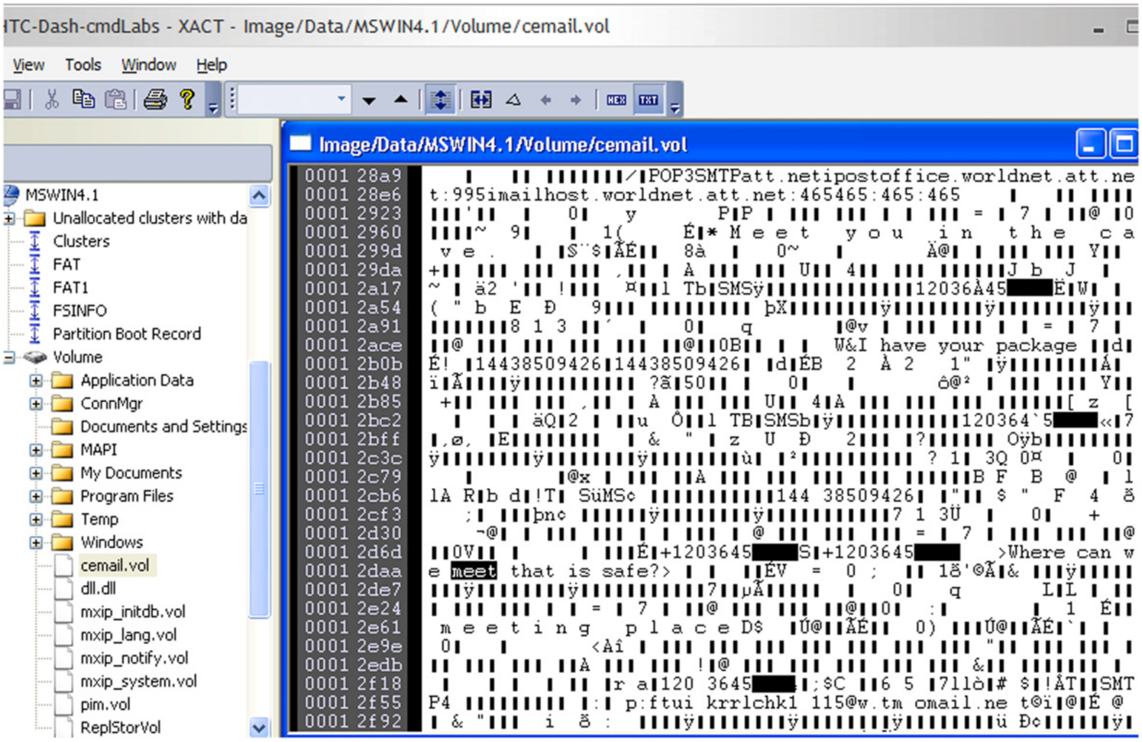
Processing
XACT is querying the device for information.

Connected to Motorola Q Motorola Q []

```
Module      Status    Message
MAIN        Success   Initiating Process at 1:39
MAIN        Success   XACT Version 4.4
MAIN        Success   Processing device [Motorola Q]
MAIN        Success   Connected to Motorola Q Motorola Q []
MAIN        Success   Starting process of WMDUMPER (4.3)
WMDUMPER    Success   Created dumper process with PID 44451294.
WMDUMPER    Success   Dumping device TRUEFFS_IMFS/Part00
WMDUMPER    Success   Received start of device.
WMDUMPER    Success   Received end of device.
WMDUMPER    Success   Dumping device TRUEFFS_IMFS/Part01
WMDUMPER    Success   Received start of device.
WMDUMPER    Success   Received end of device.
WMDUMPER    Success   Dumping device TRUEFFS_IMFS/Part02
WMDUMPER    Success   Received start of device.
WMDUMPER    Success   Received end of device.
WMDUMPER    Success   Dumping device TRUEFFS_TFAT/Part00
WMDUMPER    Success   Received start of device.
```

Sector 60800 of 108969

Fig. 3 – XACT acquisition screenshot of Motorola Q.



```
0001 28a9   ▮   ▮▮ ▮▮▮▮▮▮▮/▮POP3SMTPatt.netipostoffice.worldnet.att.ne
0001 28e6 t:995imailhost.worldnet.att.net:465465:465:465   ▮  ▮▮ ▮▮▮▮
0001 2923 ▮▮▮'▮▮  ▮   0▮   y      P▮P ▮ ▮▮▮ ▮▮▮ ▮ ▮ ▮▮▮ = ▮ 7 ▮ ▮▮@ ▮0
0001 2960 ▮▮▮▮~  9▮   ▮  1(     É▮* M e e t   y o u   i n   t h e   c a
0001 299d  v e .   ▮ ▮S¨$▮ÃÉ▮▮  8à  ▮   0~   ▮       Ä@▮ ▮ ▮▮▮ ▮▮▮ Y▮▮
0001 29da +▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ,▮▮ ▮ À ▮▮▮ ▮▮▮ U▮▮ 4▮▮ ▮▮▮ ▮▮▮▮▮▮J b  J  ▮
0001 2a17 ~ ▮ ä2 '▮▮ !▮▮▮  ¤▮▮l Tb▮SMSÿ▮▮▮▮▮▮▮▮▮▮▮▮▮12036À45[redacted]Ë▮W▮ ▮
0001 2a54 ( " b  E  Đ   9▮▮▮ ▮▮▮▮▮▮▮▮▮ þX▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮ÿ▮▮▮
0001 2a91 ▮▮▮▮▮▮▮8 1 3 ▮▮´  ▮    0▮    q        ▮@v ▮ ▮▮▮ ▮▮▮ ▮ ▮ = ▮ 7 ▮
0001 2ace ▮▮@ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮@▮▮0B▮▮ ▮ ▮  W&I have your package ▮▮d▮
0001 2b0b É! ▮14438509426▮14438509426▮ ▮d▮ÉB  2  À 2   1" ▮ÿ▮▮▮▮▮▮▮▮▮Á▮
0001 2b48 ï▮Ã▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮▮ ?ã▮50▮▮ ▮    0▮    ▮       ô@² ▮ ▮▮▮ ▮▮▮ Y▮▮
0001 2b85  +▮▮ ▮▮▮ ▮▮▮ ,▮▮ ▮ À ▮▮▮ ▮▮▮ U▮▮ 4▮À ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮▮▮▮[ z  [
0001 2bc2  ▮  ▮ ▮ äQ▮2 ▮ ▮▮u  Ô▮▮l TB▮SMSb▮ÿ▮▮▮▮▮▮▮▮▮▮▮▮▮120364`5[redacted]«▮7
0001 2bff ▮,ø, ▮E▮▮▮▮▮▮▮▮  ▮ &  " ▮ z  U  Đ   2▮▮▮ ▮?▮▮▮▮▮▮ Oÿb▮▮▮▮▮▮▮▮
0001 2c3c ÿ▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮ù▮ ▮²▮▮▮▮▮▮▮▮▮▮ ? 1▮ 3Q 0¤  ▮   0▮
0001 2c79    ▮      ▮@x ▮ ▮▮▮ ▮▮À ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮▮▮▮B F  B  @  ▮ l
0001 2cb6 lA R▮b d▮!T▮ SüMS¢ ▮▮▮▮▮▮▮▮▮▮▮▮144 38509426▮ ▮"▮▮ $ "  F  4  ŏ
0001 2cf3   ;▮ ▮▮▮þn¢ ▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮▮▮▮7 1 3Ü  ▮   0▮   +
0001 2d30    ¬@▮ ▮ ▮▮▮ ▮▮▮ ▮ ▮ ▮▮▮ ▮ @ ▮▮▮ ▮▮▮ ▮▮▮ = ▮ 7 ▮ ▮▮▮ ▮▮▮ ▮▮@
0001 2d6d ▮▮0V▮▮ ▮     ▮ ▮▮▮É▮+1203645[redacted]S▮+1203645[redacted]   >Where can w
0001 2daa e meet that is safe?> ▮ ▮  ▮▮ÉV  =  0 ;  ▮▮ 1ŏ'©Ã▮& ▮▮▮ÿ▮▮▮▮
0001 2de7 ▮▮▮ÿ▮▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮▮▮▮▮▮7▮▮µÃ▮▮▮▮▮  ▮    0▮   q      L▮L ▮ ▮▮
0001 2e24 ▮ ▮▮▮ ▮▮▮ ▮ ▮ = ▮ 7 ▮ ▮▮@ ▮▮▮ ▮▮▮ ▮▮@▮▮0▮   :▮       ▮  1  É▮▮
0001 2e61  m e e t i n g   p l a c e D$  ▮Ú@▮▮ÃÉ▮▮  0) ▮▮▮Ú@▮▮ÃÉ▮`▮ ▮
0001 2e9e  0▮   ▮       <Aî ▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ "▮▮ ▮▮▮ ▮▮▮ ▮
0001 2edb ▮▮ ▮▮▮ ▮▮▮ ▮▮À ▮▮▮ ▮▮▮ !▮@ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ ▮▮▮ &▮▮ ▮▮▮▮▮▮▮ ▮
0001 2f18  ▮   ▮ ▮  ▮ ▮▮ ▮r a▮120 3645[redacted]▮;$C ▮▮6 5 ▮711ò▮# $▮!ÅT▮▮SMT
0001 2f55 P4 ▮▮▮▮▮▮▮▮▮ ▮:▮ p:ftui krrlchk1 115@w.tm omail.ne t©ï▮@▮É @
0001 2f92 ▮ & "▮▮▮  i  ŏ :  ▮▮▮▮ÿ▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮▮ü Đ¢▮▮▮▮▮▮ÿ▮
```

Fig. 4 – Windows Mobile file system viewed using XACT with missing folders.

```
CopySDCardToFile(remote, 2, 0x0, 0x36e1000, C:\Samsung-i607.bin)
```

When psdread does not work with a given Windows Mobile device, it may still be possible to acquire data using pdocread. The pdocread utility only acquires individual partitions and sometimes must rely on the Windows disk on a chip API, which may restrict the amount of data that can be acquired.

3.2. *Deleted file recovery*

Although forensic tools can recover deleted file names from the TFAT volume of Windows Mobile devices, forensic analysts may encounter barriers to recovery of files. For instance, failure to correctly reconstruct the TFAT file system on a Windows Mobile device can result in missing files and folders. Fig. 4 shows the file system acquired from an HTC S620 (Dash) acquired using psdread, missing subfolders under "Documents and Settings".

In some cases, important files like pim.vol are missing from the file system view. The incomplete reconstruction of file systems is not limited to mobile devices, and has occurred in file system forensic tools (Casey, 2005). The difficulty of reconstructing file systems on mobile devices is exacerbated by the presence of repeated "DONT DEL" directory entries, and the rapidly changing nature of mobile devices. These kinds of discrepancies emphasize the importance of validating what mobile device forensic tools present, as discussed in the "Tool Validation" section of this paper.

Another barrier to recovering deleted files is that some Windows Mobile devices appear to overwrite the contents of deleted files with a repeated 0xFF pattern. Keep in mind that the original contents of such deleted files may be salvageable using advanced forensic techniques that provide access to full contents of physical Flash memory.

File carving techniques will have limited success if deleted file content has been overwritten, unless more advanced techniques are used to obtain a full physical memory dump. Although deleted files may be difficult to recover, copies may exist elsewhere on the device as attachments to MMS messages or e-mail messages as demonstrated later in this paper. Keep in mind that keyword searching may be the most effective approach to finding data fragments of interest in some cases.

3.3. *Examining embedded databases*

Windows Mobile devices store some significant information in volume files that encapsulate multiple embedded databases that include details about communications, contacts, and calls (Microsoft, 2005, 2010). For example, pim.vol contains embedded database information such as call history and contact information through the clog.db and contacts databases. Although the format is not formally documented, many aspects of the pim.vol and cemail.vol files have been explored by application developers. The relationship between the databases within cemail.vol is depicted in Fig. 5. Each database contains multiple records, each with its own OID (object identifier) that provides the fastest mechanism for finding a specific record in the cemail.vol file. Each record contains fields (a.k.a. properties) that store the actual data.

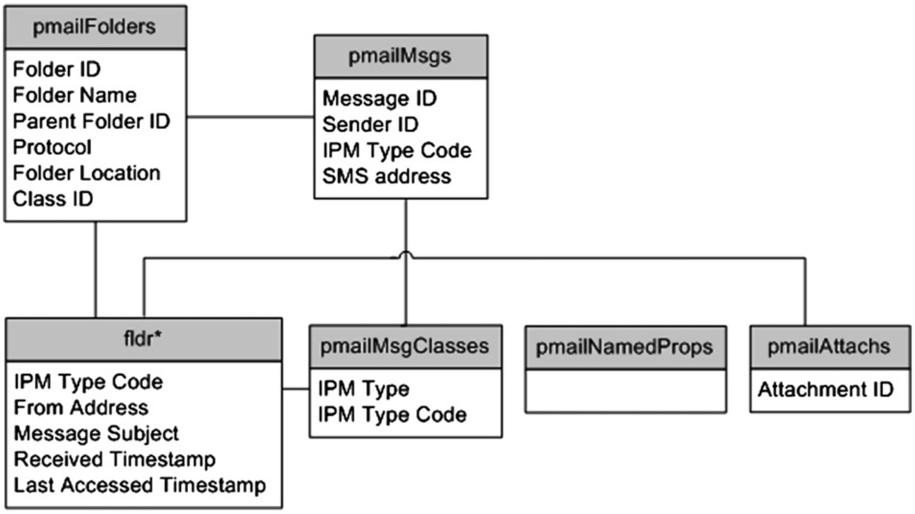


Fig. 5 – Overview of `cemail.vol` file.

The `cemail.vol` file stores details about each message and contains references to associated content in other files on the device like voluminous message bodies and attachments. Generally, components of messages are stored in several locations: "pMail*" and "fldr*" databases in `cemail.vol`, and ".mpb" and ".att" files on the device. The most useful embedded databases within `cemail.vol` are described here:

pmailFolders: This database defines the message folder hierarchy (e.g. Inbox, Outbox, Drafts, Deleted Items, etc.) for each address that the Windows Mobile device is configured with. For each message folder, there is a record in "pmailFolder" that shows the "fldr*" database with associated message details.

pmailMsgs: This database contains summary details about messages on the device, including message ID, message type, and message address information. Values in this database indicate which "fldr*" database each message is associated with based on the folder ID, typically in the format of "fldr" + "folder ID" (e.g. fldr31000026). Table 3 below describes some useful property identifiers within each record.

Table 3 – Property identifiers for useful items within the "pmailMsgs" database.

Property ID	Description
0x800C	Contains sender identification information, such as a phone number in the case of an SMS message.
0x8001	Contains the Interpersonal Message (IPM) type code, which indicates the type of message sent (e.g. SMS, MMS, e-mail). The lookup table for IPM type code resides within the "pmailMsgClasses" database.
0x0E09	Contains the Folder ID in decimal form. This must be converted into its hexadecimal equivalent to determine the containing "fldr*" database.

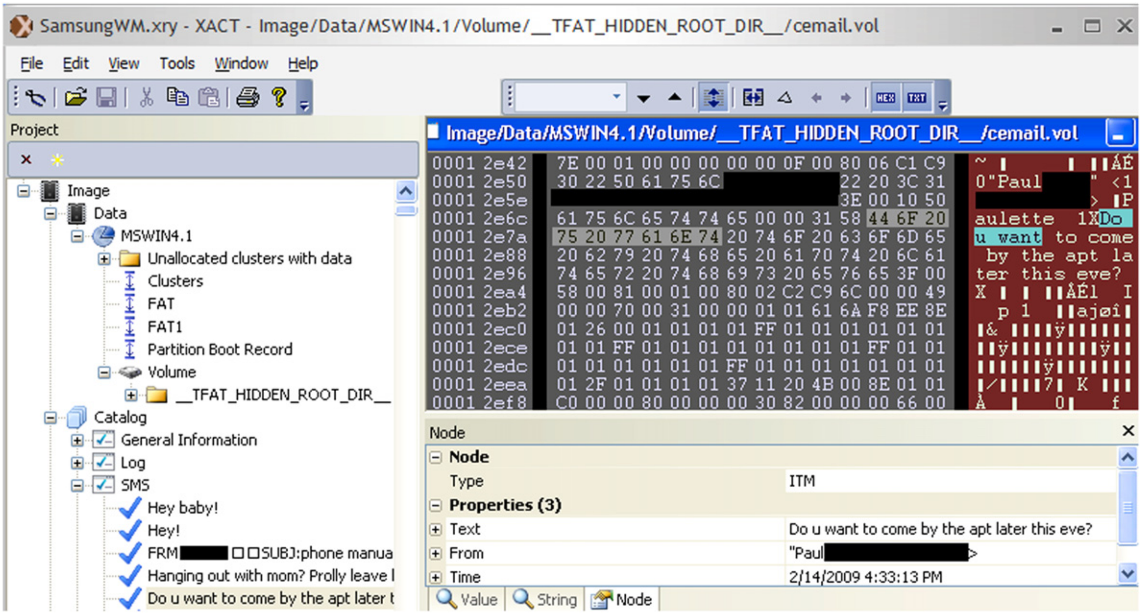
pmailMsgClasses: This database provides a lookup table of IPM types used in the "pmailMsgs" database and "fldr*" databases. For instance, the IPM associations from "pmailMsgClasses" on an HTC S620 (Dash) are listed here with the content type on the left and the associated identifier on the right:

IPM.MMS	822083597
IPM.Note	822083598
IPM.SI	822083600
IPM.SL	822083601
IPM.SMStext	822083599
IPM.SMStext.SIM	855638066
REPORT.IPM.Note.DR	822083603
REPORT.IPM.Note.IPNNRN	822083606
REPORT.IPM.Note.IPNRN	822083605
REPORT.IPM.Note.NDR	822083604
REPORT.IPM.Note.Status	822083602

pmailNamedProps: This database contains a lookup table of object property names that reside within the device (e.g. SMS:SMSCAddress, Meeting:Reminder). Its structure is similar

Table 4 – Property identifiers for useful items within "fldr*" databases.

Property ID	Description
0x8005	OID used as a lookup value.
0x0C1F	From address (contact name unresolved)
0x0C1A	From address (contact name resolved)
0x003D	Denotes the message prefix, either "Re: ", "Fw: ", or "" denoting reply, forward, and null, respectively.
0x0037	Message subject or, when applicable, the message body if it is small enough.
0x0E06	Message received timestamp.
0x3008	Message last modified timestamp.
0x001A	Lookup field, which links this database to the "pmailMsgClasses" database.



SamsungWM.xry - XACT - Image/Data/MSWIN4.1/Volume/__TFAT_HIDDEN_ROOT_DIR__/cemail.vol

```
Image
  Data
    MSWIN4.1
      Unallocated clusters with data
      Clusters
      FAT
      FAT1
      Partition Boot Record
      Volume
        __TFAT_HIDDEN_ROOT_DIR__
  Catalog
    General Information
    Log
    SMS
      Hey baby!
      Hey!
      FRM[redacted]□□SUBJ:phone manua
      Hanging out with mom? Prolly leave l
      Do u want to come by the apt later t
```

Image/Data/MSWIN4.1/Volume/__TFAT_HIDDEN_ROOT_DIR__/cemail.vol

```
0001 2e42  7E 00 01 00 00 00 00 00 0F 00 80 06 C1 C9  ~ ▮        ▮ ▮▮ÁÉ
0001 2e50  30 22 50 61 75 6C [redacted]  22 20 3C 31  0"Paul[redacted]" <1
0001 2e5e                                3E 00 10 50  > ▮P
0001 2e6c  61 75 6C 65 74 74 65 00 00 31 58 44 6F 20  aulette  1XDo 
0001 2e7a  75 20 77 61 6E 74 20 74 6F 20 63 6F 6D 65  u want to come
0001 2e88  20 62 79 20 74 68 65 20 61 70 74 20 6C 61   by the apt la
0001 2e96  74 65 72 20 74 68 69 73 20 65 76 65 3F 00  ter this eve? 
0001 2ea4  58 00 81 00 01 00 80 02 C2 C9 6C 00 00 49  X ▮ ▮ ▮▮ÂÉl  I
0001 2eb2  00 00 70 00 31 00 00 01 01 61 6A F8 EE 8E    p 1  ▮▮ajøî▮
0001 2ec0  01 26 00 01 01 01 01 FF 01 01 01 01 01 01  ▮& ▮▮▮▮ÿ▮▮▮▮▮▮
0001 2ece  01 01 FF 01 01 01 01 01 01 01 01 FF 01 01  ▮▮ÿ▮▮▮▮▮▮▮▮ÿ▮▮
0001 2edc  01 01 01 01 01 01 FF 01 01 01 01 01 01 01  ▮▮▮▮▮▮ÿ▮▮▮▮▮▮▮
0001 2eea  01 2F 01 01 01 01 37 11 20 4B 00 8E 01 01  ▮/▮▮▮▮7▮ K ▮▮▮
0001 2ef8  C0 00 00 80 00 00 00 30 82 00 00 00 66 00  À  ▮   0▮   f
```

Node

Node	
Type	ITM
Properties (3)	
Text	Do u want to come by the apt later this eve?
From	"Paul[redacted]>
Time	2/14/2009 4:33:13 PM

Fig. 6 – XACT showing data in `cemail.vol` file.

to the "pmailMsgClasses" database but uses a colon for demarcation within values instead of a period.

fldr*: These databases contain a wealth of information about messages that are on the device, including the IPM type, the subject, the sender's address, and when the message was received and last modified. When the body of the message is small enough, the full contents are stored within the embedded database. Specific properties that may be stored in records of an "fldr*" database are listed in Table 4 with their associated property identifiers. Any property may or may not be present depending on the record type.

The "fldr*" databases take a static approach to storing data, such that when a contact's name is deleted from the device's contacts, prior messages retain the contact's name. Subsequent messages will not contain the contact's name, and both from addresses listed in Table 4 will contain the same value. This can be of particular interest to investigators if a user deletes a contact from their address book in an attempt to conceal a personal relationship.

3.4. *Tools and interpretation*

Forensic tools have been developed to interpret some information in the `cemail.vol` file. For example, Fig. 6 shows data from the `cemail.vol` file on a Samsung i607 (Blackjack) device in both interpreted and raw form using XACT. The Catalog list on the bottom left displays recoverable items, including SMS messages. Details of the selected text message are displayed in the Node pane on the bottom right. On the top right, the same information in the `cemail.vol` file is shown in both hexadecimal and ASCII formats.

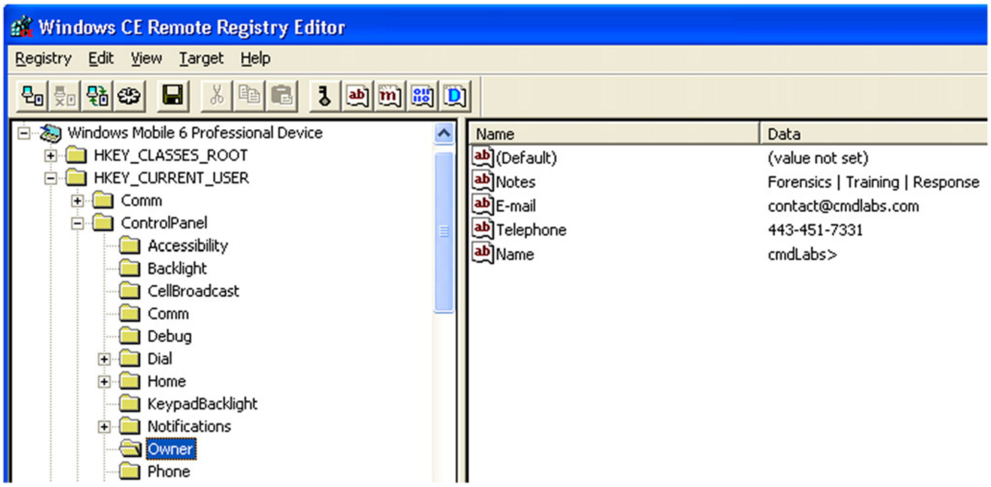
Fortunately, much of the text in `cemail.vol` is ASCII, including SMS text. Since deleted records are not purged from the `cemail.vol` file immediately, it is advisable to examine `cemail.vol` files in a hexviewer to look for text associated with deleted items that is not accessible using the aforementioned methods.

Through the course of a digital investigation, it is imperative that the data being interpreted is being correctly rendered by the forensic analyst's extraction tools. One approach to verify that important values are being interpreted correctly is to view them in hexadecimal form, provided the forensic analyst understands the format of the data. Another approach to detect interpretation errors is to compare the information with another tool or in an emulator.

One approach to view a `cemail.vol` file in its native environment is to extract the file from the subject system, save it into a folder on the examination computer, and then configure a Windows Emulator to treat that folder as a virtual Storage Card (Casey, 2009). In this way, the emulator can be used to open the evidential `cemail.vol` file using a tool like itsutils or Pocket dbExplorer. Another approach is to extract the `cemail.vol` file from the subject system and load it into the emulator, overwriting the generic `cemail.vol` file. A barrier to this method is that the default `cemail.vol` file cannot be easily overwritten because it is locked by the operating system. A possible work around for this issue is available at the XDA developer forum (XDA, 2006).

Once a `cemail.vol` file is mounted in the emulator, another component of the `itsutils` packaged called pdblist can be used to parse the contents of this embedded database (Casey, 2009). The following is output from this command for the "fldr31000028" database.

```
T:\itsutils>pdblist -d fldr31000028
330007ec (332 13 8)
    8005 T13 L0000 F0000 UI4 1006634986
    8011 T13 L0000 F0000 UI4 74
    001a T13 L0000 F0000 UI4 822083597
```



Name	Data
(Default)	(value not set)
Notes	Forensics | Training | Response
E-mail	contact@cmdlabs.com
Telephone	443-451-7331
Name	cmdLabs>

Fig. 7 – Registry values on a Samsung i607 (Blackjack) device.

```
      0e07 T13 L0000 F0000 UI4 40
      0c1f  T1f  L0000  F0000  STR  [00172838](12)
'+14431234567'
      0c1a  T1f  L0000  F0000  STR  [00172854](12)
'+14431234567'
      003d T1f L0000 F0000 STR [00172870](4) 'FW:'
      0037  T1f  L0000  F0000  STR  [0017287c](26)
'FWD:FW: FWD:Fw: FWD:Fw:Fw:'
      0e08 T13 L0000 F0000 UI4 41057
      0e17 T13 L0000 F0000 UI4 64
      0e06 T40 L0000 F0000 FT 2009-04-15 15:37:23.000
      3008 T40 L0000 F0000 FT 2009-04-15 15:37:23.000
      8001 T13 L0000 F0000 UI4 855640044
```

This output demonstrates that there is only one record currently in the requested database, which is an MMS message (IPM ID 822083597). The message ID of this item can be determined by taking the number shown in bold (1006634986), converting it to hexadecimal (0x3C0007EA), and then shifting the last two digits to the front (0xEA3C0007). This value is useful for locating related files on the mobile device as demonstrated in the "Examining E-mail and MMS Remnants" section of this paper.

When using a Windows Emulator to view data in a cemail.vol file, be aware that some tools apply the timezone setting to date–time stamps while others do not. For instance, comparing messages details extracted using multiple tools reveals that Pocket dbExplorer applies the timezone setting within the emulator to date–time stamps whereas pdblist and XACT interpret date–time stamps in raw form.

3.5. *Examining Registry hives*

The Registry on Windows Mobile contains various details about the configuration and use of a device. The Registry on Windows Mobile devices has a hierarchical structure similar to other Microsoft operating systems as shown in Fig. 7 using the Microsoft Remote Registry Editor. The System Registry hive contains information such as network connections. For instance, information about recently connected WiFi access points is recorded under the "HKLM\Comm\ConnMgr\Providers" key. The User Registry hive contains information associated with a particular user profile on the device, such as contact details entered by the owner of the device as shown in Fig. 7.

Examples of other useful keys in the User Registry hive are listed in Table 5.

Table 5 – Items in the user Registry hive on Windows Mobile devices of potential interest.

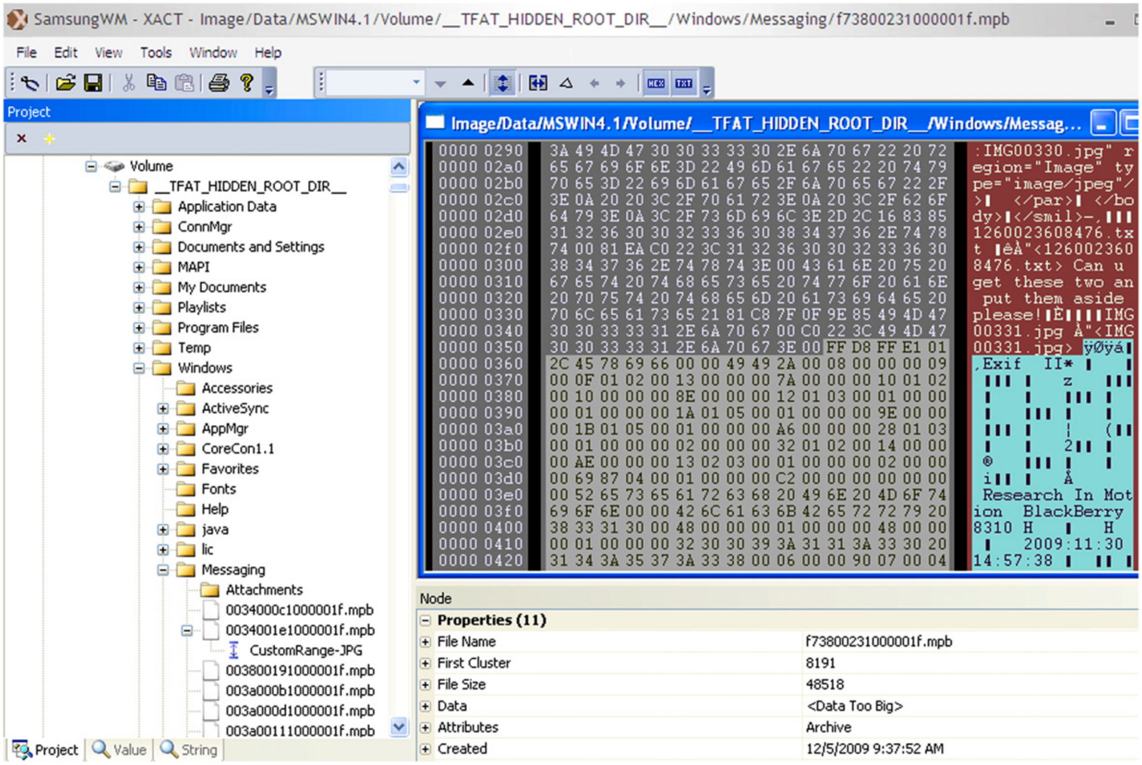
Registry key	Description
HKCU\ControlPanel\Owner	Contact details entered by user
HKCU\System\State\Shell	Most recently used (MRU) items
HKCU\Software\Microsoft\pMSN\SavedUsers	Windows Live ID
HKCU\ControlPanel\Home\CurBgImageName	Home screen background image
HKCU\Comm\EAPOL\Config	WiFi access point information

3.6. *Examining e-mail and MMS remnants*

When MMS and e-mail messages are received and opened, or are composed and sent, on a Windows Mobile device, certain artifacts of these activities are created. These artifacts can be useful to forensic analysts because they indicate when specific messages were created or viewed on the device, even after the original message has been deleted. In addition, when dealing with deleted messages, associated artifacts can remain on the device indefinitely and may contain data associated with the original message.

E-mail message header details, including To, From, Subject, and attachment name, are stored in the cemail.vol file. When these messages are opened on a Windows Mobile device, ".mpb" files are created in the "\Windows\Messaging" folder with message content. In addition, when e-mail attachments are opened on a device ".att" files are created in the "\Windows\Messaging\Attachments" folder.

Data from viewed SMS/MMS messages, stored in "\Windows\Messaging" in ".mpb" files, can include remnants of



SamsungWM - XACT - Image/Data/MSWIN4.1/Volume/__TFAT_HIDDEN_ROOT_DIR__/Windows/Messaging/f73800231000001f.mpb

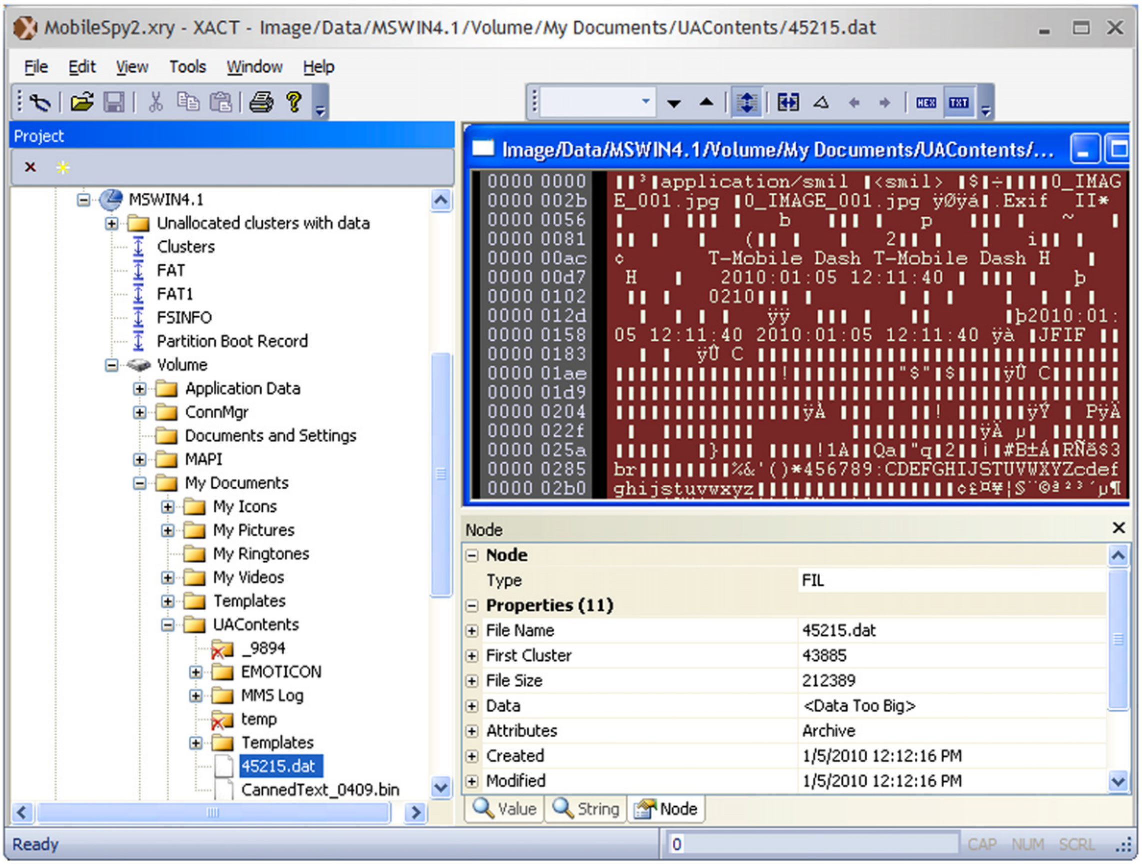
Image/Data/MSWIN4.1/Volume/__TFAT_HIDDEN_ROOT_DIR__/Windows/Messag...

```
0000 0290  3A 49 4D 47 30 30 33 33 30 2E 6A 70 67 22 20 72  :IMG00330.jpg" r
0000 02a0  65 67 69 6F 6E 3D 22 49 6D 61 67 65 22 20 74 79  egion="Image" ty
0000 02b0  70 65 3D 22 69 6D 61 67 65 2F 6A 70 65 67 22 2F  pe="image/jpeg"/
0000 02c0  3E 0A 20 20 3C 2F 70 61 72 3E 0A 20 3C 2F 62 6F  >▮  </par>▮ </bo
0000 02d0  64 79 3E 0A 3C 2F 73 6D 69 6C 3E 2D 2C 16 83 85  dy>▮</smil>-,▮▮▮
0000 02e0  31 32 36 30 30 32 33 36 30 38 34 37 36 2E 74 78  1260023608476.tx
0000 02f0  74 00 81 EA C0 22 3C 31 32 36 30 30 32 33 36 30  t ▮êÀ"<126002360
0000 0300  38 34 37 36 2E 74 78 74 3E 00 43 61 6E 20 75 20  8476.txt> Can u
0000 0310  67 65 74 20 74 68 65 73 65 20 74 77 6F 20 61 6E  get these two an
0000 0320  20 70 75 74 20 74 68 65 6D 20 61 73 69 64 65 20   put them aside
0000 0330  70 6C 65 61 73 65 21 81 C8 7F 0F 9E 85 49 4D 47  please!▮È▮▮▮▮IMG
0000 0340  30 30 33 33 31 2E 6A 70 67 00 C0 22 3C 49 4D 47  00331.jpg À"<IMG
0000 0350  30 30 33 33 31 2E 6A 70 67 3E 00 FF D8 FF E1 01  00331.jpg> ÿØÿá▮
0000 0360  2C 45 78 69 66 00 00 49 49 2A 00 08 00 00 00 09  ,Exif  II* ▮   ▮
0000 0370  00 0F 01 02 00 13 00 00 00 7A 00 00 00 10 01 02   ▮▮▮ ▮   z   ▮▮▮
0000 0380  00 10 00 00 00 8E 00 00 00 12 01 03 00 01 00 00   ▮   ▮   ▮▮▮ ▮
0000 0390  00 01 00 00 00 1A 01 05 00 01 00 00 00 9E 00 00   ▮   ▮▮▮ ▮   ▮
0000 03a0  00 1B 01 05 00 01 00 00 00 A6 00 00 00 28 01 03   ▮▮▮ ▮   ¦   (▮▮
0000 03b0  00 01 00 00 00 02 00 00 00 32 01 02 00 14 00 00   ▮   ▮   2▮▮ ▮
0000 03c0  00 AE 00 00 00 13 02 03 00 01 00 00 00 02 00 00   ®   ▮▮▮ ▮   ▮
0000 03d0  00 69 87 04 00 01 00 00 00 C2 00 00 00 00 00 00   i▮▮ ▮   Â
0000 03e0  00 52 65 73 65 61 72 63 68 20 49 6E 20 4D 6F 74   Research In Mot
0000 03f0  69 6F 6E 00 00 42 6C 61 63 6B 42 65 72 72 79 20  ion  BlackBerry
0000 0400  38 33 31 30 00 48 00 00 00 01 00 00 00 48 00 00  8310 H   ▮   H
0000 0410  00 01 00 00 00 32 30 30 39 3A 31 31 3A 33 30 20   ▮   2009:11:30
0000 0420  31 34 3A 35 37 3A 33 38 00 06 00 00 90 07 00 04  14:57:38 ▮  ▮▮ ▮
```

Node

Properties (11)	
File Name	f73800231000001f.mpb
First Cluster	8191
File Size	48518
Data	<Data Too Big>
Attributes	Archive
Created	12/5/2009 9:37:52 AM

Fig. 8 – Message contents on a Windows Mobile device that contains a digital photograph with embedded EXIF header details from a Blackberry.



MobileSpy2.xry - XACT - Image/Data/MSWIN4.1/Volume/My Documents/UAContents/45215.dat

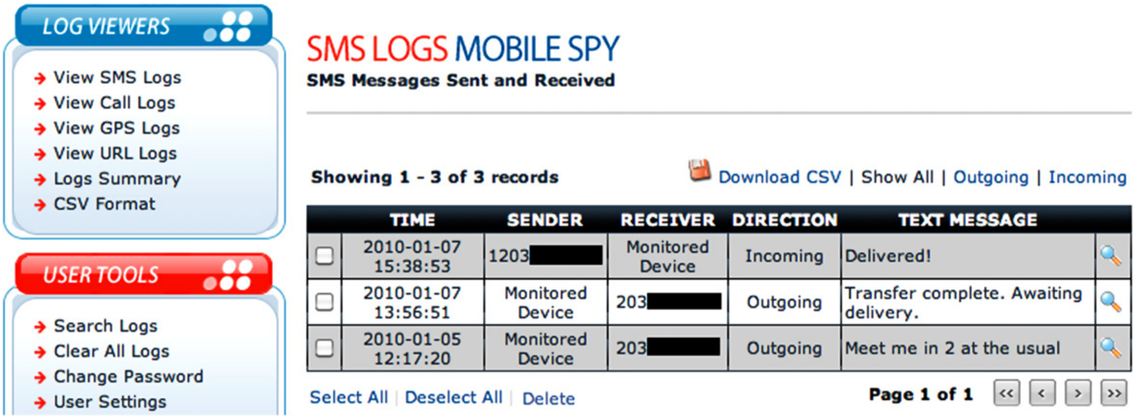
Image/Data/MSWIN4.1/Volume/My Documents/UAContents/...

```
0000 0000  ▮▮³▮application/smil ▮<smil> ▮$▮÷▮▮▮▮0_IMAG
0000 002b  E_001.jpg ▮0_IMAGE_001.jpg ÿØÿá▮.Exif  II*
0000 0056  ▮   ▮ ▮▮▮ ▮    b   ▮▮▮ ▮    p   ▮▮▮ ▮   ~   ▮
0000 0081  ▮▮ ▮    ▮    (▮▮ ▮    ▮    2▮▮ ▮    ▮    i▮▮ ▮
0000 00ac  ¢        T-Mobile Dash T-Mobile Dash H    ▮
0000 00d7   H    ▮    2010:01:05 12:11:40 ▮ ▮▮▮ ▮    þ
0000 0102   ▮▮ ▮    0210▮▮▮ ▮          ▮  ▮  ▮     ▮  ▮  ▮  ▮
0000 012d    ▮   ▮  ▮  ▮    ÿÿ  ▮▮▮ ▮    ▮▮         ▮þ2010:01:
0000 0158  05 12:11:40 2010:01:05 12:11:40 ÿà ▮JFIF ▮▮
0000 0183    ▮  ▮   ÿÛ C ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮
0000 01ae  ▮▮▮▮▮▮▮▮▮▮▮▮▮▮!▮▮▮▮▮▮▮▮▮"$"▮$▮▮▮▮ÿÛ C▮▮▮▮▮▮
0000 01d9  ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮
0000 0204  ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ÿÄ ▮▮▮ ▮ ▮▮! ▮▮▮▮▮▮ÿÝ ▮ PÿÄ
0000 022f   ▮  ▮▮▮▮▮▮▮▮           ▮▮▮▮▮▮▮▮▮▮▮▮ÿÄ µ▮ ▮▮▮▮▮▮
0000 025a  ▮▮▮▮▮   ▮}▮▮▮ ▮▮▮▮!1A▮▮Qa▮"q▮2▮▮¡▮#B±Á▮RÑð$3
0000 0285  br▮▮▮▮▮▮▮▮▮%&'()*456789:CDEFGHIJSTUVWXYZcdef
0000 02b0  ghijstuvwxyz▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮¢£¤¥¦§¨©ª²³´µ¶
```

Node

Node	
Type	FIL
Properties (11)	
File Name	45215.dat
First Cluster	43885
File Size	212389
Data	<Data Too Big>
Attributes	Archive
Created	1/5/2010 12:12:16 PM
Modified	1/5/2010 12:12:16 PM

Fig. 9 – Example ".dat" file containing data associated with a sent MMS message.



SMS LOGS MOBILE SPY

SMS Messages Sent and Received

Showing 1 - 3 of 3 records

	TIME	SENDER	RECEIVER	DIRECTION	TEXT MESSAGE	
□	2010-01-07 15:38:53	1203[redacted]	Monitored Device	Incoming	Delivered!	🔍
□	2010-01-07 13:56:51	Monitored Device	203[redacted]	Outgoing	Transfer complete. Awaiting delivery.	🔍
□	2010-01-05 12:17:20	Monitored Device	203[redacted]	Outgoing	Meet me in 2 at the usual	🔍

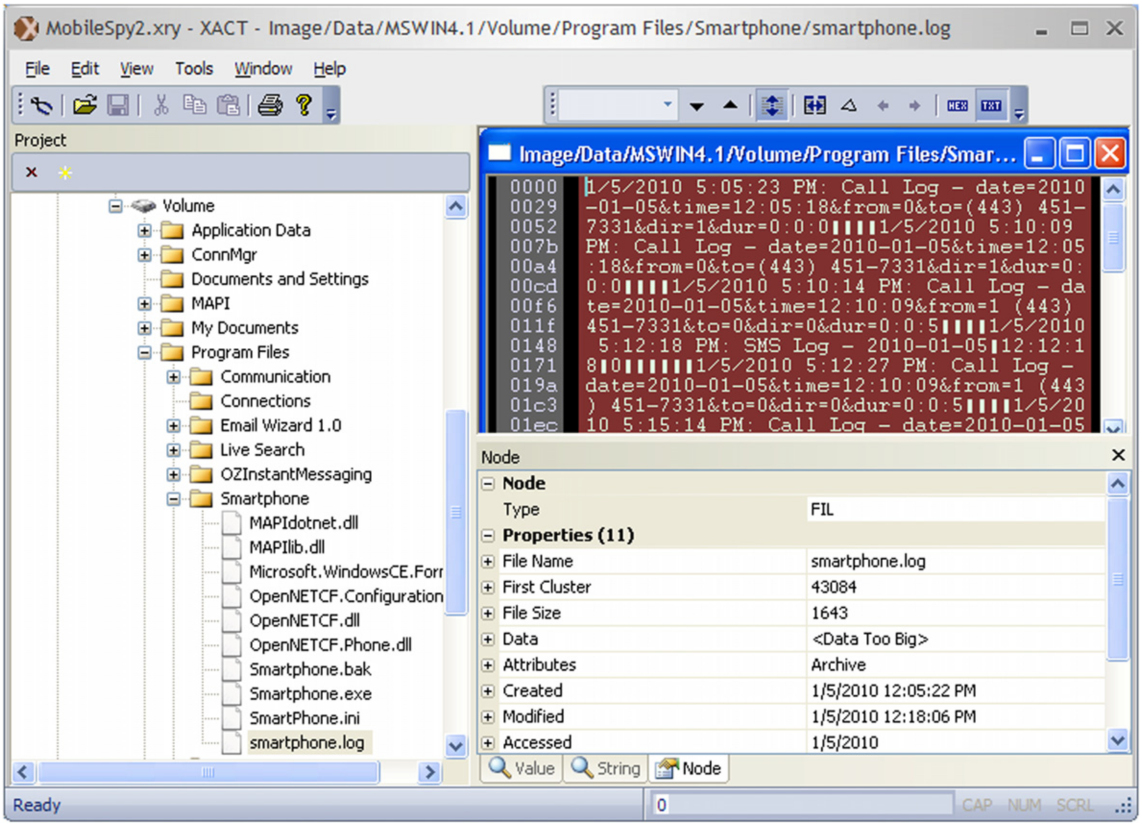
Page 1 of 1

Fig. 10 – MobileSpy Web site showing SMS traffic on a monitored device.

items that were deleted from the cemail.vol file. Fig. 8 shows an ".mpb" file associated with an MMS message on a Samsung i607 (Blackjack) device with a file creation date–time stamp that indicates the message was opened on December 5, 2009. This file includes a digital photograph with embedded EXIF header information showing that it was taken with a Blackberry on November 30, 2009. The original receive message associated with this ".mpb" file was deleted.

The object identifier (OID) of a particular message can be used to associate entries in the cemail.vol file with corresponding ".mpb" files in the "\Windows\Messaging" folder. For instance, content associated with the sample message listed in the previous section is stored in "\Windows\Messaging\EA3C00071000001f.mpb", where the file name starts with the message OID (0xEA3C0007). The last 8 characters of an ".mpb" file define the Microsoft property tag value for the file, which in this instance is the full text of the original message (Microsoft, 2008b).

Some devices also have a "\My Documents\UAContents" folder that contains remnants of sent messages. This folder contains ".dat" files with copies of images sent via MMS, even after the original message has been deleted. Fig. 9 shows the contents of a file "\My Documents\UAContents\45215.dat" that includes a digital photograph that was taken using an



```
0000 1/5/2010 5:05:23 PM: Call Log - date=2010
0029 -01-05&time=12:05:18&from=0&to=(443) 451-
0052 7331&dir=1&dur=0:0:0▮▮▮▮1/5/2010 5:10:09
007b PM: Call Log - date=2010-01-05&time=12:05
00a4 :18&from=0&to=(443) 451-7331&dir=1&dur=0:
00cd 0:0▮▮▮▮1/5/2010 5:10:14 PM: Call Log - da
00f6 te=2010-01-05&time=12:10:09&from=1 (443)
011f 451-7331&to=0&dir=0&dur=0:0:5▮▮▮▮1/5/2010
0148  5:12:18 PM: SMS Log - 2010-01-05▮12:12:1
0171 8▮0▮▮▮▮▮▮1/5/2010 5:12:27 PM: Call Log -
019a date=2010-01-05&time=12:10:09&from=1 (443
01c3 ) 451-7331&to=0&dir=0&dur=0:0:5▮▮▮▮1/5/20
01ec 10 5:15:14 PM: Call Log - date=2010-01-05
```

Node

⊟ **Node**	
Type	FIL
⊟ **Properties (11)**	
⊞ File Name	smartphone.log
⊞ First Cluster	43084
⊞ File Size	1643
⊞ Data	<Data Too Big>
⊞ Attributes	Archive
⊞ Created	1/5/2010 12:05:22 PM
⊞ Modified	1/5/2010 12:18:06 PM
⊞ Accessed	1/5/2010

Fig. 11 – MobileSpy program installed in "Program Files\Applications\Smartphone" with "smartphone.log" file recording activities on the device.

HTC S620 (Dash) and sent in an MMS message. The creation date–time stamp of this ‘‘.dat’’ files shows when the MMS message was composed. Additional details about sent and received MMS messages are recorded in text files in the ‘‘\My Documents\UAContents\MMS Log’’ folder.

4. Malicious eavesdropping case study

The emergence of programs that can monitor activities remotely on Windows Mobile devices has raised privacy and security concerns in governments and businesses. MobileSpy and FlexiSpy are two such programs that can be installed on a Windows Mobile device to enable a remote individual to monitor user activities like SMS and voice conversations. These programs send information from the mobile device to a Web server where the remote individual can review the gathered information as shown in Fig. 10.

Average users will not notice that such a program is running on their device. Although the MobileSpy process (Smartphone.exe) can be seen running in memory on the device using Remote Process Viewer, it does not appear in the Task Manager. However, these programs leave sufficient traces to be detectable by forensic analysts. Forensic analysis of a Windows Mobile device with Mobile Spy installed reveals traces on the file system and Registry. For instance, the MobileSpy program is placed in the ‘‘Program Files\Applications\Smartphone’’ folder. As shown in Fig. 11, this folder includes a file ‘‘smartphone.log’’ which maintains a record of activities that were monitored by the MobileSpy program.

In addition, MobileSpy places a shortcut file in ‘‘Windows\StartUp\’’ and creates the following Registry entries:

```
[HKEY_CURRENT_USER\Software\RetinaxStudios]
  "isLogUrl" = dword:1
  "isLogSMS" = dword:1
  "isLogPhoneCall" = dword:1
  ``Username" = "''
  "ReportTimer" = dword:f
  "RememberUser" = dword:1
  ``Password" = "''
  ``BlackList" = "0010001''
  "AutoLogin" = dword:1
```

In early versions of MobileSpy, the username and password for authentication between the device and Web server were stored in the Registry in plaintext (Fogie, 2007). Later versions protect the username and password, but it can still be obtained by dumping memory of the ‘‘Smartphone.exe’’ process.

5. Conclusions

Despite their small size, Windows Mobile devices can contain substantial amounts of information about their users, including with whom they were communicating and what they were doing at particular times. Although there are aspects of Windows Mobile devices that will be familiar to forensic analysts, there are sufficient variations to make Windows Mobile Forensics a distinct discipline with its own unique tools and techniques. As Windows Mobile devices become more prevalent, there is a growing need for forensic analysts who can acquire evidence from these devices, and examine their contents. There is also a need for further research and development to improve our ability to extract information from Windows Mobile devices, including more deleted data.

REFERENCES

Casey. Digital evidence and computer crime. In: Byard R, Corey T, Henderson C, editors. The encyclopedia of forensic and legal medicine. Elsevier; 2005.

Casey. Recovering deleted text messages from Windows Mobile devices, https://blogs.sans.org/computer-forensics/2009/10/22/recovering-deleted-text-messages-from-windows-mobile-devices/; 2009.

Fogie S. Inside mobile-spy ‘‘spouseware’’, informIT. Indianapolis: Pearson Education, http://www.informit.com/articles/article.aspx?p=1077909; 2007.

Klaver C. Windows Mobile advanced forensics. Journal of Digital Investigation; 2010.

van der Knijff R. Embedded systems analysis in handbook of digital forensics and investigation. San Diego: Elsevier; 2009.

Microsoft. EDB data types and size limits, http://msdn.microsoft.com/en-us/library/ms885368.aspx; 2010.

Microsoft. Embedded database system technologies, http://msdn.microsoft.com/en-us/library/ms838188.aspx; 2005.

Microsoft. File system boot process, http://msdn.microsoft.com/en-us/library/aa912276.aspx; 2008a.

Microsoft. Message content properties, http://msdn.microsoft.com/en-us/library/bb446140.aspx; 2008b.

XDA. Backup and restore your cemail.vol easily, http://forum.xda-developers.com/showthread.php?t=302909; 2006.