HTC Fuze

Senator Patrick Leahy

Center for Digitial Investigation

By: Colby Lahaie

12/12/2012

Table of Contents

Executive Summary

Many people use their cell phones to do a variety of different things, from storing word documents, using programs, playing games, using the GPS for travel, and other such things. We received the AT&T HTC Fuze from Microsoft to perform different research and development techniques with the phone. We chose to research how to acquire data off of the phone. Data acquisition and analysis is one of the most important things that a forensic investigator would want to do because it is a way to extract and preserve all of the data. In this report, specific tools such as MIAT (Mobile Internal Acquisition Tool) and Cellebrite were used to find key aspects of the HTC Fuze that would be helpful during a forensic investigation. These tools make it easier for law enforcement and private forensic investigators to extract data off the HTC Fuze mobile phone to use in criminal investigations. In this report we were able to figure out how to extract all of the data from the phone, but we were not able to create a forensic image nor were we able to recover deleted data or extract the wireless network connections.

Background

We wanted to find key aspects of the AT&T HTC Fuze mobile device using specific forensic tools. We needed to figure out a way to take a physical image of the phone without the need for a memory card. The phone operates on a Windows Mobile 6.1 Professional Operating System.

Figure 1: HTC Fuze Specifications

Operating System	Windows Mobile® 6.1 Professional
Processor	Qualcomm® MSM7201A™, 528 MHz
Memory	ROM: 512MB / RAM: 288MB
Display	2.8-inch TFT-LCD flat touch-sensitive screen with VGA resolution
Dimensions	4.0 inches (H) X 2.0 inches (W) X 0.7 inches (T)
Weight	5.8 ounces (with battery)
Network	HSDPA/WCDMA: • 2100/1900/850 MHz • 384kbps up-link and 3.6 Mbps down-link speeds Quad-band GSM/GPRS/EDGE: • 850/900/1800/1900 MHz (Band frequency and data speed are operator dependent.

(HTC)

Procedures

The HTC Fuze that was given to us was a new phone with no SD card. The HTC Fuze file system is Transaction-Safe FAT File System, or TFAT, which is a modified version of the FAT File System. The LCDI initially extracted data with the Cellebrite UFED Physical Pro, using the "Phone Data Extraction" option, and installed a registry viewing program. According to the Cellebrite website, Cellebrite is supposed to be able to extract existing, hidden, and deleted phone data, such as call history, text messages, contacts and other such data. However, when the phone was connected to the Cellebrite UFED Physical Pro, we were only able to recover existing phone data (call logs, contacts, SMS – text messages, images, ringtones, audio, video), but not deleted data.

Our team then connected the HTC Fuze phone to LCDI Research and Development Workstation 5, but the phone was not recognized as any type of device. Our team searched how to connect the HTC Fuze to Windows 7 and how to detect the HTC Fuze on the computer because it would not connect on its own. We found many websites that said to download the program called Windows Mobile Device Center, which is synchronization software that allows various content such as video, music, contacts, and other files, to be synched through Microsoft Outlook between Windows Mobile devices and Windows operating systems. We downloaded the software from Microsoft's website and found that it allowed the computer to see the HTC Fuze hard drive as a portable device, but not as a removable device.

When the HTC Fuze is plugged into a computer via USB three options should show up on the phone screen: ActiveSync, Disk Drive, or Internet Sharing. Click on Disk Drive and the SD card will show up as a removable device. If this doesn't show up or you want another way to find out how to change the connection type to the computer; click the drop down start icon in the upper left corner, click Settings → Connections → and then USB to PC, which will show you the different types of USB connections to the PC.

When clicking on the portable device icon it shows the phones hard drive, but it has no name or letter, it just has a (\) symbol, therefore there is no way to change the name. Our team researched how to mount an HTC Fuze hard drive, but found nothing. There is also no description of the drive under the properties tab. When you click the hard drive you can view some of the folders on the hard drive which are: My Documents, Program Files, Music, Content, Documents and Settings, ConnMgr, and Application Data. You can copy these files off of the hard drive onto the desktop, but there is no way to create a forensic image of the hard drive itself. This is because it is not recognized by the computer and none of our forensic software can recognize it either. Our team tried using Raptor 2.0, which is a program based on Ubuntu that focuses on computer forensics, to see if we could view the phone's hard drive. Raptor 2.0 could not verify or see the cell phone drive either. We also tried to see if we could view the hard drive in EnCase and FTK Imager, but we could not. Our team also tried finding the drive in computer management; however, the drive won't show up in Disk Management.

While researching, our team was able to find a product made by AccessData called 'Mobile Phone Examiner Plus' (MPE+). According to the product website, this software allows for examiners to extract data, such as contacts, photos, messages, call history, emails, and the like, from over 3,000 different mobile devices, which includes Apple, Android, Blackberry and Windows Mobile devices (AccessData, 2011). The software does have support for the HTC Fuze as well. The software can acquire file systems, carve data, and do much more which is extremely beneficial for a forensic investigator. It also integrates with FTK and FTK Imager, making it easier for one to acquire images and data of mobile devices. A demo video about MPE+ found on the website shows how the program works and how to create an image and extract data from a Windows Mobile phone. Our team downloaded MPE+ and attempted to use it, but because it was a demo version, we were not to do create an image of the phone. If you buy the program or the license; however, you will be able to acquire the file system of the HTC Fuze.

Our team noted on the phone under Settings → System → Memory that the phone has two types of internal Memory (Storage and Program). Comparing the total size of the Storage memory with the disk that is recognized under Microsoft Windows it shows that they are likely the same (Both list 294.90 MB as the total capacity). Also, under Programs → Tools we found a File Explorer application. The File Explorer lists appear to be the same files and folders that are viewable under Microsoft Windows. However, in addition there are multiple files and two directories that are not listed under windows unless you disable the option to hide protected operating system files. We also noted that the .vol files appear to be the data bases containing call logs, calendars, etc. When attempting to open or copy the files off of the phone to inspect them, an access denied error was shown.

Our team then found an open source application called the MIAT (Mobile Internal Acquisition Tool). The application requires an SD Card to function. According to the website, MIAT is designed to acquire internal data without any external software (University of Rome). It is also said to be forensically sound. The program is installed on a blank removable SD Card, placed in the phone, and then the program runs. It seizes all of the phone’s data to the SD Card. (Note: MIAT is designed to only work with mobile phones running Windows Mobile and Symbian Operating Systems.) A 4GB SD card was provided to us in order to extract information from the phone to the card using MIAT.

Our team downloaded the software and followed the instructions that came with the program when downloaded, which you can find in the Instructions tab below. We created a folder called 2577 on the blank SD card, put the miat.exe in the folder, put the SD card into the phone, and then waited for it to extract the data. According to the instructions, the program was supposed to autorun on the phone, but it did not. We removed the SD Card from the phone a few times and then put it back in the phone and again nothing happened. We had to manually access MIAT through the phones’ file explorer. In the phone, click the drop down start button, click programs, click tools, and then click file explorer. In the file explorer, click the memory card, click the 2577 folder, and then click the miat.exe. The MIAT program will open and a button that says “seize” will appear. After you choose to save it to the memory card, the program will start seizing the information from the phone. It takes about 2-3 hours. Our team removed the SD card from the phone and plugged it into a SD card reader connected it to LCDI Research and Development Workstation 5 and copied the seized files to a network hard drive. Many files and folders were seized. The instructions said that a copy of the file system would be in the "root" folder, but we could not find it.

Another member of the LCDI was added to the HTC Fuze project so that we could have a new set of eyes to try and find a different approach to retrieving the data. We also wanted to see if we could image the HTC Fuze using FTK Imager without any other program, without MIAT. We also wanted to create "evidence" on the SD card from the phone so that we would have some files to look at in FTK Imager. Our team did some research and found that the HTC Fuze has a program on it called Sprite Backup that will back up all of the information off of the phone's memory to the SD card in the phone. We conducted the backup using Sprite Backup and were able to copy content from the phone to the SD card. We also took a few pictures with the phone and saved them to the SD card to be used as “evidence”. We then plugged the Fuze into the Workstation 5 via a USB cable and began to image the SD card, which showed up as the “F” drive, as an E01 file. It took FTK about 25 minutes to image the SD card. Although we used the backup utility on the phone to add more data to the SD card, in the image file in FTK, it only showed the pictures that we had taken as evidence.

Our team successfully used FTK Image to create an image of all of the files and folders that were seized from the HTC Fuze phone with MIAT. When viewing the .ad1 file of the files seized by MIAT, we found it took a much more comprehensive image of the entire phone than Sprite Backup. Our team then tried copying a folder from the phone to the SD card as a test and we were able to successfully copy it. We then thought that it would be able to copy all of the files and folder without MIAT from the phone to the SD card, but some of the files, specifically the Windows files, were not able to be copied because of denied access and because they were being used by the phone, and therefore, copying all of the files from the phone will not get all of the data needed for an investigation, but using MIAT would.

Since there was no data on the phone, because it was a test phone, we planted evidence on the phone for a test forensic investigation. We took the HTC Fuze around Champlain College campus for a week, connecting to different wireless networks to get different IP addresses so that we could see how the phone connects. We then re-seized the data from the phone onto the SD card with MIAT. We also wanted to see if we could do more than just recover data from the HTC Fuze, so we started taking random pictures with the HTC Fuze. We kept some of the files as well as deleted some. We wrote down all of the information; what type of file, the size of each file, and the file name of each picture taken and each picture deleted. We also added fake contacts. After we completed all of these steps, we re-seized the data on the phone with MIAT, which took 2-3 hours. We then attempted retrieving all of the pictures, contacts and wireless access points off of the MIAT image using our Cellebrite UFED Physical Pro and carving in EnCase.

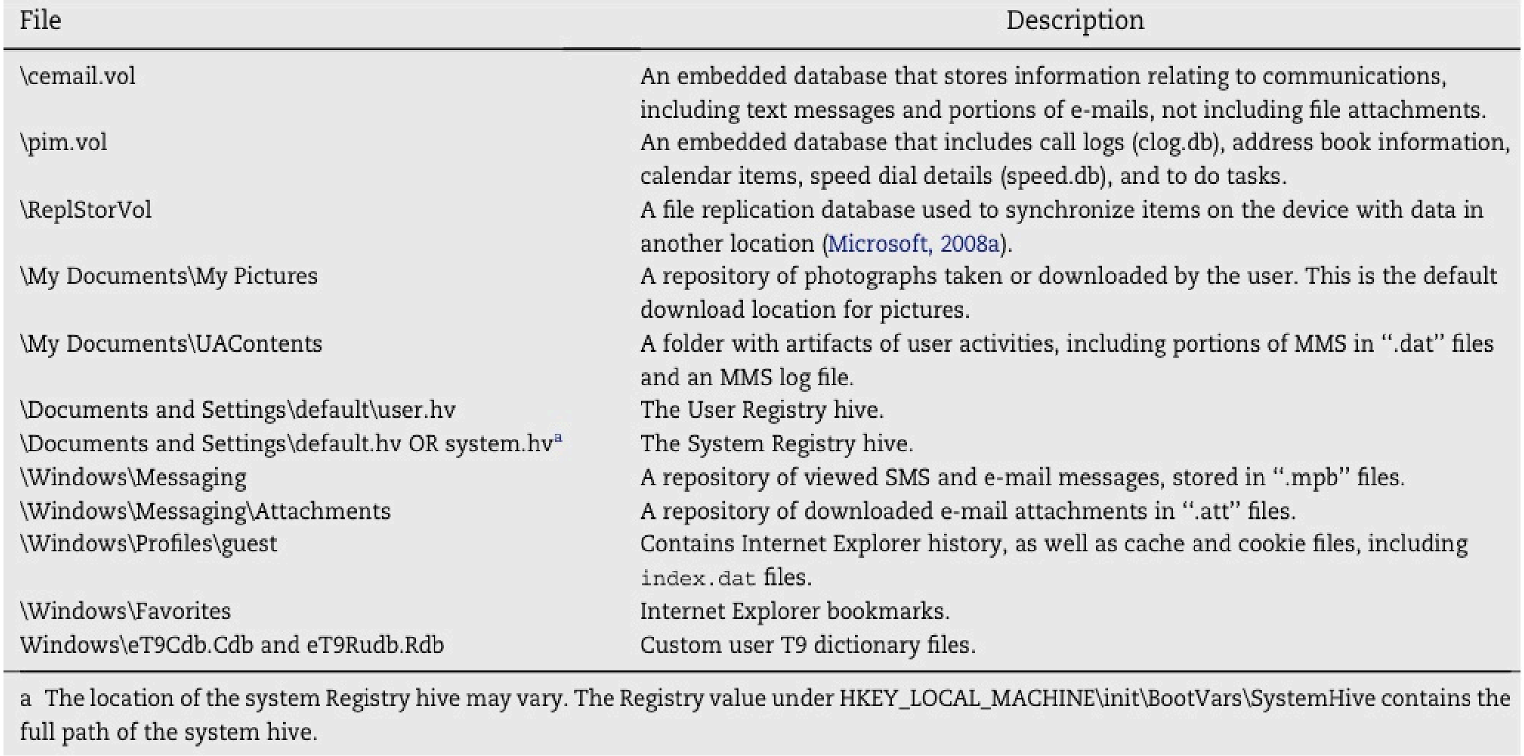
Our team attempted to retrieve the deleted images and the wireless access points using the Cellebrite UFED Physical Pro, but we were unable to retrieve the deleted images and the wireless access points. We also tried carving for data off of the image of the extracted phone data, from MIAT, and we were unable to find the deleted pictures. In our research we found that MIAT was unable to seize deleted data off of phones. However, if a suspect had an SD card in this phone he or she would most likely save the pictures to the SD card, which most people do in case they want to access them whenever they want later down the road, we would be able to image the SD card and recover the deleted images and other data.

Another member of LCDI was added to the HTC Fuze project as the student lead because she had experience with mobile operating systems, such as the Kindle, and we needed her input. Our team conducted more research on the phone and we also consulted with industry professionals, but they had no insight and had never gone beyond an acquisition using Cellebrite or similar tools.

The project was on standby for about a month when it was picked back up by me on May 30, 2012. I tried to make outgoing calls and send text messages, but because we did not have a SIM card, they were not able to be sent, however; the outgoing calls were still added to the outgoing calls list and the text messages were saved as drafts. I then attached the HTC Fuze to the Cellebrite UFED Physical Pro again and I extracted phone data as well as conducting a physical extraction of the phone, which was not previously done before, to be analyzed the UFED Physical Analyzer software. The Cellebrite provides a nice examination report that will contain a summation of all images, texts (documents not SMS messages), video, and audio. This report includes active and deleted data, however; when reviewing the report it does not show the images that I deleted. When analyzing the physical extraction of the phone, I was able to view the same files and folders of the file system of the phone, which we extracted with MIAT, such as the Windows folder, the application data folder, .vol files, etc. Also, when analyzing the .ufd file that Cellebrite created, I was able to figure out that there are 4 partitions on the HTC Fuze, but all of the phone's data is stored on the fourth partition.

I was also able to find different sources that explained what files different information on a Windows Mobile phone is stored as well as how to recover deleted text messages. One source that I found was a blog on the cmdLabs website called "Advances in Windows Mobile Forensics", which talks about different files that are very important for forensics. Two very important files are the cemail.vol file and the pim.vol file. The cemail.vol file is a database that contains information on text messages and portions of e-mails (Casey, Advances in Windows Mobile Forensics, 2010). The pim.vol file is "an embedded database that contains call logs (clog.db), address book information, calendar items, speed dial details (speed.db), and to do tasks" (Casey, Advances in Windows Mobile Forensics, 2010). There are also other very important files such as, the user/system registry hive, internet explorer history etc., which you can find in figure 2.

Figure 2: Useful Windows Mobile Device Evidence



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.

(Casey, Advances in Windows Mobile Forensics, 2010)

During my research I was also able to find a blog on the SANS Computer Forensics website called, “Recovering Deleted Text Messages from Windows Mobile Devices”. The blog talks about recovering deleted text messages from the cemail.vol file using different tools which include: Windows Mobile Device Center, Microsoft Device Emulator, Microsoft Visual Studio 2008, a Windows Mobile 6.1.4 Emulator Image, the pdblist utility from the itsutils suite, and a copy of the acquired cemail.vol file (Casey, Recovering Deleted Text Messages from Windows Mobile Devices, 2009). I downloaded the Microsoft Device Emulator and the Windows Mobile 6.1.4 Profession Emulator image. I then downloaded the itsutils suite, which is a suite of tools used for extracting different data off of Windows Mobile phones other than just recovering deleted texts, and a 90 day trial of Microsoft Visual Studio 2008. I downloaded and installed all of the files on a Windows Vista x86 virtual machine, because all of the files worked on Windows Vista.

The first step to accessing the deleted text message data is to mount the cemail.vol file into the Windows Mobile Emulator. To do this, first open up the Windows Mobile 6.1.4 Professional emulator. Then, click file and click configure. Under the general tab change the shared folder to the folder that the cemail.vol file is located in and click ok. To access the file in the emulator, click the start icon, click programs, and click file explorer. Click the drop down button that says “My Documents” and change it to storage card. You should then see the cemail file, which means you have successfully mounted the acquired cemail.vol file.

“After launching and configuring the desired Windows Mobile Emulator, it is necessary to create a conduit that itstutils uses to send commands to the Emulator by establishing an ActiveSync connection” (Casey, Recovering Deleted Text Messages from Windows Mobile Devices, 2009). To do this, open up Microsoft Visual Studio 2008, click tools and then click Device Emulator Manager. Find the emulator you are using and right-click on it and select Cradle (Casey, Recovering Deleted Text Messages from Windows Mobile Devices, 2009). You will also have to allow DMA connections in Windows Mobile Device Center. In order to do this open up Windows Mobile Device Center and click on Connections Settings. Click the drop down button and change it to DMA. Hit ok and then the Emulator will automatically connect to Windows Mobile Device Center.

Once you have finished all these steps successfully, you can now use the pdblist utility from the itsutils suite to examine the cemail.vol file. Save the itsutils suite onto the desktop so that you can easily navigate to it in command prompt. Open up command prompt and cd to the desktop and then cd to the itsutils folder. The pdblist utility can list accessible volumes, including the virtual SD card of the emulator, where we saved the cemail.vol to, as shown below:

```
C:\Users\clahaie\Desktop\itsutils>pdblist –v
volume {00000000-0000-0000-0000-000000000000} \Documents and Settings\default.vol
volume {d4353000-f9ee-b0b5-6f27-4a5f1daea1d0} \ReplStorVol
volume {2fbc9100-1032-ef8d-0ad8-b9daf1c3e5d0} \mxip_notify.vol
volume {7b508a00-7bb1-7d36-de99-eb6df2d8b518} \mxip_swmgmt.vol
volume {77868700-df80-7502-b92e-386aa142ba4d} \cemail.vol
volume {d3872200-005b-8665-6a00-895ddf520898} \mxip_system.vol
volume {d071d100-fb8f-1505-782c-e71b23e00165} \mxip_lang.vol
```

To list the components of databases that are accessible via the emulator (Casey, Recovering Deleted Text Messages from Windows Mobile Devices, 2009), type the following command:

```
C:\Users\clahaie\Desktop\itsutils>pdblist –D
volume {77868700-df80-7502-b92e-386aa142ba4d} \cemail.vol
oid31000077: dbase F00000017 T00000000    0    356 ... 'fldr31000028'
  ORDERING: 0e060040:00000000 0c1a001f:00000002 0037001f:00000002 001a0013:00000000
oid31000072: dbase F00000017 T00000000    0    356 ... 'fldr31000026'
  ORDERING: 0e060040:00000000 0c1a001f:00000002 0037001f:00000002 001a0013:00000000
oid3100006d: dbase F00000017 T00000000    0    356 ... 'fldr31000024'
  ORDERING: 0e060040:00000000 0c1a001f:00000002 0037001f:00000002 001a0013:00000000
oid32000068: dbase F00000017 T00000000    0    356 ... 'fldr3100001d'
  ORDERING: 0e060040:00000000 0c1a001f:00000002 0037001f:00000002 001a0013:00000000
[cut for brevity]
```

The last option for pdblist that was used in this blog allows an investigator to extract information from a particular object by name. If you work through the components from the above command output, you will find details of different text messages, including the actual text and the date/time stamp of the text (Casey, Recovering Deleted Text Messages from Windows Mobile Devices, 2009), using the following command:

```
C:\Users\clahaie\Desktop\itsutils>pdblist –d fldr31000026
```

Unfortunately we did not have an active SIM card in our HTC Fuze so we were unable to send any text messages and therefore we were unable to recover any text messages.

After conducting all of this research we have been able to access and recover some of the most important aspects of the HTC Fuze to be used during a criminal investigation. We have learned how to extract almost all of the data off of the phone, using Cellebrite, and we have learned how to acquire deleted text messages using tools from the itsutils suite. We have concluded our research on the HTC Fuze and we are closing this project.

Results

During the investigation of the HTC Fuze, we found that data can be extracted from the phone without any outside software such as Windows Mobil Device Center (WMDC). You can copy and paste pictures, for instance, from the phone to the removable SD card inside the phone. However, in order to extract all of the data off of the phone for a complete image, you would need a tool, Mobile Internal Acquisition Tool (MIAT), that allows you to essentially retrieve a complete back up of the phone. The tool extracts or seizes the data from the phone and places it onto a removable SD card that can then be imaged with FTK Imager or EnCase. According to the makers of MIAT, the tool is forensically sound. We found out that we do not need to use WMDC to extract the data off of the phone. When we first received the phone we didn't have an SD card so we found WMDC which sees the phone and allows data through Outlook to be synced between a Windows Mobile phone and a Windows Operating System. This program is needed to view data on the phone and to sync data, unless you have an SD card, then this program isn't required.

After further investigation of the phone we have concluded that it is possible to find deleted data using the pdblist utility from the itsutils suite, but it is not possible to data carve with EnCase, directly from the HTC Fuze, unless the images were saved on a removable SD card. The phone is locked tightly, but developers and mobile forensic professionals have developed tools to access and analyze important data of a Windows Mobile phone. However, there is no way to get into it with any operating system or any notable forensic software (EnCase or FTK), but there are tools. The project is complete and our research has been concluded.

Discussion

According to all of our research and insight into the HTC Fuze, we have realized that the phone was essentially built to be unbreakable, unless you figure out how to jailbreak it, and that none of the forensic software that most industry professionals use, such as FTK, EnCase, or Cellebrite can be used to break in. However, there are tools such as, MIAT or itsutils that can be used to acquire data off of the phone, but some of them are not technically forensically sound. Windows phones are not commonly sold or bought and because there are other more popular mobile phones and operating systems such as phones with the Android OS or the Apple iOS. Also, all of the research that we have conducted cannot go to any other phone because the tools that we used are Windows Mobile Operating System specfic. It could also be that the HTC Fuze is different than other Windows or Symbian phones. We can try to find a different Windows or Symbian phone to see if our research, as well as additional research, can be used to essentially break into the phone and extract the data that is needed for a thorough criminal investigation case.

In addition, we can try to find additional research to find key aspects of the phones with Android OS and Apple iOS, which would be helpful during a forensic investigation.

Instructions

Cellebrite Instructions:

1) Go to www.cellebrite.com.
2) Click on the "support center" tab at the top of the page, and select "mobile data support."
3) Click the "supported phones" link on the left side of the page.
4) Now, using the dropdown menu in the center of the page, select the make of the phone, then select the model.
5) The results of the phone found should then be displayed. Take note of the cable number, for this is the cable that you will find in the Cellebrite case to attach your given phone.
6) Now, attach the cable to the phone, and the phone to the Cellebrite machine on the left side. Also make sure that on the right side of the Cellebrite machine the flash drive is inserted into the USB port.
7) Power up Cellebrite. A menu should appear on the screen. There are a variety of options you can chose from, but in this case we are going to select "Extract Phone Data" and press OK.

8) The next menu that appears has you select the source vendor of the phone. In this case we chose "HTC 6850 Touch Pro/Fuze."
9) After selecting the phone, a select source memory screen will come up. Scroll up and down using the arrow keys and select where you want to get data from on the phone. Press next.
10) Select USB flash drive as the target output, and press OK.
11) The next screen lets you select different content types from the phone. You can select all or none of the boxes here depending on what data you want to retrieve off of the phone. For the HTC Fuze, I selected all of the boxes. Then, press the right arrow once, then again to start the extraction.
12) The attached flash drive can then be inserted into a PC to read the report that Cellebrite generates about the info it seized off of the phone.

MIAT Instructions:

Installation:

1) Create a folder named "2577" in an empty SD card.
2) Copy MIAT.exe in the 2577 folder.

Use:

1) Put the SD in the device to be seized.
2) Autorun will run the MIAT automatically. (If it does not autorun manually access the program. In the phone, click the drop down start button, click programs, click tools, and then click file explorer. In the file explorer, click the memory card, click the 2577 folder, and then click the miat.exe.)
3) Once MIAT started, you must specify in "Save" field the path to the SD card (e.g. /Storage Card).
4) Click "Seize" button and wait for the seizure process is complete (a pop-up will appear). Depending on the device hardware and on the file system's occupation, the process may take a while.
5) Extract the SD and place it in a SD-reader connected to your PC.
6) In the root folder of the SD you will find the device's file system copy, plus a folder named "Statistics", containing all logs and statistics about the seizure process.

FTK Imager Instructions:

Creating Image:

1) Open FTK Imager.
2) Click the file button in the tool bar and then click create disk image.
3) Select Physical Drive and click next.
4) Choose which device/drive you want to image from the available drives and hit finish.
5) Click Add to add a destination and a destination image type for your image of the drive.
6) Choose E01 file so that the image can be opened in both FTK Imager and EnCase for later use, and then click next.
7) Fill out the Evidence Item Information and click next.
8) Choose the name of the image and the destination where the image will be saved and then click finish.
9) The image of the drive will be created.

Opening Image with FTK Imager:

1) Open FTK Imager.

2) Click the file button in the tool bar and then click add image or click the picture underneath the file button with the magnifying glass and the plus.
3) Click Image File in the source type and click next.
4) Click Browse and find the image file that you created and open it. Then click finish.
5) The image will show up in the evidence tree so you can look at all of the data of the hard drive that you imaged.

Opening Image with EnCase:

1) Open EnCase.
2) Click the file button in the tool bar and then click new or click the new icon underneath the file button to add a new case.
3) Fill out the case options and click finish.
4) Find and open the folder where you saved your image file. Drag the file into EnCase and it will open the image for forensic use.

Recovering Deleted Text Message Instructions:

Adding cemail.vol to Emulator:

1) Open Emulator.
2) Click file
3) Click configure
4) Next to shared folder click the button with three dots (…).
5) Point the shared folder to the folder holding the cemail.vol file.
6) Click ok.

Accessing cemail.vol in Emulator:

1) Click start.
2) Click programs.
3) Click File explorer.
4) Click the drop down button that says “My Documents”, change it to storage card.
5) You should see the cemail.vol file.

Microsoft Visual Studio 2008 Cradling Emulator:

1) After launching and configuring the desired Windows Mobile Emulator, it is necessary to create a conduit that itstutils uses to send commands to the Emulator by establishing an ActiveSync connection. Open Windows Visual 2008.
2) Click the tools menu.
3) Click on Device Emulator Manager in Visual Studio.
4) Find the Emulator that you are currently using in the list.
5) Right-click the selected Emulator and select Cradle.
6) In addition, within ActiveSync connection settings it is necessary to allow DMA connections.

Allow DMA connections in Windows Mobile Device Center:

1) Open WMDC.
2) Click Connection Settings.
3) Click the drop down button and change it to DMA.
4) Hit OK.
5) The Emulator will automatically connect.

Pdblist (itsutils suite tool):

1) pdblist –v: lists accessible volumes, including virtual storage card of the Windows Mobile Emulator.
2) pdblist –D: list components of databases that are accessible via the emulator.
3) pdblist –d: dump a particular object/file by name.

Tools Used

Cellebrite UFED Physical Pro 1.1.9.7:
http://www.cellebrite.com/forensic-products/forensic-products.html?loc=seg

Windows Mobile Device Center (WMDC) 6.1.6965:
http://www.microsoft.com/download/en/details.aspx?id=3182

Raptor 2.0:
http://forwarddiscovery.com/Raptor

Mobile Internal Acquisition Tool (MIAT) 1.0:
http://miatforensics.org/

FTK Imager 3.0.1.1467:
http://accessdata.com/support/adownloads#FTKImager

EnCase v6.19:
http://www.guidancesoftware.com/

Microsoft Device Emulator 3.0 (x86):
http://www.microsoft.com/en-us/download/details.aspx?id=5352

Windows Mobile 6.1.4 Emulator Image (Professional Version):
http://www.microsoft.com/en-us/download/details.aspx?id=9263

Microsoft Visual Studio 2008 (90 day trial .iso file):
http://www.microsoft.com/en-us/download/details.aspx?id=3713

Daemon Tools Lite v4.45.4.0315 (to istall Microsoft Visual Studio 2008):
http://www.daemon-tools.cc/eng/products/dtLite

Itsutils:
http://itsme.home.xs4all.nl/projects/xda/tools.html

References

AccessData. (2011). MPE+ Mobile Phone Forensics. Retrieved from http://accessdata.com/products/computer-forensics/mobile-phone-examiner

Casey, E. (2009, October 22). Recovering Deleted Text Messages from Windows Mobile Devices. Retrieved from http://computer-forensics.sans.org/blog/2009/10/22/recovering-deleted-text-messages-from-windows-mobile-devices/

Casey, E. (2010, August 29). Advances in Windows Mobile Forensics. Retrieved from http://blog.cmdlabs.com/2010/08/29/advances-in-windows-mobile-forensics/

HTC. (n.d.). Support - HTC Fuze. Retrieved from http://www.htc.com/us/support/htc-fuze-att/tech-specs

University of Rome. (n.d.). MIAT - Home Page. Retrieved from http://www.miatforensics.org/

http://accessdata.com/WEBINAR/Windows_Mobile_Image.mp4