

FTK Imager Step by Step

FTK IMAGER AS AN ARCHIVER?

HOW TO INVESTIGATE FILES WITH
FTK IMAGER

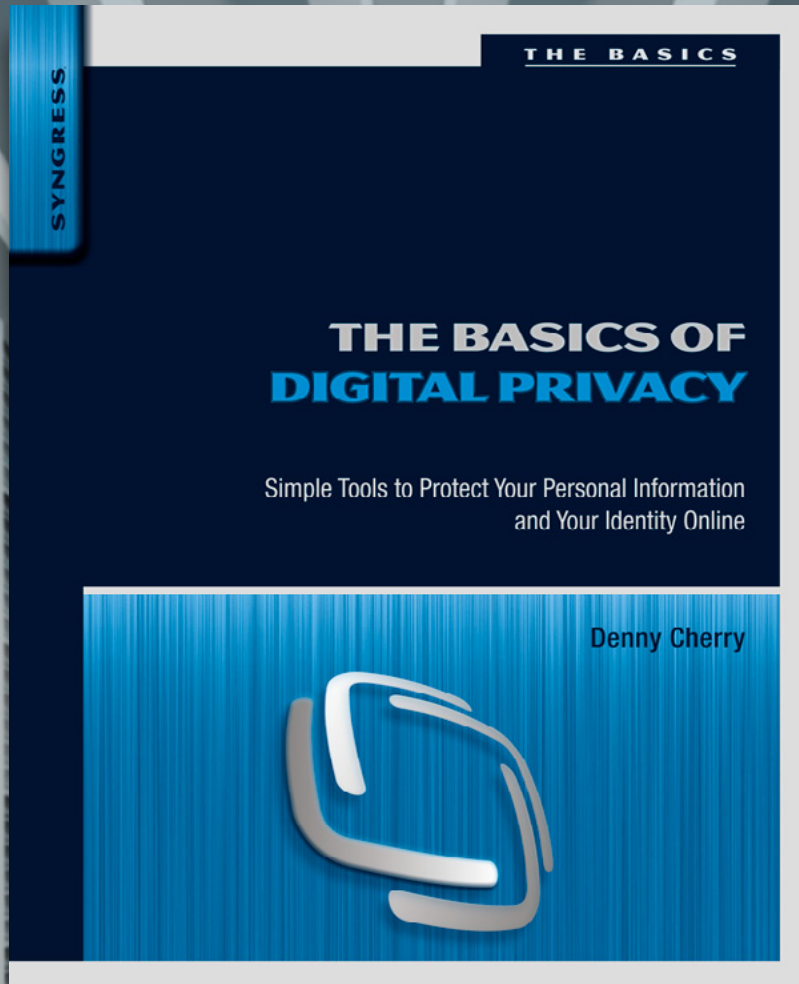
CREATING A FORENSIC IMAGE
OF A HARD DRIVE

DETECTING EVIDENCE OF
INTELLECTUAL PROPERTY THEFT

FILE RECOVERY AND RECUVA
SOFTWARE

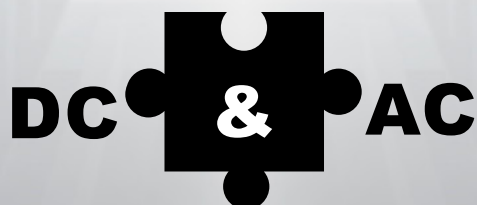
BONUS

FTK – IMPROVING PASSWORD
RECOVERY



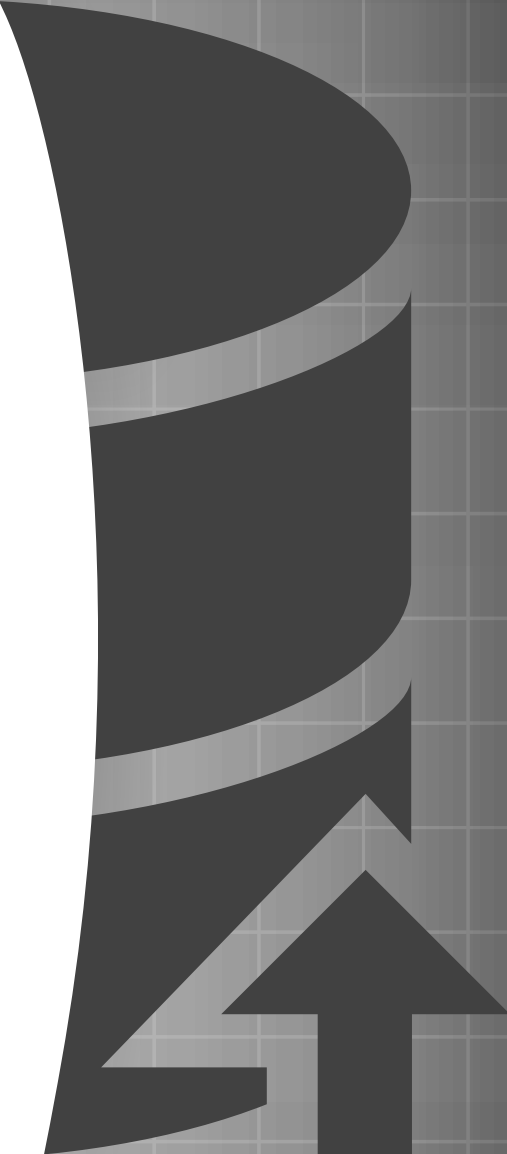
www.basicsofdigitalprivacy.com

The most straightforward and up-to-date guide to privacy for anyone who goes online for work, school, or personal use



DENNY CHERRY & ASSOCIATES CONSULTING

IS YOUR DATABASE... HEALTHY?



CRITICAL ALERT MONITORING
DISASTER RECOVERY PLANNING
SQL SERVER HEALTH CHECK
VSPHERE / HYPER-V HEALTH CHECK
STORAGE HEALTH CHECKS

AND MUCH MORE



WWW.DCAC.CO

Editor:

Agata Wieliczko
agata.wieliczko@software.com.pl

Betatesters/Proofreaders:

Olivier Caleff, Johan Scholtz,
Nicolas Villatte, Simohammed Serrhini,
Luca Losio, Robert Vanaman,
Massa Danilo, Kishore P V

Senior Consultant/Publisher:

Paweł Marciniak

CEO: Ewa Dudzic

ewa.dudzic@software.com.pl

Production Director: Andrzej Kuca

andrzej.kuca@software.com.pl

Marketing Director: Joanna Kretowicz

jaonna.kretowicz@eforensicsmag.com

Art Director: Ireneusz Pogroszewski

ireneusz.pogroszewski@software.com.pl

DTP: Ireneusz Pogroszewski

Publisher: Hakin9 Media Sp. z o.o. SK

02-676 Warszawa, ul. Postępu 17D

Phone: 1 917 338 3631

www.eforensicsmag.com

DISCLAIMER!

*The techniques described in our articles
may only be used in private, local net-
works. The editors hold no responsibility
for misuse of the presented techniques or
consequent data loss.*

Dear Readers,

Welcome to our new issue devoted to Forensics with FTK Imager. I want to invite you on a journey exploring the powerful features of this forensic toolkit. Our experienced experts prepared specialistic as well as introductory articles, that will satisfy you with the comprehensive knowledge.

This Access Data tool which is available free of charge has been used to institute forensic pictures of any kind of hard drives and disks as well as single files. It is also capable of recovering data so hackers – beware, your activity is traced!

In this issue you will learn to catch digital evidence ‘on the fly’, how FTK Imager can be helpful in archiving numerous data, how to recover deleted evidence files. In addition to this we present 2 tutorials of data recovery with FTK Imager in combination with two other softwares and a few real case examples where that Access-Data tool became crucial to conduct the investigation.

Enjoy reading and join us in discovering digital forensics! Please give us your feedback, follow us on Facebook. Google + and Twitter and be updated with our latest news and suggest future topics. We will be happy to hear from you!

eForensics Magazine Team

FTK IMAGER, MORE THAN AN JUST AN IMAGER

by Sam Pepenella

Although a considerable amount of investigators utilize FTK Imager as an imager and preview tool, it has many more capabilities which could assist any investigator during the examination of some digital devices. Have a look at some of the features of FTK Imager and get some tips how to simplify future investigations!

08

FTK IMAGER AS AN ARCHIVER?

by Keith Swanson

One of the trials and tribulations of Digital Forensics is what to do when everything when you're done. If you have worked for many years, you have folder after folder of data. Reports, images, exports, everything and anything can be involved in a case. All of it has to be put away nicely, in a manner that someone years from now can open and see what you have done. Save your time and money by learning how to archive with FTK Imager!

16

HOW TO INVESTIGATE FILES WITH FTK IMAGER

by Mark Stam

The Master File Table or MFT can be considered one of the most important files in the NTFS file system, as it keeps records of all files in a volume, the physical location of the files on the drive and file metadata. One of the most important tasks of a computer forensics expert is making file artifacts and metadata visible. Learn how in a straightforward manner, conduct the process of extracting NTFS file system data from a physical device. NTFS uses the Master File Table (MFT) as a database to keep track of files. We can use the MFT to investigate data and find detailed information about files. In this example we use FTK Imager 3.1.4.6 to find a picture (JPEG file) in Windows 7.

22

USING FTK IMAGER CREATE FORENSICALLY-SOUND COPIES OF DIGITAL MEDIA

by Austin Troxell

The first step in Digital Forensic examinations is to create precise duplicates of any storage media collected as potential evidence. One of the key principles of Digital Forensics is that examiners must eliminate or minimize the risk of altering any information contained on the original evidence items. Where at all possible, the analyst will make digital copies of the media to be examined and work from these duplicates, preserving the originals. The Digital Forensics examiner has numerous options for creating exact bit-stream representations of digital media, including hardware duplicators as well as various software tools that create digitally identical copies. In this article Austin Troxell focuses on the features and use of AccessData's FTK Imager.

30

CREATING A FORENSIC IMAGE OF A HARD DRIVE USING FTK IMAGER AND IMAGER-LITE FROM ACCESSDATA

by Bridgette Braxton

The advancement in the world of computer forensics has provided many tools to assist incident responders perform live analysis on a computer. The capabilities of forensics tools have improved by making analysis feasible by integrating enhanced interfaces, documentation, built-in detection methods, and new ways to collect evidence. Let's see how FTK Imager can be used in those processes and how to do it!

36

FTK IMAGER ON THE FLY

by Robert C. DeCicco

Practicing computer forensics often times means having to jump on a plane or in a car to get someplace quickly to collect evidence. In part, response to the often reactive nature of the work, agencies and firms have developed fly away kits, mobile labs or other solutions that are prepped and ready to go and can handle a variety of environments or evidence types. What about when you're not prepared for a collection? What about those instances where you may be only scheduled to attend a meeting or scoping exercise at a client site? Robert DeCicco will show you how FTK Imager literally saved the day when the circumstances suddenly changed.

48

HIDING INFORMATION THEFT: HOW TO FIND EVIDENCE OF DATA THEFT

by Mark Garnett

It is fair to say that most of today's computer users know that when they "delete" a file from a computer system, it is not really deleted. They have the repository of all answers, Google, available to

54

them that they can use to research ways in which to cover their tracks and prevent computer examiners from finding evidence that may get left behind from any wrongdoing. However, Mark Garrett will show you that even this might not be so successful if the forensic examiner knows how to use FTK Imager!

Discover how he investigated the real case of stole intellectual property and learn how to do it!

62

DETECTING EVIDENCE OF INTELLECTUAL PROPERTY THEFT USING FTK® IMAGER (AND FTK® IMAGER LITE)

by Ana M. San Luis and Robert K. Johnson

In today's world of constantly evolving technology, there arise a number of options for thieves, embittered and disgruntled employees, or naive colleagues to participate in the theft of intellectual property, whether intentional or otherwise. IP theft can cost victims their jobs, reputations, and even millions of dollars, depending on what is stolen. Experts and investigators have a number of industry and court accepted tools available at their fingertips to investigate suspicions or allegations of IP theft. Some of these tools allow forensic experts and investigators to examine live running suspect machines or media, while making little to no changes to the suspect machines or media. Two such tools are AccessData®'s FTK® Imager and FTK® Imager Lite.

72

FILE RECOVERY – PART 01

by Everson Probst

One of the core activities of a computer forensic expert is the file recovery. Through recovering, it is possible to examine records deleted by users or deleted automatically by the system. This tutorial will show you how to recover files as well as the technical properties performed with FTK Imager and Recuva software. Recuva is the free software distributed by Piriform whose main function is to recover deleted files. It uses the archive system index to recover deleted files and also runs Data Carver, but in this aspect, it is not very efficient when compared to Foremost.

78

FILE RECOVERY – PART 02

by Everson Probst

In this tutorial you will learn how to conduct file recovery with FTK Imager and Foremost software. Foremost is the free software that has the function of recovering files based on the Data Carver method. It is capable of recovering files whose record entries are no longer found in the archive system. That makes it a very useful tool to recover older files, despite it is not capable of recovering all original properties of the recovered file.

88

THE OTHERFTK!: FORENSICSTHAT KONVICT!

by Christopher M. Erb

Recently released studies have shown 93% of criminal and civil cases in the United States involve some type of digital evidence. Large capacity storage media containing massive amounts of digital evidence and constant changes in newly released software continue to bring challenges to digital forensics. That being said, computer forensic examiners are regularly tapped to process and examine vast volumes of data while removing superfluous rubbish. Today, computer forensic examiners are fortunate enough to have a host of forensic software and hardware products available to them and their respective agencies / corporations. This article discusses the best practices to preserve, examine and report the results of a digital forensic examination with the use of FTK.

102

FTK – IMPROVING PASSWORD RECOVERY

by Brian Mork

Let's be honest: in our day to day forensics work it is far more likely for us to encounter a user who has saved all of their passwords in a text file than anyone outside the forensic realm would ever guess. If a suspect hasn't written down their password it is likely as not to be along the lines of "password" or "123456." On those rare occasions when a "complex" password is chosen it will often conform to the pattern of "word from a dictionary with a capital first letter, followed by a single number or special character." Those of us who are lucky enough not to spend our days working organized crime cases will find the case where we have to recover a password of any real complexity to be the needle in the haystack.

Developing for Amazon Web Services?

Attend Cloud DevCon!



June 23-25, 2014

San Francisco

Hyatt Regency Burlingame

www.CloudDevCon.net



Attend Cloud DevCon to get practical training in AWS technologies

- Develop and deploy applications to Amazon's cloud
- Master AWS services such as Management Console, Elastic Beanstalk, OpsWorks, CloudFormation and more!
- Learn how to integrate technologies and languages to leverage the cost savings of cloud computing with the systems you already have
- Take your AWS knowledge to the next level – choose from **more than 55 tutorials and classes**, and put together your own custom program!
- Improve your own skills and your marketability as an AWS expert
- Discover HOW to better leverage AWS to help your organization today

Register Early
and SAVE!

A BZ Media Event

CloudDevCon



Amazon Web Services and AWS are trademarks of Amazon.com, Inc.

FTK IMAGER, MORE THAN AN JUST AN IMAGER

by Sam Pepenella

Over the last several years technology has grown in leaps and bounds around the world. It is not uncommon these days to see someone with digital devices, while not too long ago, only a select few had devices. Cell phones, tablets and computers have become as common as toasters in the home. If you've traveled by air within the last 5 years and looked around, you may have noticed how many people waiting for their flight were logged into the airports Wi-Fi to check their email, stock quotes, check their Facebook page or just the Internet.

What you will learn:

- certain uses of FTK Imager
- how to mount image files
- detecting encryption
- how to check a suspect's computer for viruses and malware
- how to image a file

What you should know:

- basic to intermediate forensics experience
- understanding of Windows Registry
- how to use Forensics Toolkit
- properly collect and triage evidence

Along with the explosion of the use of digital devices, the need to process these devices is also on the rise. Whether it is a criminal investigation, discovery for litigation or to foil a terrorist attack, the need for examiners is paramount.

As an examiner I'm part of a select group of individuals who have a very special skill set, a skill which is growing in popularity because of the demand for examinations. While I feel privileged to be among this group of professionals, I have a driving curiosity to know what tools others are using. I feel fortunate to network with examiners around the globe.

I often speak with other examiners and at times the topic of our conversation is what tools they are using to complete their examinations. I usually get a broad range of responses ranging from open source to commercial products. Even though I receive a wide range of responses, one tool often designated as an important part of their investigation to image their devices the Access Data's FTK Imager.

FTK Imager is a simple, but very powerful tool for any examiner who is involved with doing digital forensics. FTK Imager is more than just an imager, it holds several other features which are indispensable to any examiner. It is a tool used to preview data and perform imaging of digital devices and if utilized

properly could limit the time an investigator dedicates to a case. Because of its versatility, an investigator can determine if further investigation is needed. Some examples of its uses are as follows:

- Create forensics images
- Preview files and folders
- Mount images
- Export files and folders
- Recover deleted files from the Recycle Bin
- Create hash of files

Although a considerable amount of investigators utilize FTK imager as an imager and preview tool, it has many more capabilities which could assist any investigator during the examination of some digital devices.

Let's look at some of the features FTK Imager has, and for this writing I utilized Access Data's latest release of FTK Imager, v3.1.4.6. One thing to keep in mind while reading this article, is it is not all inclusive to what FTK Imager can do, I will only show some tips which can be utilized to help you simplify future investigations.

FTK IMAGER TOOL BAR

Once you launch FTK Imager one of the things you will notice is its simple layout. As you can see from Figure 1 below, FTK Imager has the icon view on by default. Examiners can select to have them present or not, simply by selecting "view" and check or uncheck "tool bar." As an examiner, I would rather have the icons displayed, but that is my personal preference.



Figure 1. FTK Imager tool bar

A nice feature AccessData has made is how the tool bar is arranged. As stated above, having access to each feature quickly is a great benefit to any examiner.

Adding a device for examination is quick and simple. An investigator would need to select the "Add Evidence Item" Icon by either selecting to either add one item or all the devices on a computer (Figure 2), something beneficial when doing a live examination of a suspect's computer.



Figure 2. Adding device for examination

FTK Imager allows an investigator to add four types of evidence sources for preview, such as a Physical Drive, Logical Drive, Image File or Contents of a Folder. First I'll add a physical Device by either selecting the "Add Evidence Item" (left icon above) or "Add all Attached Devices" (icon above).

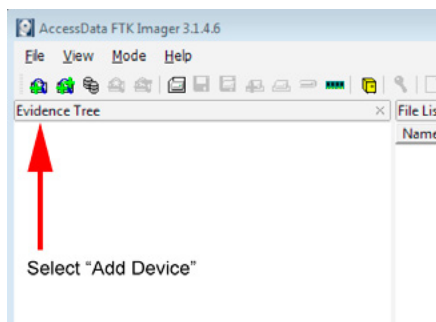


Figure 3. Selected „Add Device“ icon

ADDING A DEVICE

Once you select the “Add Device” icon (Figure 3), you are presented with the “Select Source” box (Figure 4), asking to select an evidence file to add to FTK Imager.

Looking at the four options, an investigator can select to add a source depending on the type of case they are working or limitations set by the courts. Adding a physical drive will add the entire contents of a physical drive, to include all partitions associated with that device. Adding a logical source will only load the contents of a logical drive including unallocated space. When you add an image file, this will load the image file as it was created, either Raw (dd), SMART (S01), E01 or AFF. Lastly, you can add an individual folder for an analysis.

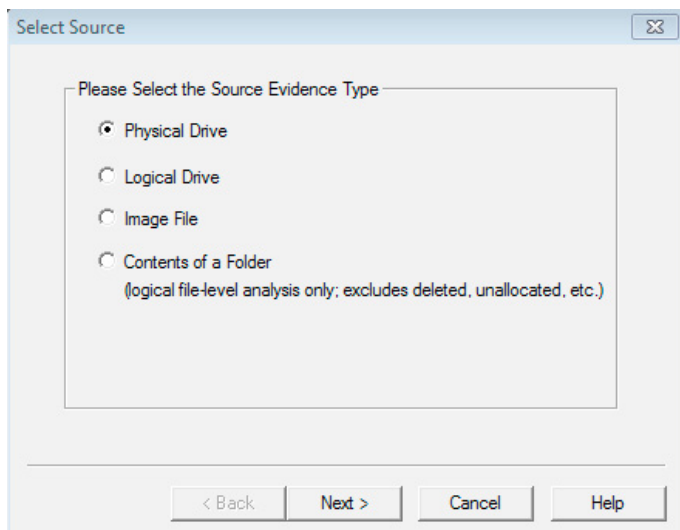


Figure 4. Select Source

After you have selected the type of evidence for your case, “Physical Drive”, presented with the “Source Drive Selection.” From here, you will see what devices are attached to your machine and an investigator can select what device best fits their investigation.

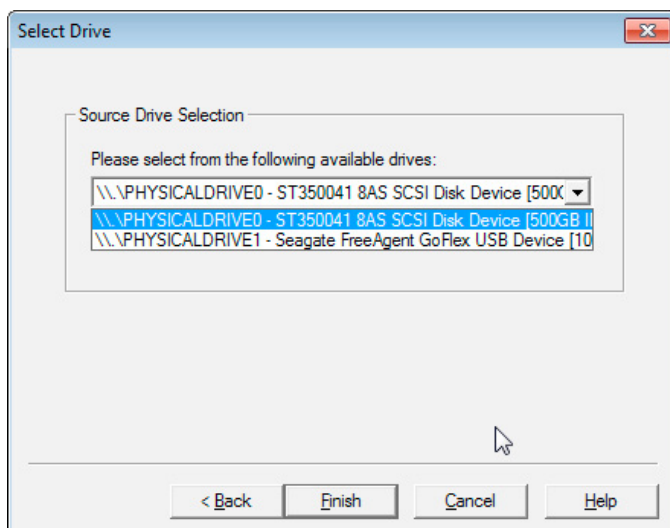


Figure 5. Select Device section

As you can see (Figure 5), I have two attached device selections I can choose from. One of my devices is my computer’s hard drive and another attached device. Once the device is selected, FTK will populate the “Evidence Tree” in the program so an investigator can begin to do a preview of the file and folder structure of the subject(s) computer. This will give an investigator an idea of what type of file system they are working with and can plan their investigation accordingly.

As you can see in the Evidence Tree (Figure 6), FTK Imager was able to see my computers physical hard drive and determine I have 3 partitions. This is beneficial for an investigator so they can determine how the hard drive is partitioned and can verify, by adding up the partitions, to see if there are any hidden partitions.

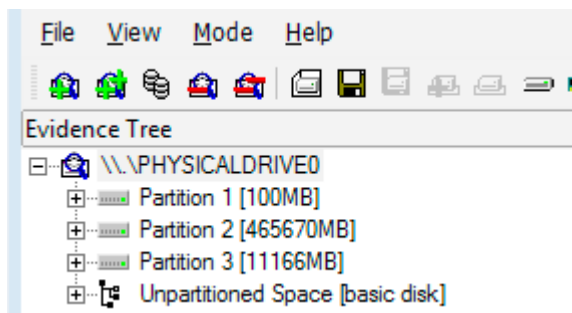


Figure 6. Evidence Tree

MOUNTING AN IMAGE FILE

An important feature of FTK Imager, is its image mounting function. The image mounting function allows the investigator to mount foreign file systems to windows, run Antivirus software, make virtual writes using a cache file, run 3rd party software against an image and navigate directory structure in Read only mode.



Figure 7. Mounting Icon

Looking at Figure 7, you can see Access Data put the Image Mounting Icon into the tool bar. This makes quick access for anyone wanting to mount an image file to drive. As an investigator, besides adding evidence to a case, I often use this feature. With nearly every case I examine, I utilize this function to mount the image and run a anti-virus and malware program against the image. If you have done computer forensics long enough, you will know that attorneys like to claim “a virus put all those illegal images on my clients’ computer.” It is my practice to take away any chance the defense can claim this as a defense, by doing a thorough and complete investigation. Running a virus and malware program against an image file should be good practice for any investigator, to close any chance of putting reasonable doubt into an investigation.

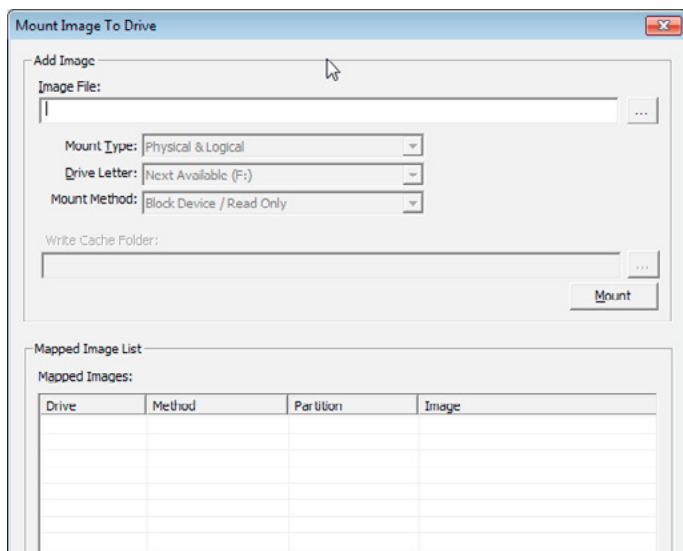


Figure 8. New windows appearing

After selecting the “image mounting” option, you will see a new window will appear (Figure 8). From here, an investigator can direct what file to load by selecting the radio button to the right of the top line. For me, I will point it to a previously created image file, in most cases, an Expert Witness File (E01). Below the box titled “Image File”, FTK Imager allows me to select a set of parameters on what special functions I would like to use. Such features as Physical & Logical, Physical only or Logical only. I can have the option to select the drive letter and specify Mount Method settings, in the event I want to see what deleting a file will do in any investigation, I can select “Block Device/Writeable”, which will allow me to delete files from the file structure in a virtual manor. With this setting, FTK Imager creates a cache file containing the information of the virtual deleted content.

If I am doing an investigation where I have an HFS system, FTK Imager allows me to also mount this type of file system. Going through the same practices as stated above for mounting an image, I would need to change the Mount Method to “File System/Read Only”, to allow FTK Imager to do the translation for me. If you do not select this setting, Windows will not translate the file structure and you will not see be able to see it.

Drive	Method	Partition	Image
PhysicalDrive3	Block Device/Read ...	Image	G:\E01\Jean\nps-2008-jean.E01
F:	Block Device/Read ...	Partition 1 [10228...	G:\E01\Jean\nps-2008-jean.E01

Figure 9. Here you can see what was mounted

After you have mounted an image, FTK Imager applies the next available letter to the drive, so it can be identified in the windows system. Looking at (Figure 9) above you can see what was mounted, showing the examiner the Drive, Method, Partition and Image. As an investigator, I know my primary focus for doing a virus and malware scan, would be the “F” drive. As a note of caution, verify the “F” drive is the correct drive. In this example, there is only one partition on the device I selected and with many large drives in circulation today, there could be multiple. If you know the size of the partition you are examining, compare the size to the drive letter or you can open Windows Computer Management or Explorer to make the correct determination.

Now that I have selected my correct image I want to utilize Windows Explorer to perform my virus and malware scan. Simply right clicking on the drive should give you the selection to choose either the virus or malware program to launch against your target drive. If that option is not available, launch your virus or malware program and select the target drive appropriately.



Figure 10. Creating an image file

CREATING AN IMAGE FILE

Now to the meat and potatoes of what FTK Imager can do, creating a bit for bit images of the original file and providing you with a verification for each image created. FTK Imager will create five types of image files, Raw (Linux DD – .001), SMART (Linux .S01), Expert Witness (Encase .E01), Advanced Forensic Format (.AFF) and Access Data Custom Content (Logical Image .AD1).

If you look at the tool bar, Access Data put the Image Icon in the reach of the examiner at the click of a mouse button (Figure 10).

By selecting this icon, you will be presented with a window which is similar to the window for the “Add Device” icon (Figure 11). Access Data also added the “Fernico Device (Multiple CD/DVD) option, (optional) to back up forensic data from a network location or from a locally attached hard drive, automatically spanning the content over a series of disks.

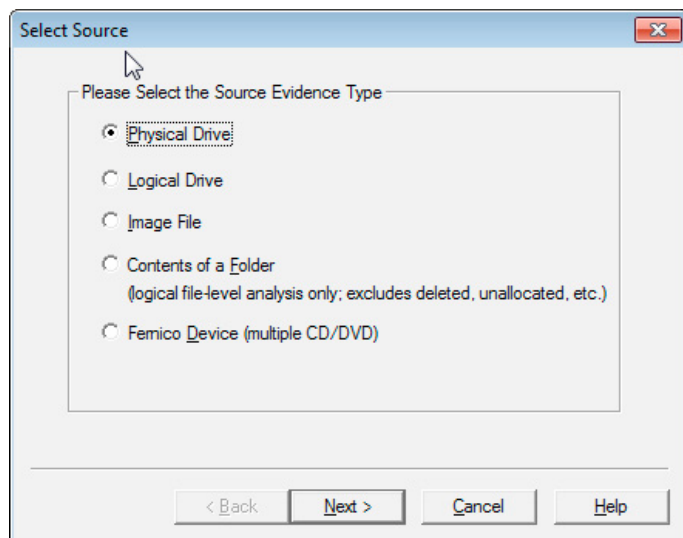


Figure 11. Watch out – similarities of two windows

One quick glance at Figure 11, and you will see what I mean about regarding the similarities with adding a device to selecting a source for imaging. Although I have never confused the two windows, it could be understandable if someone did. Starting with the top selection, “Physical Drive”, this is the one you will select if you are doing a bit for bit copy of the original file. Next is the “Logical Drive”, an option to select if you have some parameters set in place by a warrant, limitations from a server or making an image of an encrypted file system. If you have already noticed, Access Data put the option of imaging an image file. In the event you need to change an RAW DD (001) to a compressed or Encase Expert Image .E01 image, this option is available for you as well.

Depending on what analysis software you are using will depend on what image format you will utilize to add into the case. In addition the amount of storage you have will also depend on what format you use. Most examiners I speak with advise storage is a premium in their labs and they need to make good use of what they have. As a result, a majority of them utilize the Encase E01 format, which is a compressed format. There are factors which will effect just how much a file will be compressed, which is beyond the scope of this writing. Lastly, a majority of software in use, utilizes this same type of image format, nearly becoming a standard in the industry. As a Computer Forensic Examiner, you are bound to come across encrypted files at some point in time. As a Law Enforcement Officer performing computer forensics in the field, it is when you come across encrypted files and not if you come across

Three of our units last 4 investigations we have come across encrypted files on the suspect’s computer. Two were from a live system and one was from a dead box investigation. With so much information being stored on our computers these days, it is a wonder there is not more encryption being utilized.

As an investigator, whether you do forensics in the field or the offices of corporate America, understanding what data to capture before an analysis can mean a game changer in an investigation. Access Data has addressed this issue with three options available in FTK Imager. One option is the choice of “Capture Memory”, the next is “Obtain Protected Files” and the last is “Detect EFS Encryption” (Figure 12).

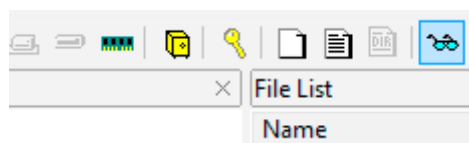


Figure 12. 3 icons of our interest

Looking at (Figure 12) above shows the 3 icons of interest. The first icon is the icon which looks like a memory module, which should be selected when capturing a running systems live memory. The next is the icon which looks like a yellow safe, which is the icon one would select to obtain protected files. Lastly, the yellow key to the right is the icon you would select to check a system for Encrypting File System (EFS).

Now, in regards to the EFS encryption, this will not only detect Windows encryption but will also recognize encryption from a third party, such as PGP, Credant, JFS, LVM and Ultimaco to name a few.

These three different options in a sense work hand in hand together. Imagine you execute a search warrant and the basis of the search warrant is the transmission of illicit material. As you come across the suspect's computer system, you notice the system is running (aka powered on). Do you pull the plug, a procedure that has been taught for several years, or do you perform a live analysis.

Every situation is different, but in the type of situation as explained above, is something our unit has had to deal with. Using a lighter version of FTK Imager called FTK Lite, you are able to perform many of the features one can perform in a lab environment with FTK Imager but through the use of a flash drive.

CAPTURING LIVE MEMORY

Something which should be noted is using a flash drive will leave a footprint on the suspect's computer, so good documentation is paramount.

Once you have mounted the flash drive into the suspect's computer, select the Capture Memory Icon. When you select the memory module, you will be prompted with a screen as Figure 13.

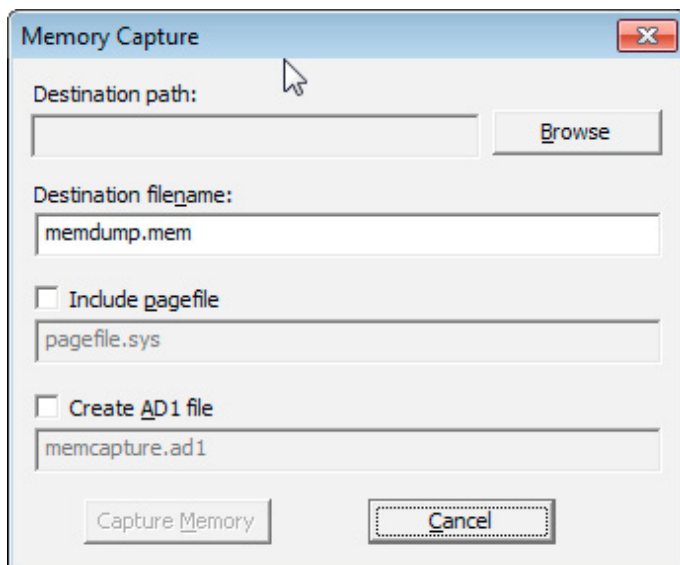


Figure 13. Capturing live memory

From here, a user will need to set the Destination path where you would want the “memdump.mem” file to be saved. One should note, if you are only using a flash drive to capture the data from a running computer, ensure you have a large enough flash drive to hold FTK Imager Lite and the storage for the memory dump file. If a computer has 8 GB of memory, ensure you have at least that much drive space available to you when performing this procedure. Included as options is the Pagefile and the Create AD1 file.

OBTAINING PROTECTED FILES

The next feature is the use of the obtaining protected files, such as System Files (Figure 14). Select the Icon which looks like a yellow safe and you are presented with a box as such. Access Data placed a warning box into this feature to warn the examiner the information they are obtaining is from a “live system only.” If you select this option from your lab, you will collect the live data from your forensic computer. This option is made to do live capturing of system files from a suspect's computer.

Once you have selected the destination folder for the captured data, you have the option of collecting the files needed for password recovery or password recovery plus all registry files. From my experience, I'm going to collect password and all registry files, which could save you time when putting that information into Password Recovery Tool Kit (PRTK).

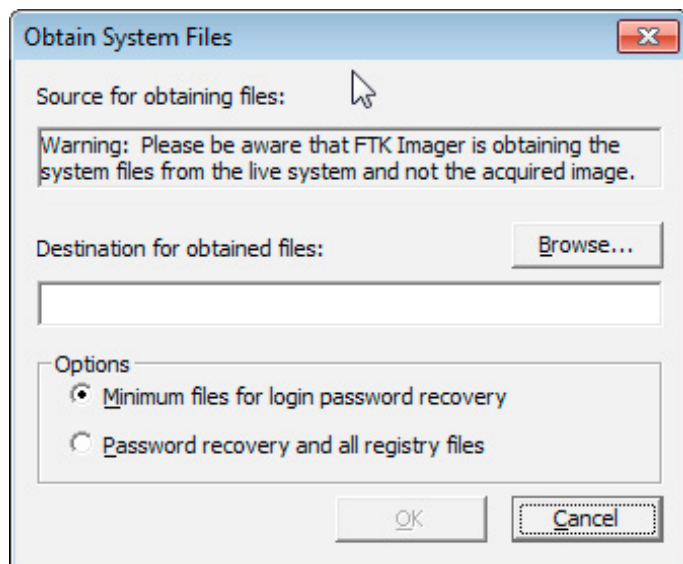


Figure 14. System Files

DETECTING ENCRYPTED FILE

Lastly is the option of checking the suspect's computer for Encrypted Files. The last Icon of the three is an icon which looks like a yellow key. When checking a system for encrypted files, you can only do so when the computer you are examining is added as a device, through the device option. After the device has been added, you will notice the yellow key goes from being grayed out to being highlighted.

When you select the EFS icon, you are presented with the following window (Figure 15). FTK Imager will now process the device you have and if it does locate an encrypted file, the program will notify you that it has located an encrypted file.

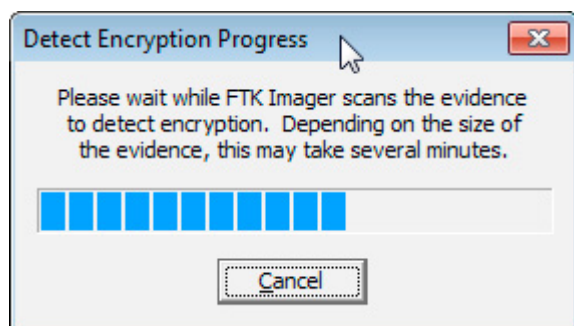


Figure 15. Detect Encryption Process

Although I have validated it with EFS, it does not cover all encryption types so a manual search may need to be done as well. As a suggestion, an investigator should check the system tray and Task Manager to see what programs and processes are running. Make sure you are able to account for the files running, because once power is lost and you do not have the live memory to recover the passwords, access to the files and folders may be permanently lost.

In this article I have only touched on a few of the capabilities FTK Imager. I would recommend one to read the Access Data FTK Imager user's manual to be more familiar with all the capabilities this tool. In this world known as the digital age, an investigator needs to have many tools available to them when performing an examination.

BIBLIOGRAPHY

Access Data. (2012, March 21st). FTK Imager Users Guide.

ABOUT THE AUTHOR

Sam Peppenella has been in law enforcement for nearly 27 years. He has worked in several facets to include patrol, secondary victims unit, community policing, economic crimes, intelligence and computer forensics/cyber crimes. Sam is credited for creating a technically advanced computer forensics capabilities for his department, which is recognized as a state of the art facility in the Tampa Bay area. Sam has taken several courses as it relates to digital investigation and holds an ACE and A+ certification. Additionally Sam is a member of task forces for both state and federal agencies as it relates to computer forensics and is an instructor of computer forensics.

FTK IMAGER AS AN ARCHIVER?

by Keith Swanson

One of the trials and tribulations of Digital Forensics is what to do with everything when you're done. If you're like me you have folder after folder of data. Reports, images, exports, everything and anything can be involved in a case. All of it has to be put away nicely, in a manner that someone years from now can open and see what you have done.

Many labs have policies for a hierarchy of folders and how to set them up. I'm not going to change that, but I may save you from spending a ton of money on some proprietary software to archive your case.

One of the greatest assets of FTK Imager is the flexibility of it, and of course it is free. One component allows you to add the contents of a folder. In my case my evidence folders look like this:

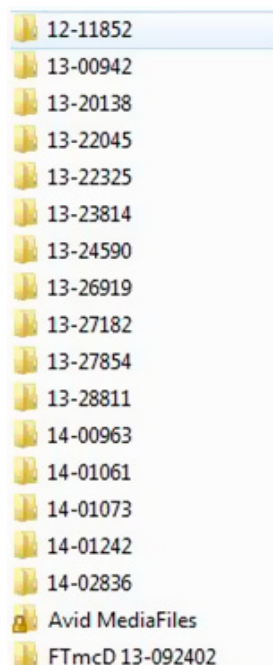


Figure 1. Case Folders

With each case maintaining folders inside it with items, exports, etc.

In my 10 years in this business I have seen a time when an administrator runs a duplicate folder and file finder on an archive. In these folders are duplicate names from other folders, and possibly duplicate case names as each of examiners may have worked on the same case. The admin deletes duplicates based on name, not content and now work is lost. This may happen even out of control of your lab, such as with your organizations IT services. No matter the culprit, this is not something conducive to forensic science.

By using FTK Imager to wrap the contents of each folder into a specified .AD1 file the contents are protected and can then be re-opened in Imager when needed and exported.

An added advantage is a small compression savings in storage space. The .AD1 compression algorithm has been defended by Access Data and accepted by the forensic community as an acceptable standard for the compression of drive image data. Therefore, using to compress and store your case work is also acceptable. Storage space savings add up over time as cases are archived in this matter, basically the volume of savings will add up to pay for more storage volumes in the future.

SO LET'S GET TO THE PROCESS SHALL WE?

First download and install FTK Imager from Access Data's website. Simple and easy to use the interface is pretty intuitive:

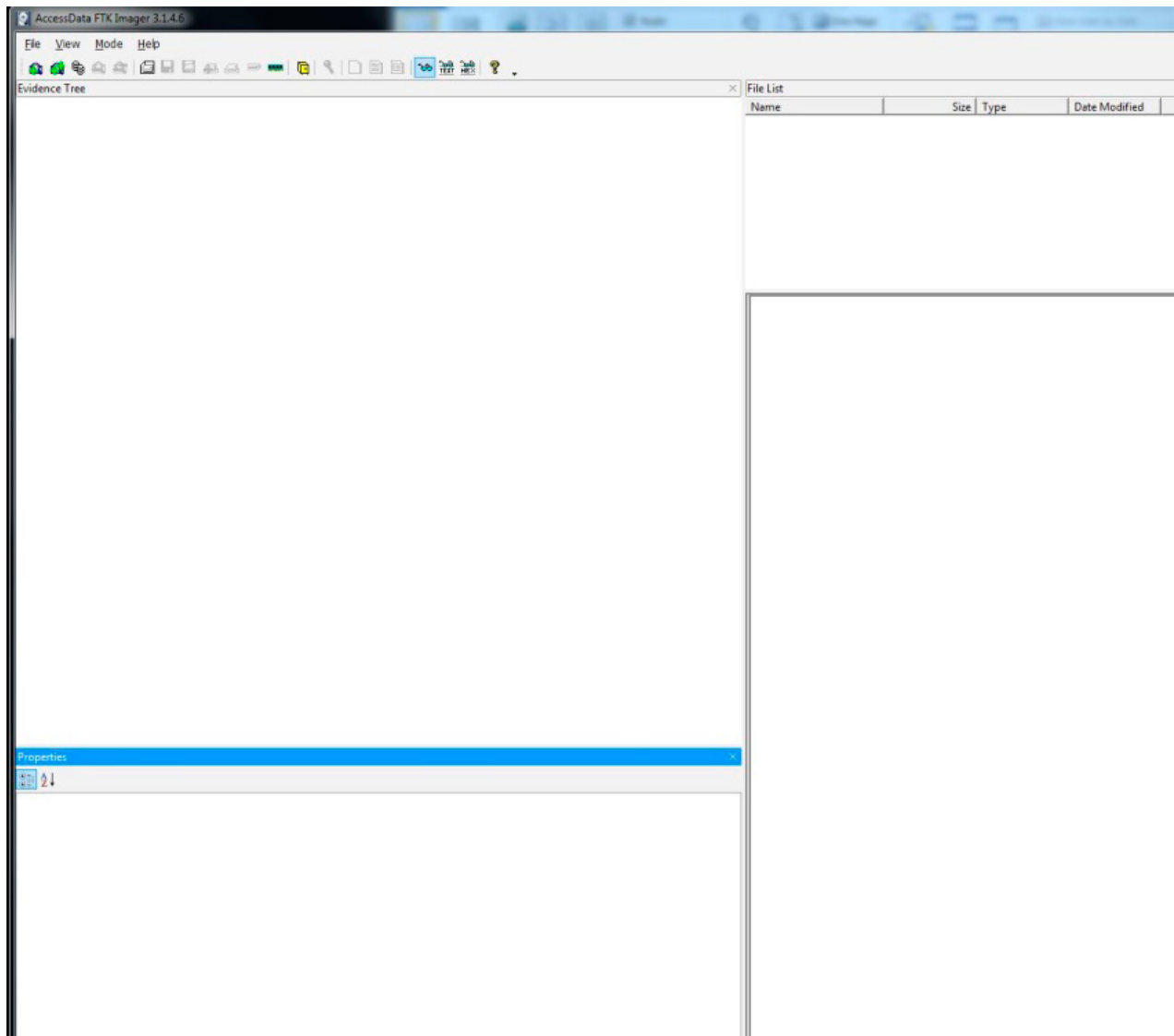


Figure 2. *Imager Interface*

Using the “Create Disk Image” option to bring up the following:

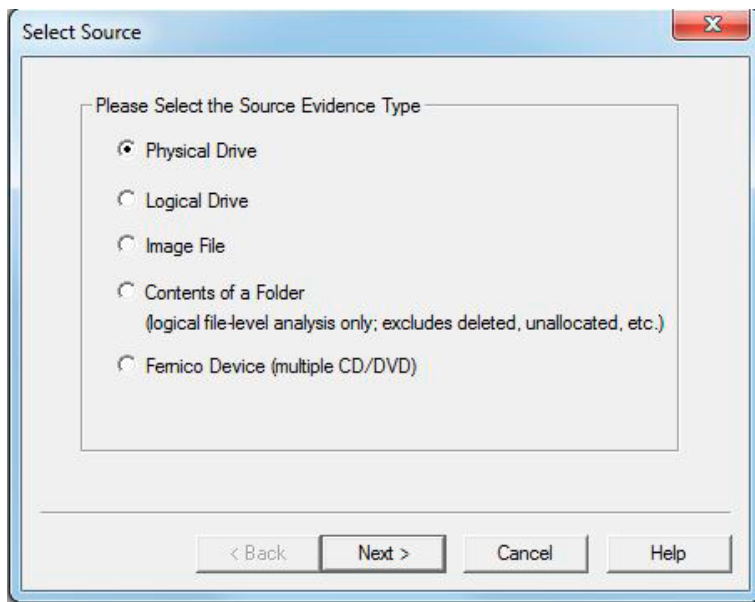


Figure 3. *Imager Interface*

We want to use the “Contents of a Folder” option. This will get us the following warning:

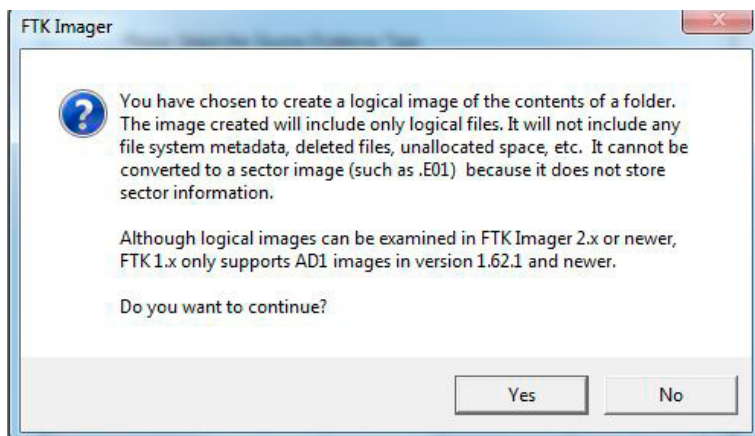


Figure 4. *Warning*

Hit “Yes” and continue:

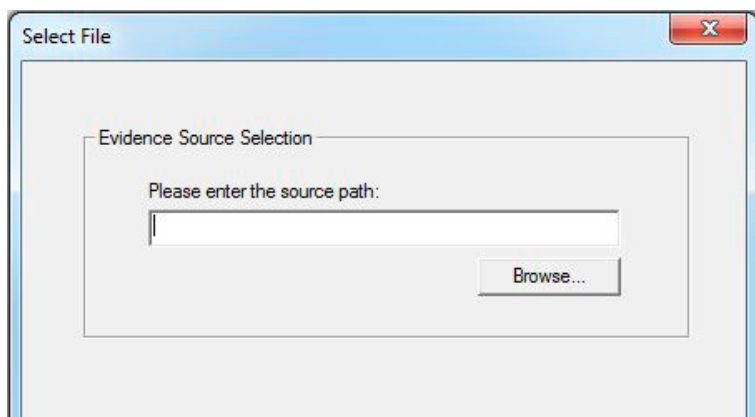


Figure 5. *Path Selection*

Select your path and folder and it appears like this:

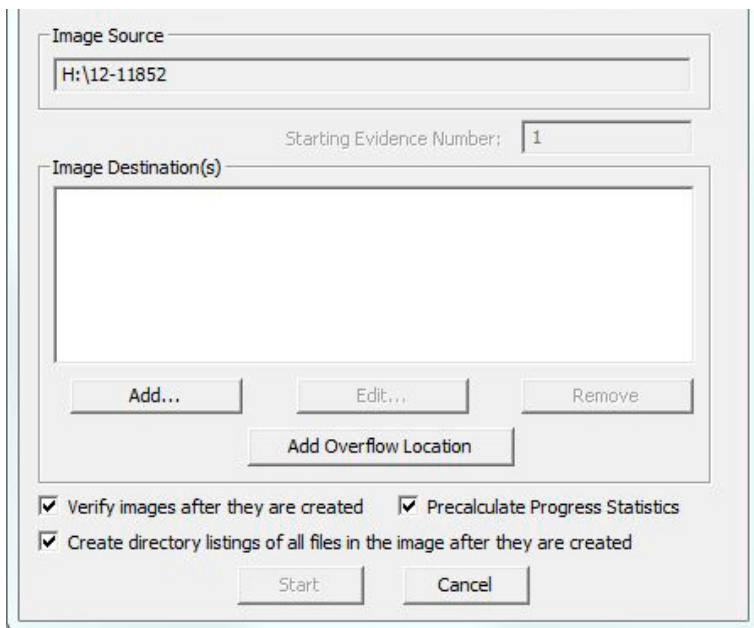


Figure 6. Image Addition

Tell FTK where to put the output and add the requisite case information:

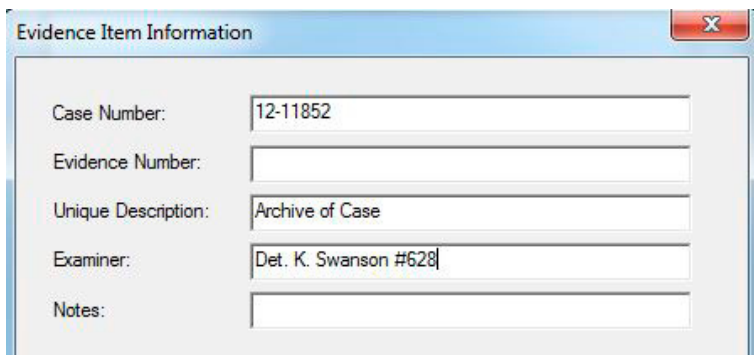


Figure 7. Information

You can even encrypt the case if you wish:

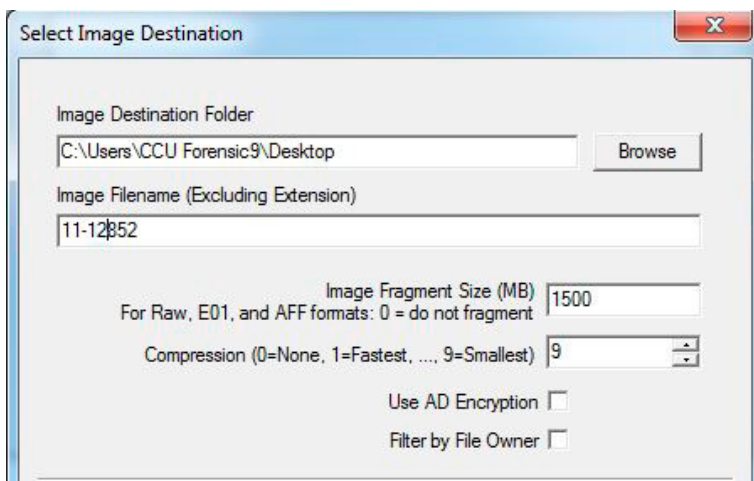


Figure 8. Finish line

Remember to make sure your fragment size is larger than the case itself. If you let the case fragment you will have a few files for the case instead of one. FTK Imager will fragment the folder into the MB size you specify, which negates the purpose of this method!

Just like making any image with FTK Imager you will see the same boxes and progress windows. The difference is that you will be making an .AD1 image of the case folder.

CONCLUSION

What you have done is created a disk image of the contents of your folder, into a single file. This protects the contents in a single file, not a folder, which can then be stored in archive. One of the benefits of using this method is that you can combine the work of several examiners into the same image, protect that work, and then make the search for the archive simpler. For example, if three examiners work on the same case, one does hard drives, one phones, and one video, all the work is stored in a single file. Searching the archive for the single archive file yields one result, once the file is opened in Imager the necessary information can be extracted.

This type of “flat” archive makes the search easier and quicker.

Also the transport and sharing of the work is far more secure with the encryption option in Imager. Discovery, Peer Review, and file review can all use the encrypted image of your folder to insure that only the right people see your work.

The .AD1 format is recognized by almost all Forensic Suites and FTK Imager will be around well past any career.

So there you have it, a simple easy way to archive your entire case in a single piece of free software. Forensic Archiving products can be priced at a couple hundred bucks, to thousands. Why? This process will work every time, with everything in the folder. *Efficient and effective, this process will allow you to archive your work safely and easily. With FTK Imager being free it also makes the budget happy. All in all this process is a win for the labs using it.*

ABOUT THE AUTHOR



Keith Swanson is currently a Detective assigned to the Computer Crimes Unit of the Scottsdale, Arizona Police Department. A 22-year veteran, He is also an Access Data Certified Examiner and has over 700 hours of Computer Forensic and Mobile Phone Forensic training. Detective Swanson holds a Masters Degree in Information Management and has been involved with Digital Forensics for over 10 years and has consulted for the last 3 years on software development.

In the field of IT security consulting and penetration testing we are the market leader in Germany.

SySS, established in 1998, advises numerous companies in a national and international context.

A large number of satisfied customers, live hacking events as well as fairs have established our role as a demanded IT company.

The following are major areas of SySS:

- **Penetration Testing**
- **Trainings**
- **Live Hacking**
- **IT Forensics**

You are looking for more than just a new working environment?

At SySS, you have the possibility to give your passion room in an experienced but young and still expanding team.

When you are facing difficulties you say „bring it on!“ and start being creative to solve the situation? And above all, you have team spirit? Excellent, because **currently we need people** in the following areas of our company in Tübingen/Germany:

- **Penetration-Testing**
- **IT Forensics**



SySS. The PenTest Experts.

HOW TO INVESTIGATE FILES WITH FTK IMAGER

by Mark Stam

The Master File Table or MFT can be considered one of the most important files in the NTFS file system, as it keeps records of all files in a volume, the physical location of the files on the drive and file metadata. One of the most important tasks of a computer forensics expert is making file artifacts and metadata visible.

What you will learn:

- How to locate file artifacts and metadata within the Master File Table
- How to recover file data with FTK Imager

What you should know:

- Familiarity with the normal layout of a Windows File System

This article describes, in a straightforward manner, the process of extracting NTFS file system data from a physical device. NTFS uses the Master File Table (MFT) as a database to keep track of files. We can use the MFT to investigate data and find detailed information about files. In this example I use FTK Imager 3.1.4.6 to find a picture (JPEG file) in Windows 7.

STARTING FTK IMAGER

Open the Physical Drive of my computer in FTK Imager. The contents of the Physical Drive appear in the Evidence Tree Pane. Click the root of the file system and several files are listed in the File List Pane, notice the MFT. Click this file to show the contents in the Viewer Pane.

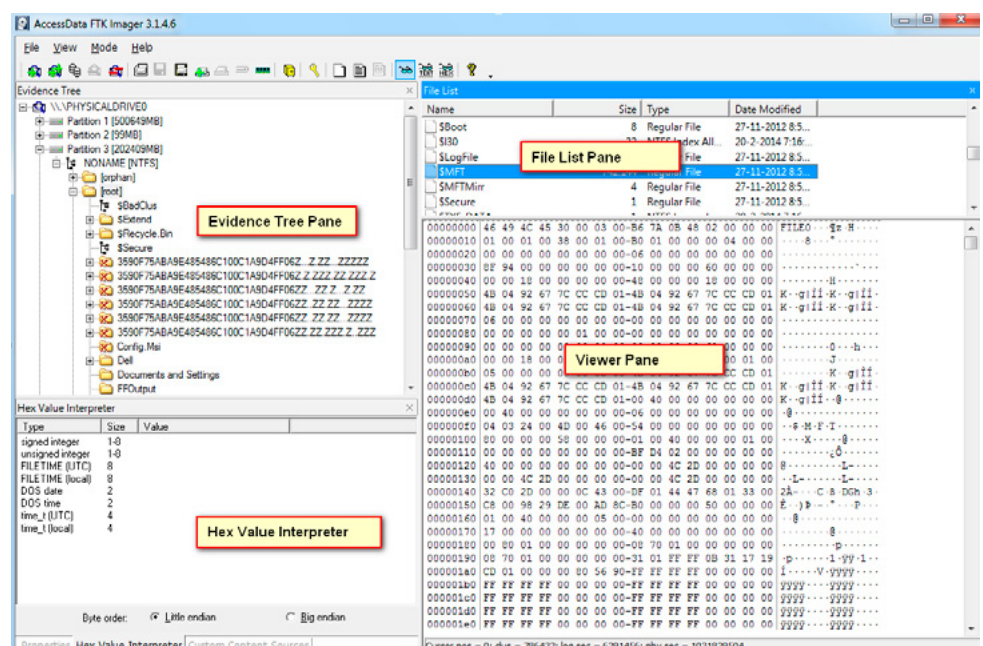


Figure 1. FTK Imager Panes

SEARCH FOR INTERESTING FILES

Click the Viewer Pane and press the CTRL + F keys to open up the Find function. Search for pictures and perhaps decide to enter the common term “IMG”.

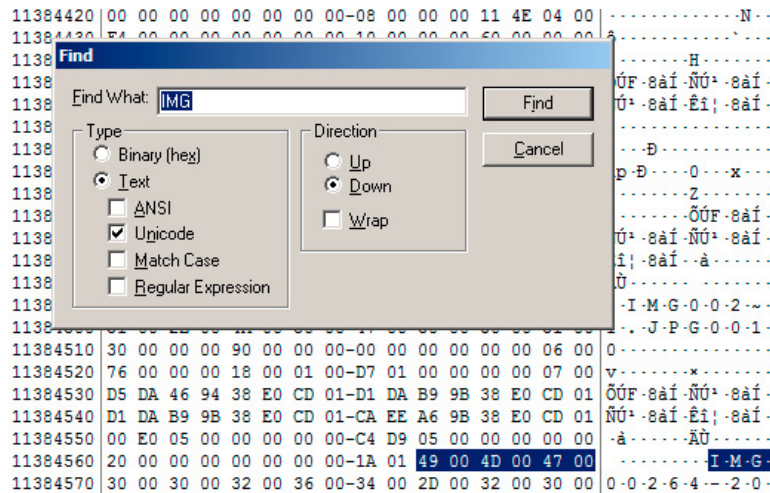


Figure 2. Search for file artifacts in the MFT (FTK)

In a short while FTK Imager finds a result. In this case, the search hit belongs to a file named IMG00264_20100109-1450.jpg. This JPEG file has more information, for instance; each MFT record has a record header, FILE0, also known as magic marker. Carefully consider the options as this magic marker is some lines *above* the search hit.

CREATION TIME

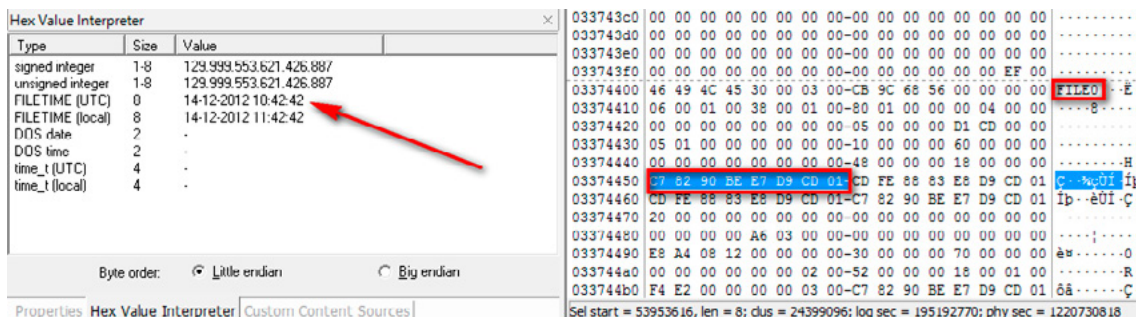


Figure 3. Creation time (FTK)

At byte offset 80 after the magic marker, shows the file creation time, which is 8 bytes in length. In order to find byte offset 80, press CTRL + G (from current position).

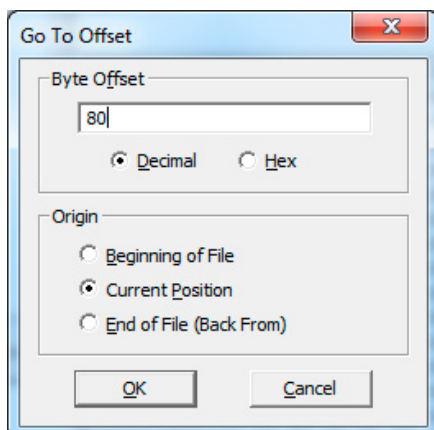


Figure 4. Byte Offset (FTK)

At byte offset 80 after the magic marker, select 8 bytes and the Hex Value Interpreter shows the creation time of the file is 14-12-2012 10:42:42 UTC.

ALTERNATION TIME

The next 8 bytes show the file alternation time (UTC)

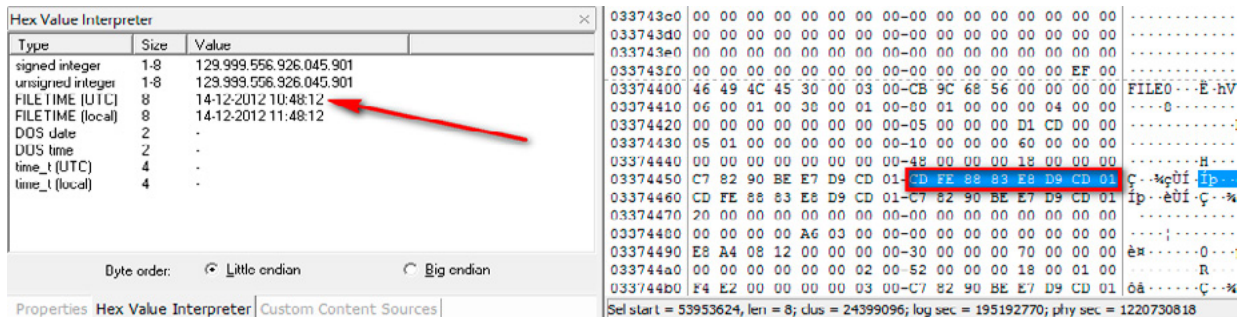


Figure 5. Alternation time (FTK)

MFT CHANGE TIME

The next 8 bytes show the MFT change time (UTC)

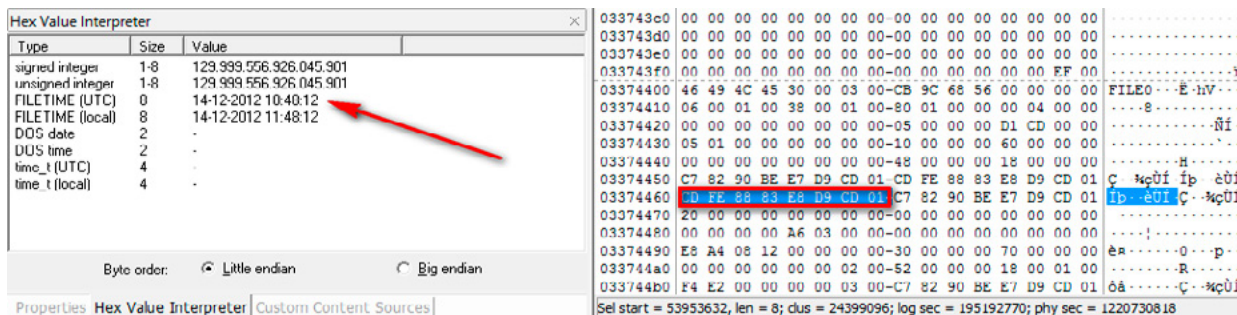


Figure 6. MFT change time (FTK)

FILE READ TIME

The next 8 bytes show the File Read Time (UTC)

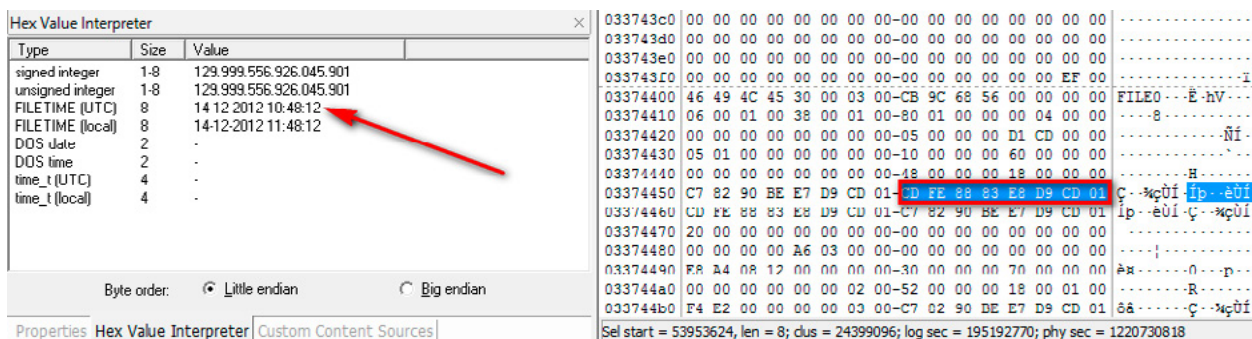


Figure 7. File Read Time (FTK)

Recover this picture for further analysis.

HOW TO RECOVER DATA

One of the MFT attributes is the \$DATA section. It starts with code 0 x 80 00 00 00. Go back to the magic marker FILE0 and use CTRL + F and do a Binary(hex) search for 80000000. This will point directly to the \$DATA section of the specific MFT record.

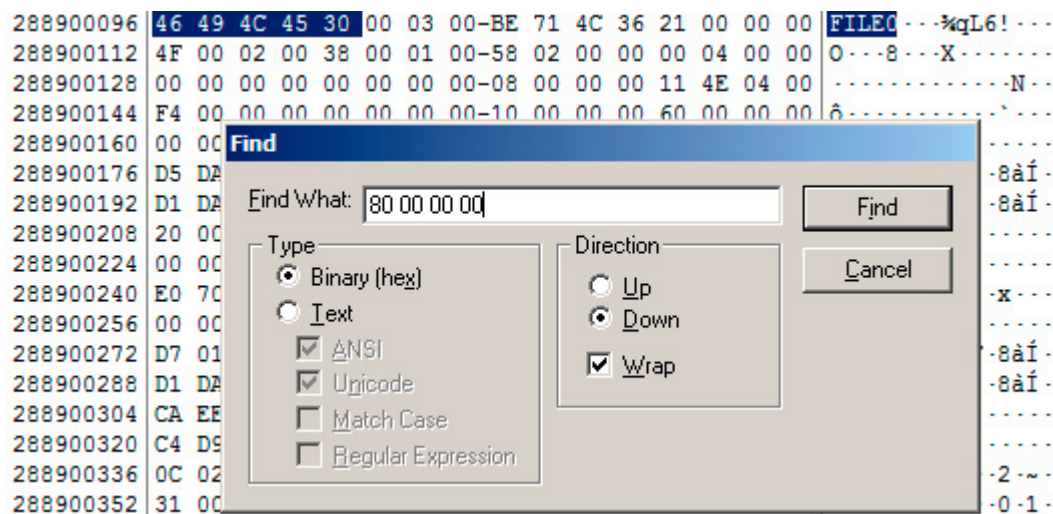
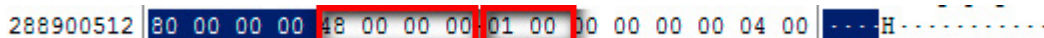


Figure 8. \$DATA section (FTK)

Notice the length of the \$DATA section is 0 x 48 00 00 00.



The 4 bytes behind 0 x 80 00 00 00 shows the length of the \$DATA section. In this case it is 0 x 48 00 00 00. The Hex Value Interpreter converts this to 72 decimal.



Figure 9. Hex Value Interpreter (FTK)

DELETED OR NOT

The code right next to 0 x 48 00 00 00 is 0 x 01 00.

- 01 00 means existing file
- 00 00 means deleted file
- 03 00 means existing folder
- 04 00 means deleted folder

The picture to be recovered has not been deleted from the hard drive. Information about the actual location of the picture on the hard drive is available in data runs, which start at byte offset 32 of the \$DATA section.

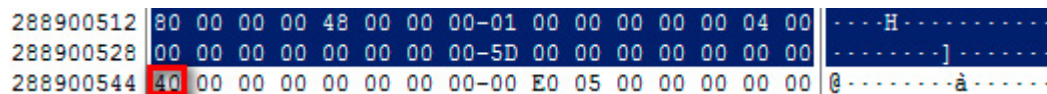


Figure 10. \$DATA section (FTK)

In this case, byte offset 32 of the \$DATA section is 0 x 40. The Hex Value Interpreter converts this to 64 decimal.

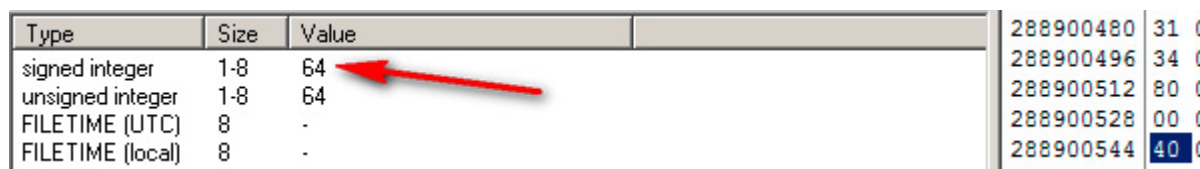


Figure 11. Hex Value Interpreter (FTK)

Now, go to byte offset 64 from the beginning of the \$DATA section where you will find the data run with information about the first cluster of the picture data. At many times the data run starts with 0 x 31 and ends with 0 x 0, but this is not always the case. In this case, the data run starts with 0 x 31.

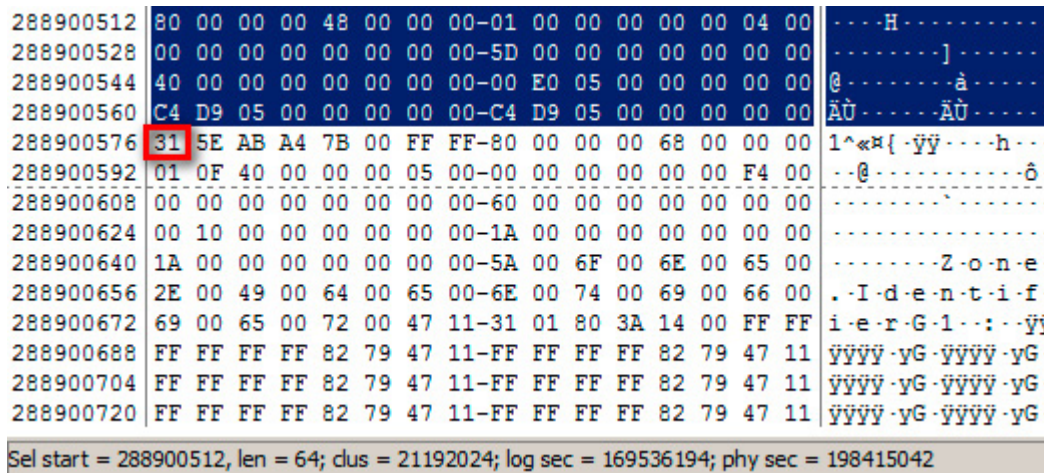


Figure 12. Data run information (FTK)

The code next to 0 x 31 (in this case 0 x 5E) shows the amount of clusters belonging to the picture data.

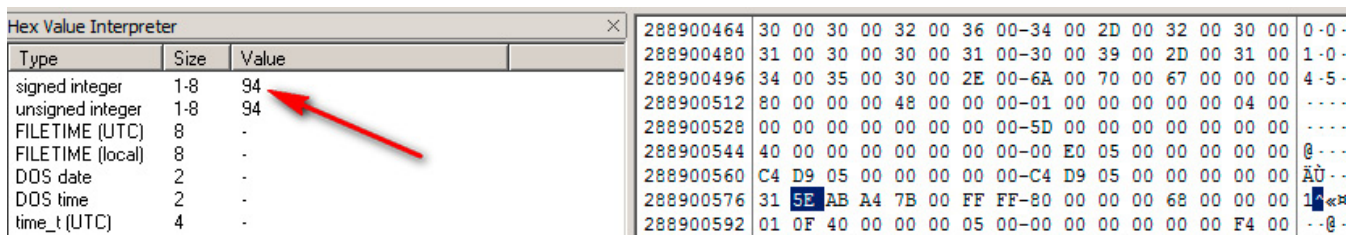


Figure 13. Clusters (FTK)

The Hex Value Interpreter converts this to 94 decimal, which means the data of the picture fills 94 clusters.

The next 3 bytes (0 x AB A4 7B) show the number of the cluster. The Hex Value Interpreter converts this to 8103083 decimal.

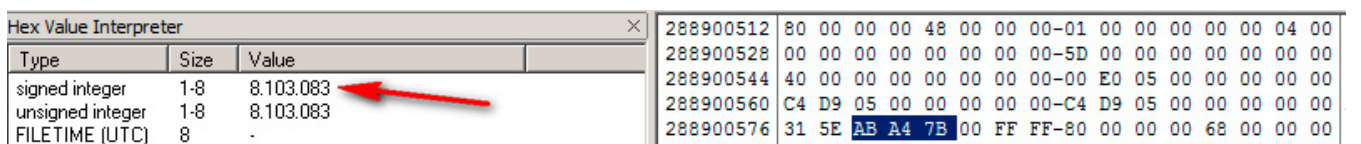


Figure 14. Hex Value Interpreter (FTK)

Click on the Volume name in the Evidence Tree Pane and the Properties tab (next to the Hex Value Interpreter) show the size of one cluster is 4096 bytes.

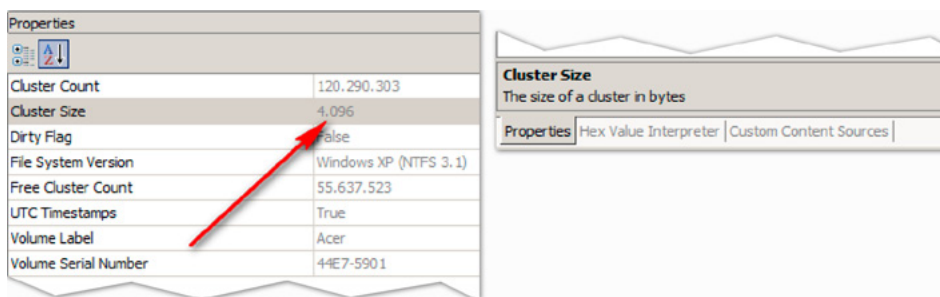


Figure 15. Properties tab (FTK)

SOME SIMPLE MATH

$$94 \times 4096 = 385024$$

Now you know:

The starting cluster is 8103083

The size of the photo is 385024 bytes.

Click the Volume name in the Evidence Tree Pane and right click the Viewer Pane.

Select “go to sector / cluster” and enter cluster number 8103083.

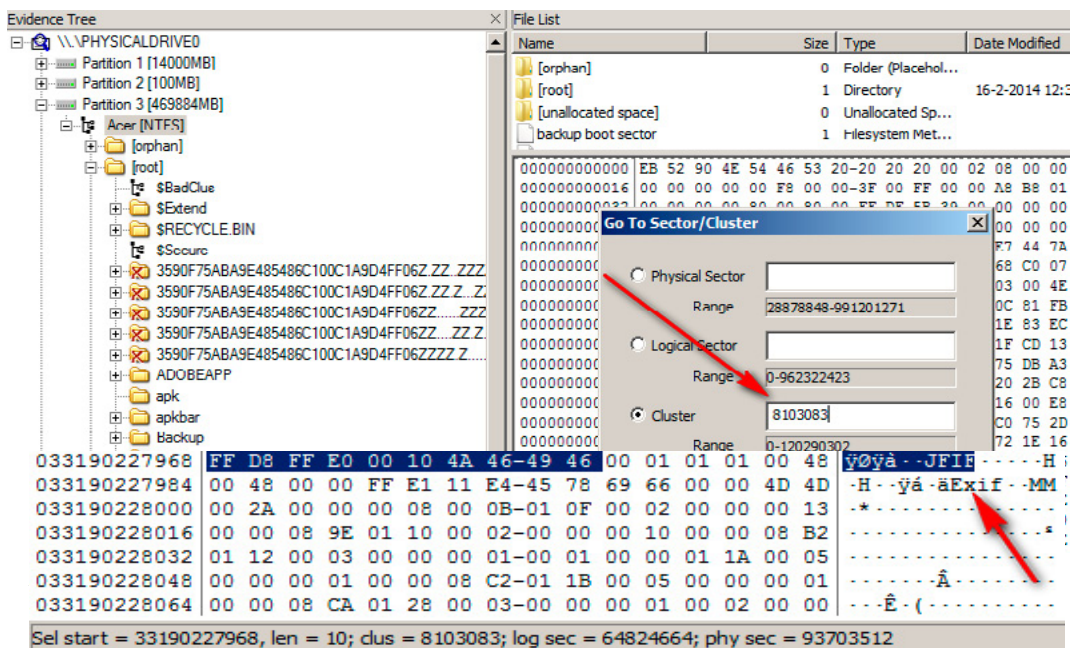


Figure 17. JPEG file header (FTK)

The file header of a JPEG file (ÿØÿà..JFIF) appears in the Viewer Pane.

Right click the Viewer Pane and enter 385024 in “Set Selection Length...”

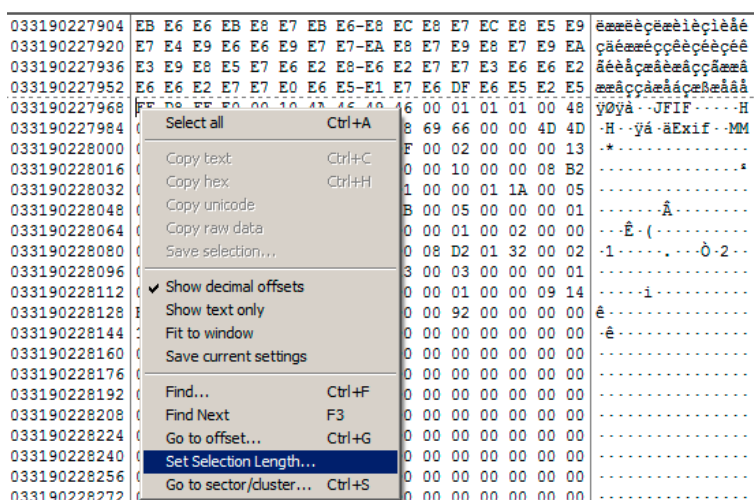


Figure 18. Set Selection Length (FTK)

Right click the selected data and use “Save Selection ...” in order to save the picture data as a file.



Figure 19. *The Result*

IN SUMMARY

NTFS uses the Master File Table (MFT) as a database to keep track of files. We can use FTK Imager to analyze the MFT and find interesting file artifacts and metadata.

REFERENCES

- NTFS Forensics: A Programmers View of Raw Filesystem Data Extraction Jason Medeiros, Grayscale Research 2008
- Computer and Information Security Handbook John R. Vacca, Elsevier 2013

ABOUT THE AUTHOR



Mark Stam is a digital forensics investigator who works at the National Police in The Netherlands. He specializes in Social Network Analysis and has given several presentations including at Data Expert's Digital Experience 2011 and 2013 in The Netherlands. Mark maintains a weblog at <http://stam.blogs.com>. A more complete profile can be accessed over at <http://nl.linkedin.com/pub/mark-stam/1/410/9a0/>.

Attend the Largest Dedicated Android Development Conference in the Universe!

AnDevCon

May 27-30, 2014

Sheraton Boston

Get the best real-world Android
developer training anywhere!

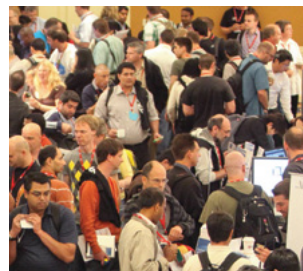
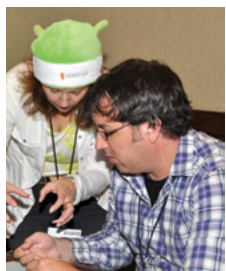
- Choose from more than 75 classes and in-depth tutorials
- Network with speakers and other Android developers
- Check out more than 40 exhibiting companies

Take your Android development skills
to the next level!



Find out why you should go
to AnDevCon! Watch the videos
at www.AnDevCon.com

Register Early
and SAVE!



Register Early and Save at www.AnDevCon.com

AnDevCon™ is a trademark of BZ Media LLC. Android™ is a trademark of Google Inc. Google's Android Robot is used under terms of the Creative Commons 3.0 Attribution License.

A BZ Media Event



#AnDevCon



USING FTK IMAGER CREATE

FORENSICALLY-SOUND COPIES OF DIGITAL MEDIA

by Austin Troxell

The first step in Digital Forensic examinations is to create precise duplicates of any storage media collected as potential evidence. The analyst must make certain that the original evidence remains unaltered and that the digital copy they are working from is a true bit-by-bit representation of the original media. A combination of hardware and software tools are available to the examiner to complete this task in preparation for analyzing the contents of the media.

What you will learn:

- How to create digitally identical copies of hard drives and other storage media
- The importance of “write-blocking”
- The concept of “hashes”

What you should know:

- How to remove a hard drive from a compute
- A basic familiarity with operating systems
- A basic familiarity with directory and file structures on hard drives

In simple terms, forensic techniques involve the scientific collection, examination and presentation of evidence in administrative, civil or criminal matters. By “scientific” I mean that the technologies used must be based on generally accepted standards, yielding reliable and repeatable results. As computer technology has become fully integrated into every facet of modern society, digital artifacts are an ever-present and invaluable source of relevant information in these cases.

One of the key principles of Digital Forensics is that examiners must eliminate or minimize the risk of altering any information contained on the original evidence items. Where at all possible, the analyst will make digital copies of the media to be examined and work from these duplicates, preserving the originals. The analyst will utilize a hardware write-blocking device that prevents any data to be written to change on the original evidence media. In addition, target media, that is the disk we copy the image to, will be the disk we, overwritten with all ones, zeroes or a random pattern to ensure no preexisting data on the target will be comingled with the subject data under scrutiny. This process is also known as “wiping” a drive. A wiping utility popular with both Digital Forensic examiners and PC Support technicians is “DBAN,” also available at no cost. [1]

The Digital Forensics examiner has numerous options for creating exact bit-stream representations of digital media, including hardware duplicators as well as various software tools that create digitally identical copies. In this article I focus on the features and use of AccessData’s *FTK Imager*.

FEATURES

FTK Imager is probably the best known software application for creating bit-stream duplicates of digital media. It is provided as a free of charge download by AccessData in both an installable and portable version (*FTK Imager Lite*). Either version can be used to create forensic copies of entire hard drives (physical imaging), partitions (logical drives), or selected folders and files (sparse imaging). Additionally, flash memory cards, USB drives, optical drives (CD/DVD) and Fernico devices (multiple-CD/DVD servers) may be duplicated, as well. Hashing of images and files is automatically performed using both MD5 and SHA1 algorithms.

At this point, I need to define this point for the beginner practitioner. Hashing is the process of creating a fixed-length digital “is the process of creating a fixed-length digital copies of entire hthe fileprocess of creating a fixedEven viewing a file using Explorer or some other utility will alter a file’s last accessed date, which will, in turn, change the hash value of that file. In order to prevent unintended changes to a file’ In order or its er to pre, ”othe examiner utilizes either a hardware device or software utility to serve as a “ther a hardware device or software u [2]. In my practice, I prefer to use a hardware write-blocker to help ensure that I do not inadvertently change the contents of the data under examination. I feel that software write-blocking utilities are too prone to mishap or forgetfulness.

A nice feature is that *FTK Imager* has the ability to convert between the most common forensic image formats, RAW, SMART, E01 and AFF. This can be very useful if you have a disk image in RAW (dd) format and need to share the image with, for example, an investigator who requests the image in E01 format for use with EnCase. For LINUX users, *FTK Imager* is also available in both 32- and 64-bit command-line versions.

Other handy features of *FTK Imager* include the ability to perform a live acquisition of a computer’s RAM as well as obtain the Registry from Windows-based PCs. Depending on output format, images may be compressed or split, which makes copying the results to external media easier and more reliable.

A feature added to the newer versions of *FTK Imager* is the ability to mount forensic images as physical or logical drives in order to quickly search or preview contents. I have found this to be particularly helpful when examining acquired images of Macintosh computers on Windows forensic workstations. Previously, this required that the examiner obtain third-party mounting utilities at additional cost. Even without explicitly mounting an image as a physical or logical drive, adding the drive -or image- as an evidence item permits the examiner to scroll through the contents of the evidence using the built-in hexadecimal/text viewer, another handy feature for previewing evidence.

LIMITATIONS

Although *FTK Imager* boasts an impressive list of features, all the more remarkable considering that it is free-ware; the application does have some limitations that an examiner needs to be aware of.

First, because it is a Windows-based product, *FTK Imager* cannot access any Host Protected Area (HPA) or Device Configuration Overlay (DCO) that may be present on a hard drive. If these areas are on a hard drive and you have a reasonable expectation that these may contain evidence, then you will need to utilize either a LINUX-based solution or a hardware disk duplicator.

Second, *FTK Imager* does not handle bad blocks well. If the number of bad blocks or unreadable sectors is significant, this application will slow to the point of being unusable. The read-ahead cache encounters the errors and becomes unresponsive. In such cases, you will need to obtain a utility that can perform “reverse cloning” to bypass the drive cache and recover data.

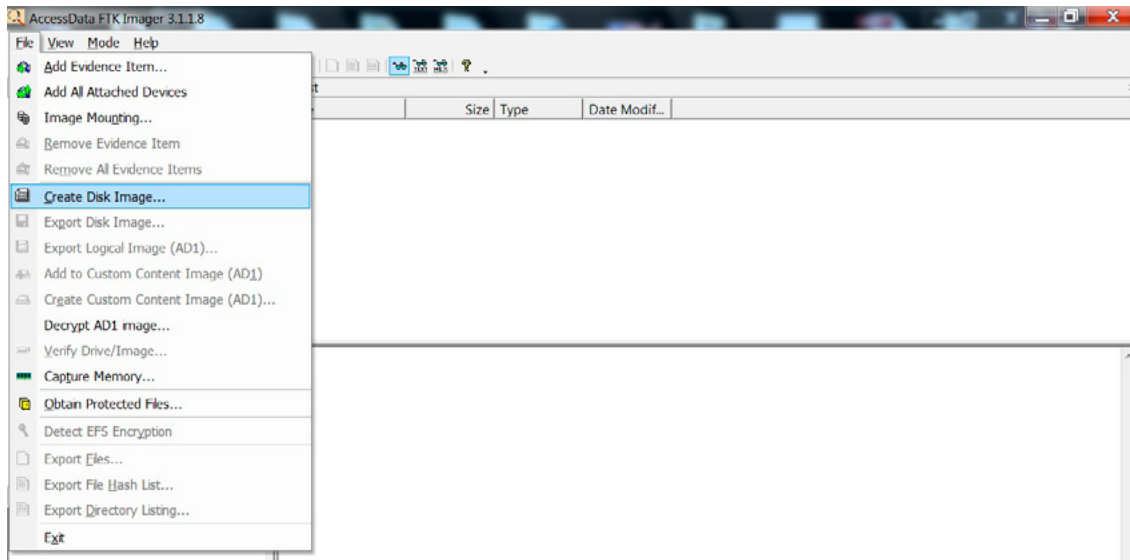
Third, creating forensic duplicates of large capacity drives takes time. This is not a limitation of *FTK Imager*, but is endemic to all software imaging tools. In my experience it requires approximately 6 hours to duplicate a 250 gigabyte hard drive; a one terabyte drive needs roughly 24 hours to completely copy. Hardware duplicators will perform the task more quickly and may be the preferred option if speed is a requirement. Keep in mind also that creating the forensic drive image is just the beginning of the examination process. Using applications such as AccessData’s *Forensic Toolkit* require at least an equal amount of time to carve and index the directories, files and deleted data within a drive image.

WALK-THROUGH

To best appreciate the process of using *FTK Imager*, let's walk through the process of creating a forensic bit-stream image of a 64 GB flash drive. For the purposes of this demonstration, we will copy the drive image to our forensic workstation. In actual practice you would want to write the image to previously 's to completely copy. Hardware duplicators

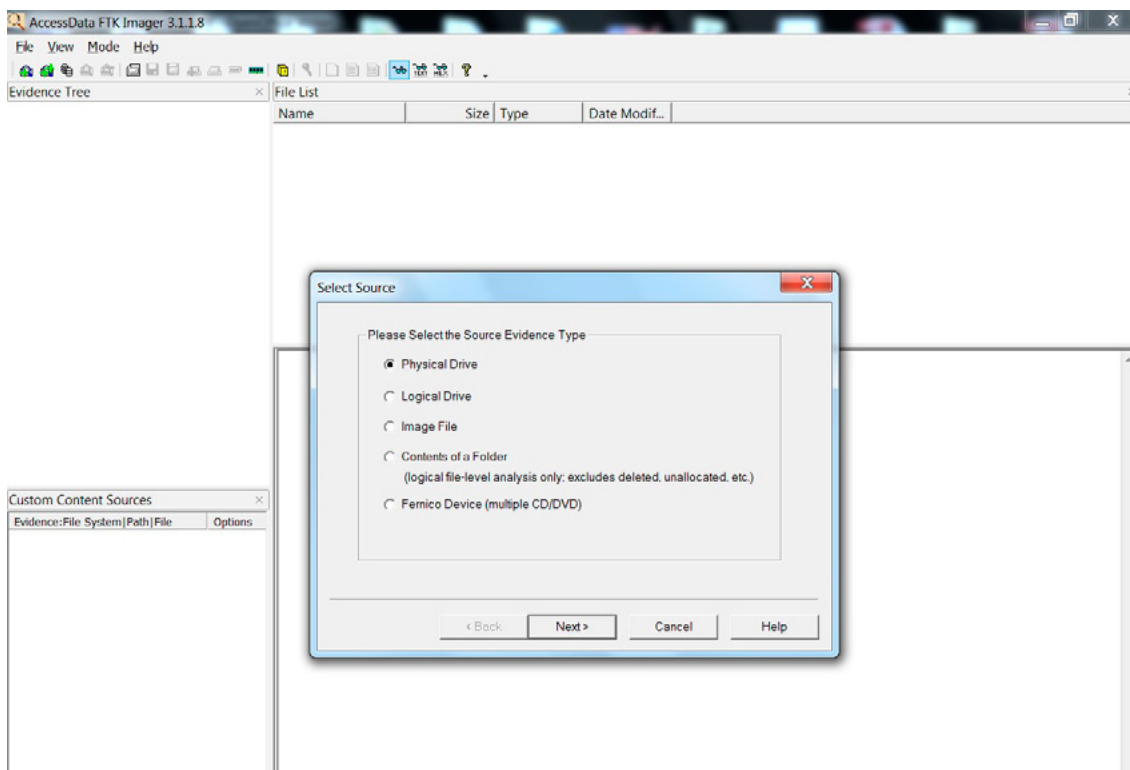
STEP 1

Download, install and run *FTK Imager* [3]. From the install menu select "Create Disk Image".



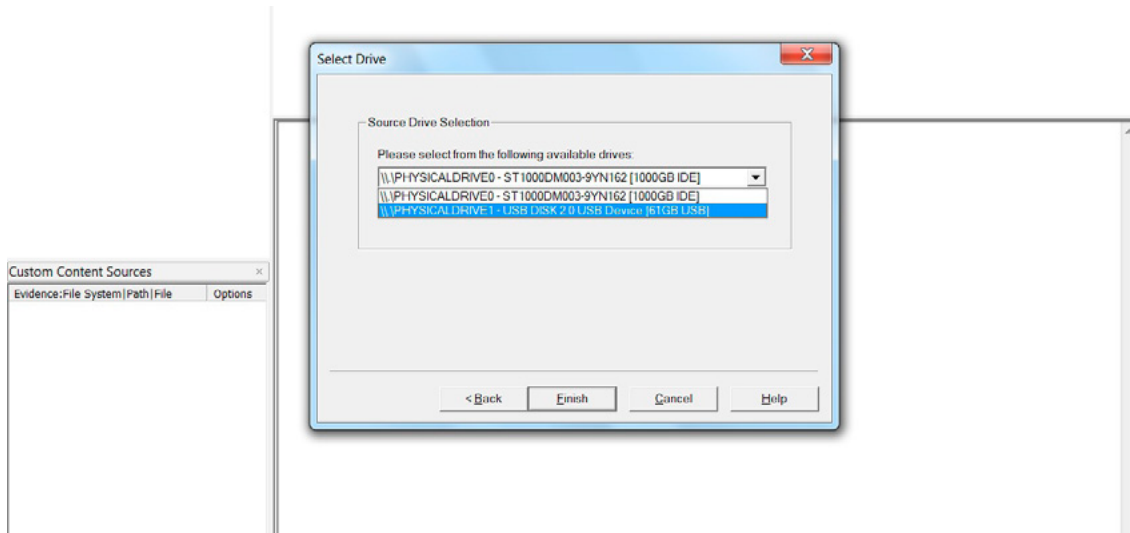
STEP 2

Next leave the default option, Create Disk Imageation, we will copy the drive image to our forensic work [4].



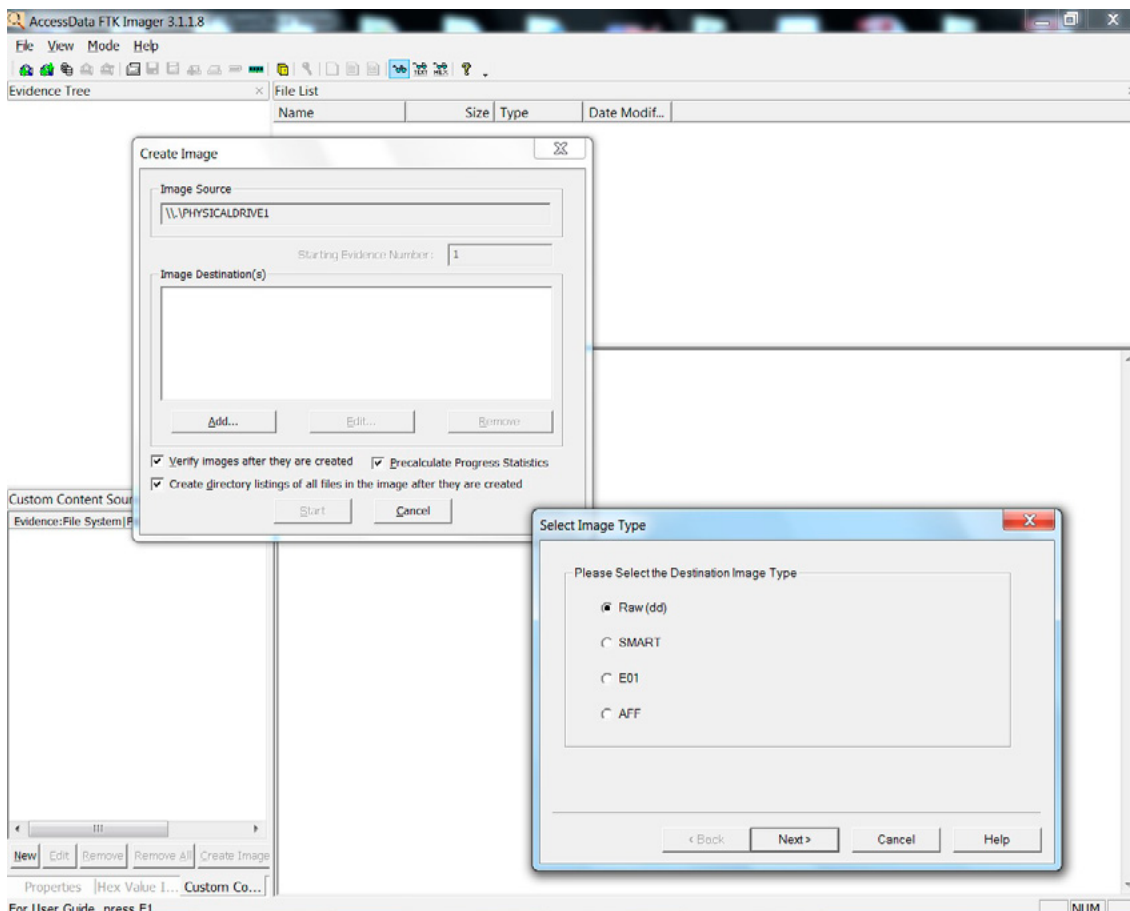
STEP 3

Select the correct drive from the "Select Drive" window, we drop-down and click "Finish."



STEP 4

The next screen allows you to select the target location for your captured image. Click "Add" in the "Image Destination(s)" window. Note: You may add multiple destinations, thus permitting you to create more than one drive image simultaneously. You are also presented with the option to select output image format (Raw (dd), SMART, E01, AFF). Clicking "Next" takes you to a screen where you will enter identifying information about the case, evidence and examiner.



STEP 5

Click “Next” to specify where you will store and name the captured image. You can also select image fragment size and degree of compression, if desired. Click “Finish” to return to the prior screen and select “Start.”

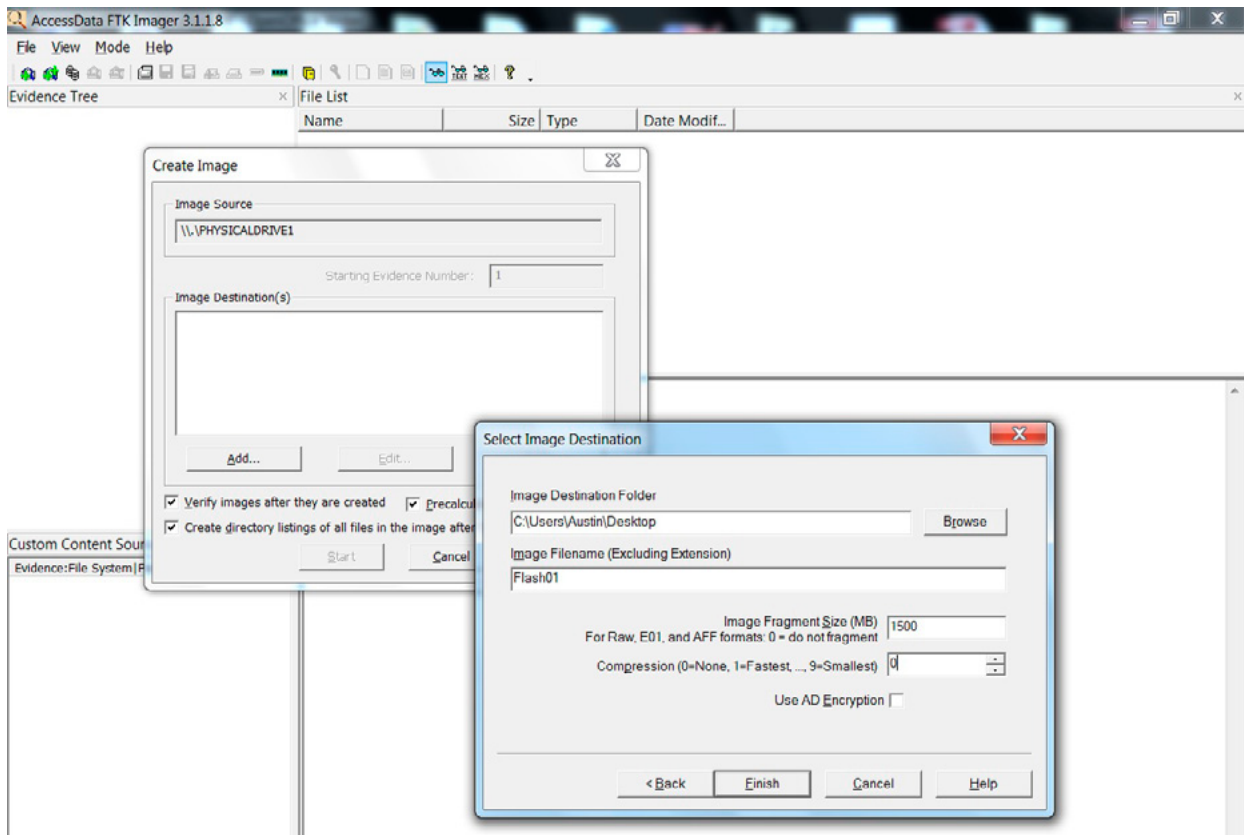
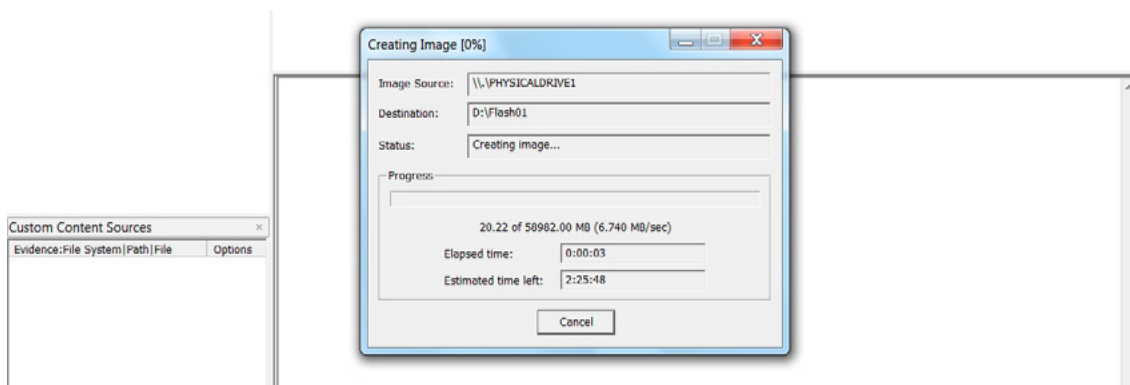


Figure 5. XXXXX

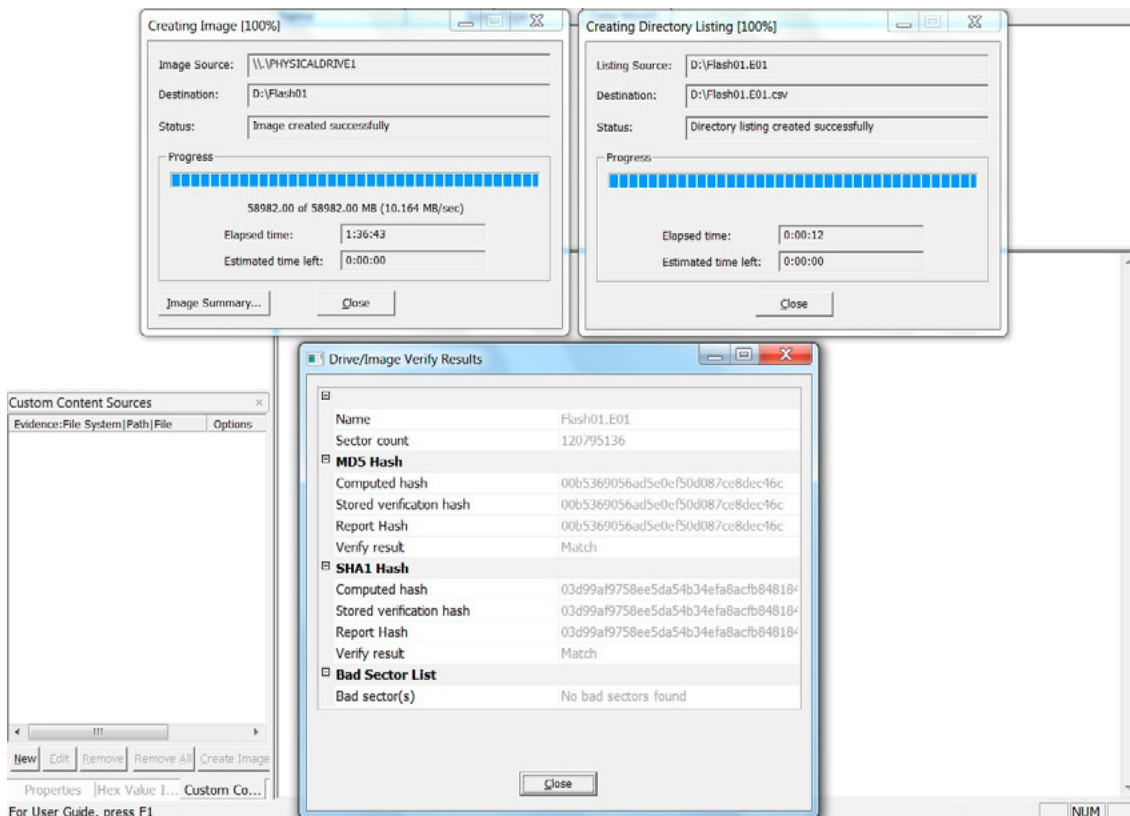
STEP 6

At this point, you will see a progress screen as the image is being written to the target location(s). FTK Imager provides both elapsed time and an estimate of the remaining time for image creation.



STEP 7

Upon completion, you will see three summary screens that show the image has been completed, the verification summary as well as confirmation that a directory listing has been completed, as well as a screen indicating whether or not hash values of the image match those of the source.



At this point, you may exit *FTK Imager* and mount the image in a forensics application, such as X-Ways Forensics [5], AccessData's *Forensic Toolkit* [6], Guidance Software's *EnCase* [7], or from within a hexadecimal viewer and commence analysis of the image contents.

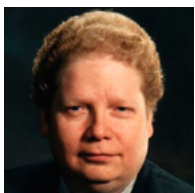
SUMMARY

As you can see, *FTK Imager* offers a well-designed, simple-to-use and feature-rich product to enable Digital Forensic examiners to capture exact bit-stream copies of source media from a variety of file systems and physical formats. When you also consider that this application is provided at a price that can't be beat (free!), you can understand why *FTK Imager* is found in the signed, simple-to-use and feature-rich prod professional.

REFERENCES

1. DBAN, also known as "Darik's Boot and Nuke," may be downloaded from <http://www.dban.org/download> (Retrieved 09 March 2014)
2. Retrieved 0 colloquially defined as "data about data," such as when a file was created, modified or viewed.
3. Available at <http://www.accessdata.com/support/product-downloads> (Retrieved 21 February, 2014)
4. You will note the use of the term "Source" drive rather than "Evidence" in this article. I feel that calling a drive "Evidence" implies a presumption of guilt. For this reason I also prefer the term "Subject" rather than "Suspect" or "Defendant." I strongly believe that Digital Forensic professionals must remain unbiased in their work.
5. <http://www.x-ways.net/forensics/>
6. <http://www.accessdata.com/products/digital-forensics/ftk>
7. <http://www.guidancesoftware.com/products/Pages/encase-forensic/overview.aspx>

ABOUT THE AUTHOR



Austin W. Troxell holds a Master of Science degree in Information Assurance (Digital Forensics specialization) from Norwich University in Northfield, Vermont, U.S.A. He earned the designation of Certified Information System Security Professional (CISSP) in 2002. Mr. Troxell has over 20 years experience in Information Technology and has been a Digital Forensics Analyst since 2007. His work has led to successful outcomes in legal cases ranging from sexual assault, possession and dissemination of indecent imagery, theft of intellectual property and web site redirection, among many others. Mr. Troxell currently teaches courses in Cyber Security and Digital Forensics at Fortis College in Centerville, Ohio, U.S.A. He may be contacted for consulting opportunities at awtroxell@aol.com.

CREATING A FORENSIC IMAGE OF A HARD DRIVE USING FTK IMAGER AND IMAGER-LITE FROM ACCESSDATA

by **Bridgette Braxton**

The advancement in the world of computer forensics has provided many tools to assist the incident responders perform live analysis on a computer. The capabilities of forensics tools have improved and have made analysis feasible by integrating enhanced interfaces, documentation, built-in detection methods, and new ways to collect evidence. Live memory analysis involves the access of physical memory. My choice of software for Imaging is AccessData FTK Imager.

What you will learn:

- Download and install AccessData
- FTK Imager and Imager-Lite
- How to create images of hard drives
- Adding evidence Items
- How to preview files and folders on the hard drive
- Mounting an image for read only viewing
- Removing Evidence
- Acquiring Protected files
- Encrypted Images
- Creating hashes using hash functions (MD5) and (SHA-1)
- Create a hash report to prove integrity of evidence

What you should know:

- Use a hardware-based write-blocker
- FTK Imager can be run from a portable USB thumb drive
- Utilizing User Interface options
- The use of hexadecimal values
- Previewing HTML content must not have active Internet connection
- Difference between Logical and Physical mounting

It provides an easy way to image a hard drive that allows the investigator to create dd images, Smart images, and EnCase images. The program loads quickly, creates forensic images that allow easy previewing of the hard drives files/folders and media, mounts images for read-only view to see the contents on the original drive, exports files, recovers files that have been deleted that have not been overwritten, creates hash files using Message Digest 5 (MD5) and Secure Hash Algorithm (SHA1) that verifies the integrity of the images have not been altered or changed. FTK Imager is a free program provided by AccessData the same company that provides AccessData FTK Imager lite and it is one of the best drive imaging and evidence collection programs I have used and it's a court-accepted digital forensic tool. The uniqueness of Imager Lite is that you can download the program to a USB and copy the files right off the hard drive without running any installation. I would highly recommend this program for Investigators and analysis as part of their imaging tools. This tool can also be used on a virtual machine workstation or a MAC.

AccessData FTK Imager and Imager-Lite are powerful forensics tools used to create forensics images of hard drives, CD's, Zip Disk, DVDs, files and individual folders. FTK imager saves images in several different formats like DD /RAW (Linux "Disk Dump"), AFF

(Advanced Forensic Format) and E01 (EnCase) that creates forensic images of data. FTK imager is used by forensics experts and professionals working with data recovery. It mounts images at the physical, logical level, and customizes content images virtually. FTK Imager and Imager-Lite create a perfect copy of forensic images of the computers data without manipulating the original evidence. It also allows you to preview forensics evidence on a local machine or network by mounting the image for read-only view of the content on the original drive. This allows you to quickly assess data and determine if further analysis is warranted.

CREATING A FORENSIC IMAGE WITH FTK IMAGER AND FTK IMAGER-LITE

Creating a forensic image of a hard drive is valuable evidence that should be treated with respect and integrity at the highest of levels. The methods used in recovering evidence cannot be overemphasized that the importance of the rules of evidence must be applied equally, ensure compliance in accordance with statutes and case laws, collection, examinations, analysis and reporting. These steps are very important to the collection of computer-based electronic evidence. The integrity of the evidence is very important because it examines patterns of activity against allegations that can lead to legal complications if not handled properly. One problem you don't want to run into is improperly accessing data stored on electronic devices, you must have the knowledge and expertise as this may violate Federal laws, including the Electronic Communications Privacy Act of 1986 and the Privacy Protection Act of 1980.

OVERVIEW

FTK Imager and Imager Lite are used to create a forensic image of a hard drive for evidence. The main function is to view what's on the hard drive and image its storage device contents. You want to make sure that you use a hardware-based write blocking device to avoid alteration of digital evidence while imaging the hard drive of your suspect's computer. Be careful of Trojan binaries and root kits that have been installed by the individual in question. Some computers are set up to notify the suspect when specific actions are taking place while they are away. The brilliant thing about this is they can remotely view the computer and everything you are doing on the screen in real time. To avoid this you can disable the internet connection so that the alert will not be sent to the subject in question to eliminate them from viewing what is forensically being done to retrieve information from the computer. You must also follow the digital chain of custody, which includes creating a digital fingerprint by hashing all the data images.

USING FTK IMAGER

Now that we have briefly covered the legal aspects of imaging a hard drive, I will now cover the basics of creating a forensic image using FTK Imager and Imager Lite. When using FTK Imager or Imager-Lite you are verifying that the tool is making a bit-stream duplicate of the original hard drive. This prevents accidental manipulation of the evidence. The forensic image is identical to the original hard drive that includes a slack/unallocated/and drive free space.

MOUNTING

AccessData FTK Imager mounts images as a drive or physical device for read only viewing. The supportive types are Encase (E01), SMART (S01) which contain their own hash value that is stored within the image, Advanced Forensics Format (AFF), and RAW/DD (001) images physically, or mount E01, S01, and RAW/dd partition images which are drive images that include the disk, partition, file structure and data drive, AD Custom Content (AD1) and (L01) are custom content images that contain the full file structure without the drive geometry or physical drive data and can only be mounted logically. FTK Imager can also read and create Advanced Forensics Format (AFF) is used for encrypting an image were a password is required (one that you create) to decrypt the file at a later time and can be used as a snapshot/RAM acquisition utility by clicking on the Capture Memory icon. The focus of data imaging is the collection of evidence that supports the investigation in any court of law. This is the most critical part of evidence collection, imaging and creating an exact replica of the device and data that is being considered as digital forensic evidence. Once the hard drive is mounted you can freely use third party applications such as Antivirus software to search for Viruses/Trojans. The benefits that FTK Imager provides is mounting a full disk and its partition at once, easily unmounts images, disk is given the next available drive letter for querying, you can run antivirus applications, can view the logically mounted image remotely, and can copy the mounted image to another location, just to name a few.

ACCESSDATA IMAGER LITE

To touch a little on AccessData Imager Lite, it's a self-executable that does not need to be installed to be used. You can run it from a CD or thumb drive to create a physical or logical image of any drive as well as the RAM. It uses raw dd format, SMART or E01 format. You will need to selection the size you want the

image to be divided into depending on the size of the media. This tool is in .exe format which allows you to use a command-line used in Linux, Windows or Mac. FTK Imager can also be used to open VMware .vmdk files. The easiest way to obtain such systems is to simply copy the .vmdk /.vmem files off of the host system. With FTK Imager, you can choose to add an Evidence Item to view the file system and extract specific files or choose Create Disk Image to acquire the .vmdk or .E0x image file to raw dd, SMART, or .E0x format. This can be extremely useful when using commercial analysis tools that may not recognize the vmdk format or may be more cumbersome than necessary for the work you intend to perform.

INSTALLING ACCESSDATA FTK IMAGER

You can install FTK Imager to any hard drive for previewing and imaging evidence from either the installation disc or from a saved download on a USB thumb drive. This can be retrieved from <http://www.accessdata.com/support/product-downloads>. Once the download is complete browse to the location you saved the file in. From there execute the setup file and follow the steps until the installation is complete. If you want to use FTK Imager Lite using a portable device, copy the FTK Imager Lite files directly to the device (USB thumb Drive/CD) this will avoid installation to the computer. Unzip the portable drive and execute the program from there. This will become your portable FTK Imager and can be connected to any running Windows OS machine. Make sure you have a target drive for saving the imaged data on your portable device other than the one that is running the program. Now you can execute the file from the thumb drive and start the imaging process.

First thing you do when using FTK Imager is go to file and add an evidence item and that will allow you to select a source and open a drive that is on your computer or one that is connected to your computer (USB thumb drive).

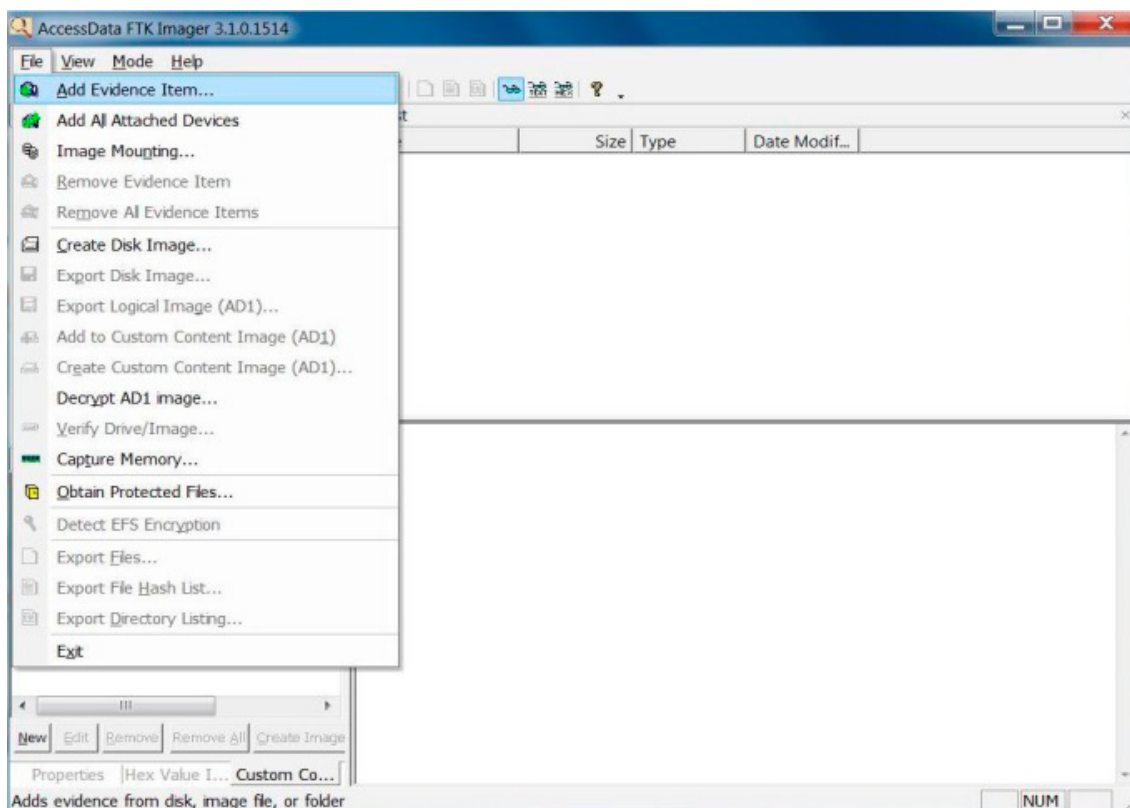


Figure 1. Add Evidence screen shot

There are several choices to select from. You have the Physical Drive which indicates you opening a drive including all the data that is on the drive (allocated, unallocated, deleted, etc.). and the Logical Drive that indicates you want to open just the drive with the logical components that are currently allocated. This includes files, folders, etc. that are still active on the drive that you can see in the windows interface. You also have the choice of just opening up an image file or the contents of a folder that excludes deleted and unallocated space.

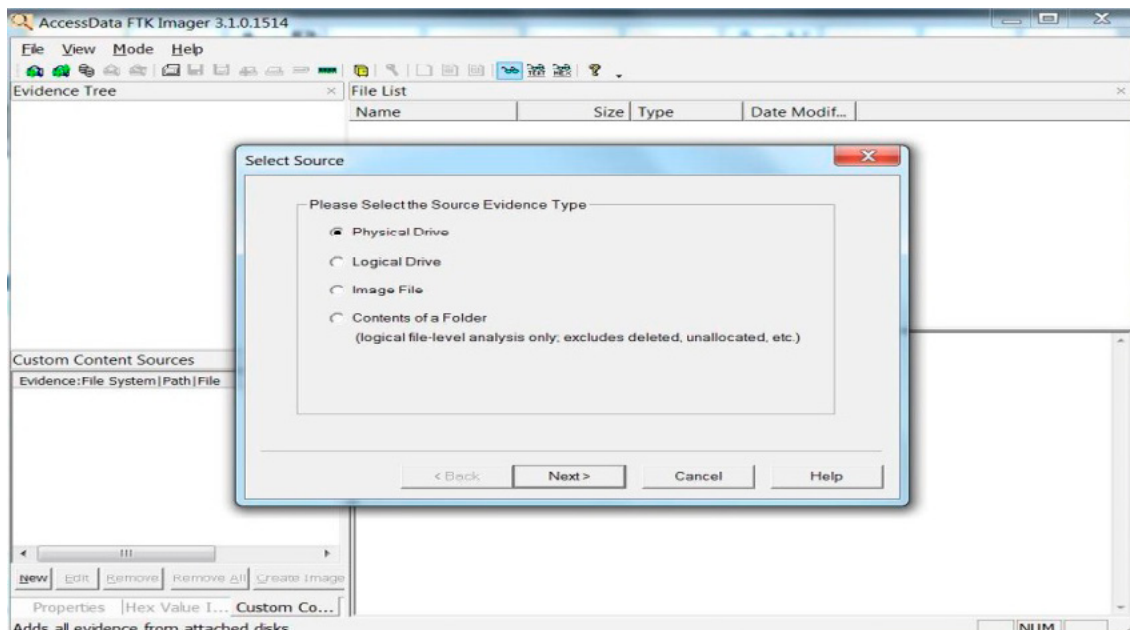


Figure 2. Select Source screen shot

For example if you have a USB thumb drive, you can open it as a physical drive and it will display all the physical drives on the computer. Choosing the USB once opened under the evidence tree you will see the physical drive of the USB you have chosen with components. If you click on the root of the file it will show the files on that USB. You will see the name of the file, the size, the type and date modified. Once you choose a file you will be able to review the hex part of the file chosen in the bottom pane. You can also choose “Add All Attached Devices button” to add all of the devices attached to the computer on the toolbar, this is known as auto-mount, and it scans the entire physical and logical devices for media. You can also choose to add multiple evidence items so that you make look at more than one at a time. The files that have an x to them are files that have been deleted, but not overwritten. By using FTK Imager it allows you to not only view active data but inactive data and deleted files, file slack and unallocated space by clicking on the file you can view the bit-by-bit contents in the lower right window.

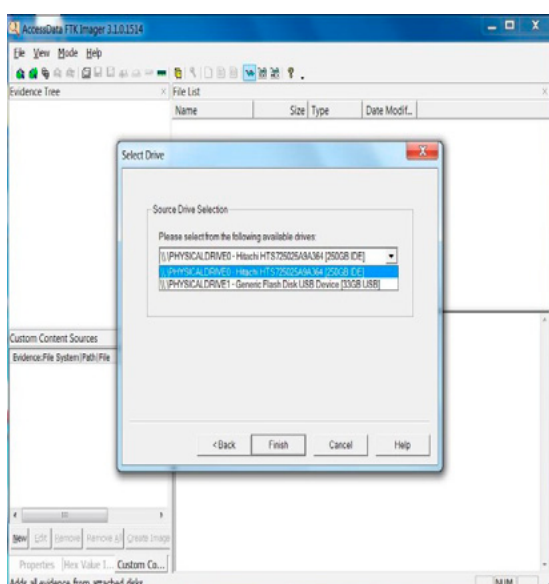


Figure 3. Select Drive screen shot

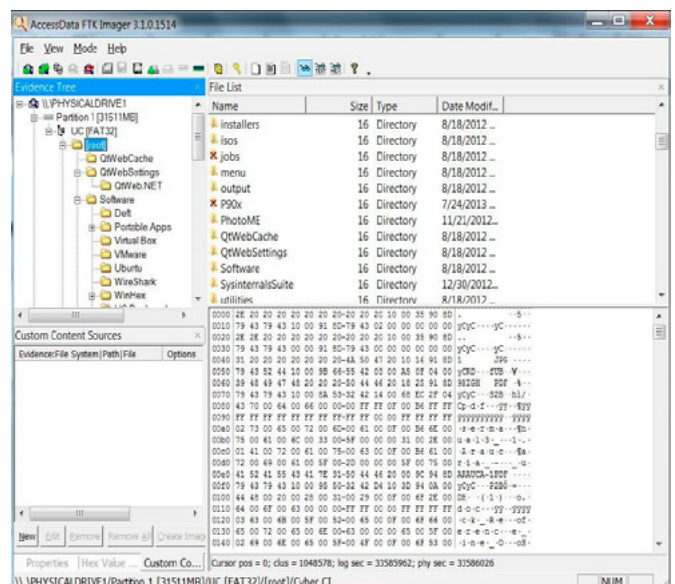


Figure 4. Evidence Tree screen shot

Some files can be opened in imager such as jpg or gif and will display the image in the view pane. You can also launch video or music files by using an external viewer like media player.

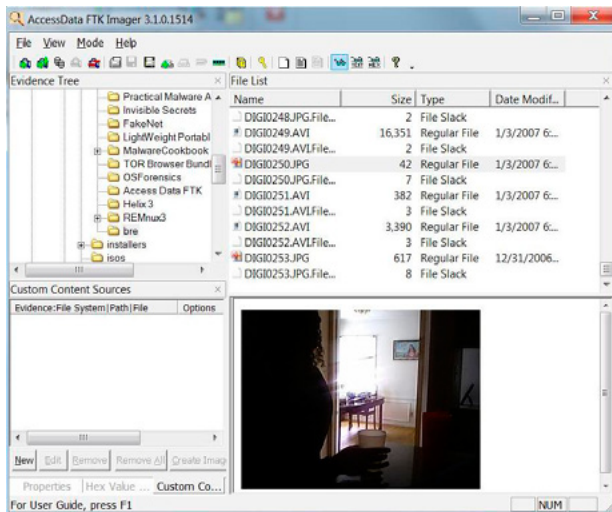


Figure 5. JPG or GIF screen shot

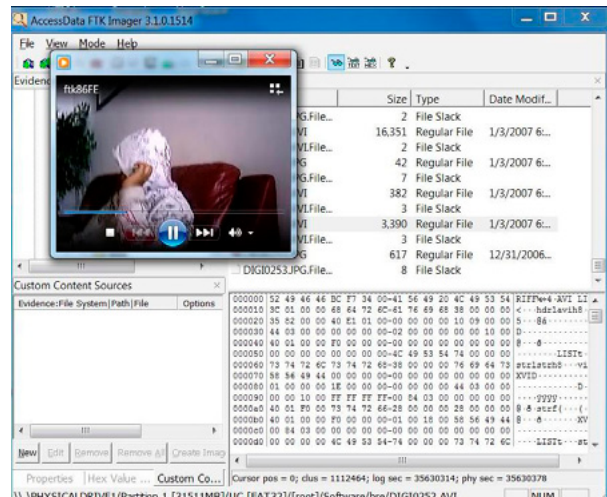


Figure 6. Video File screen shot

If you find another file of interest to your investigation you can right click on the file and export the file to your destination folder for review.

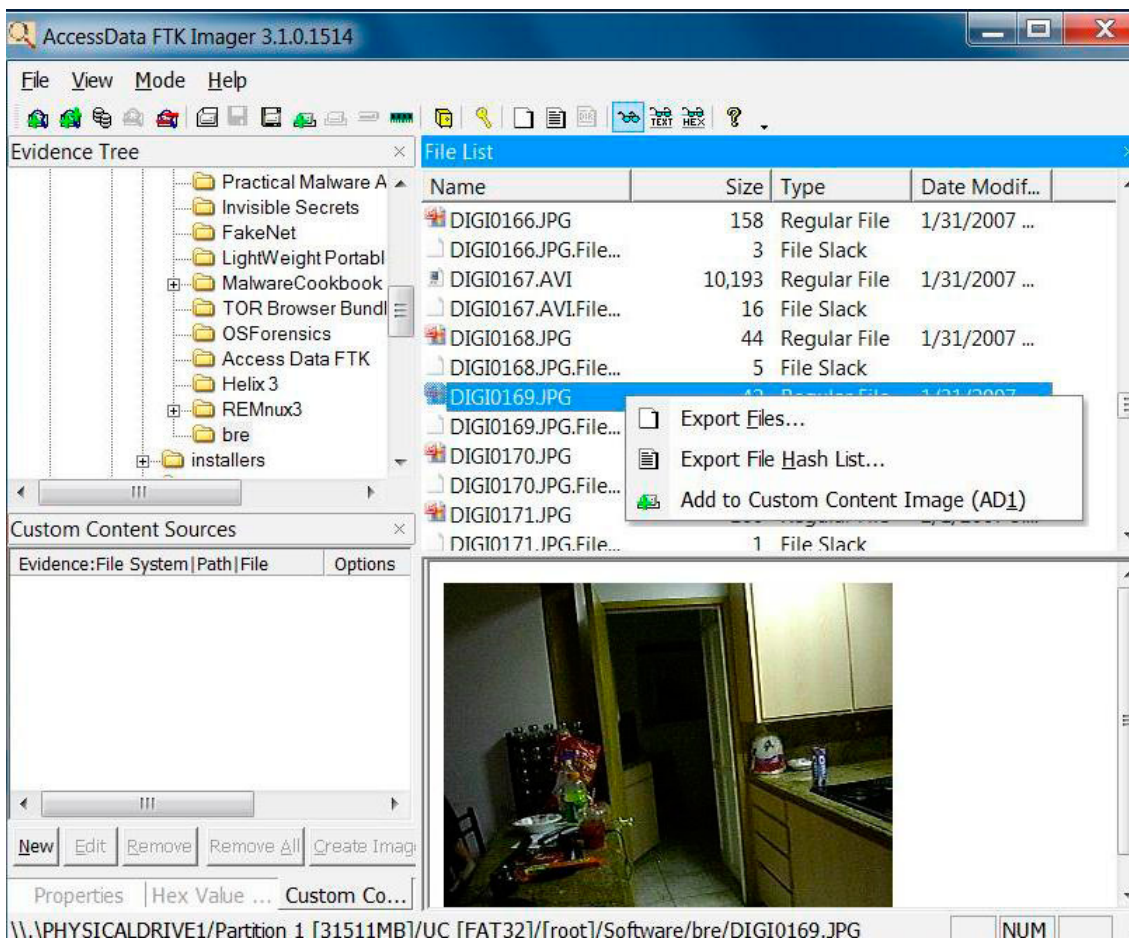


Figure 7. Export File screen shot

The next function in FTK imager is creating a disk image which is one of the biggest features in FTK imager. Open file and on the drop down menu select source evidence type (which I choose as the Physical Drive), click next and select the USB or source evidence type of choice and chose the destination folder. Click 'Add' and choose what format you want the disk image in.

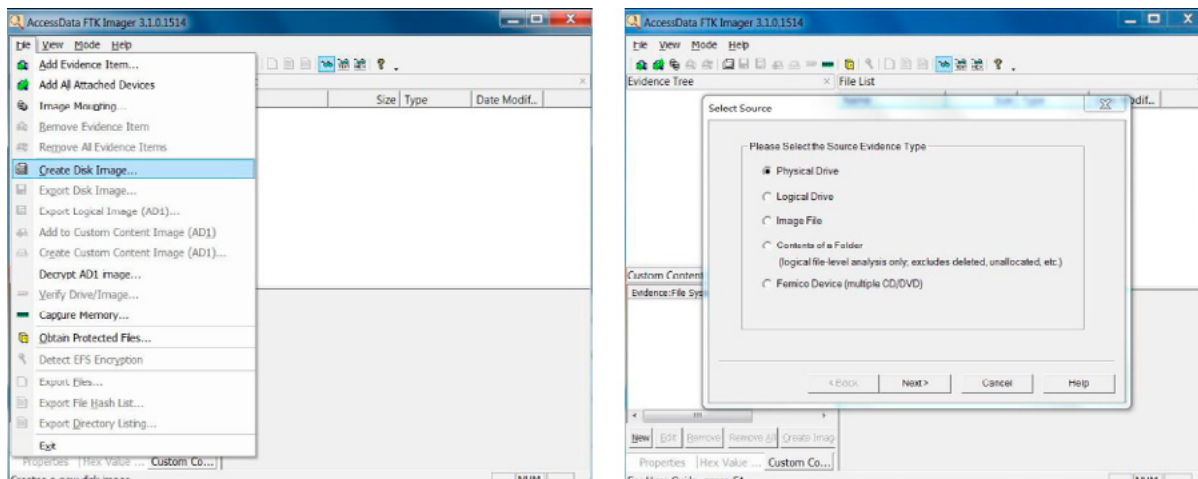


Figure 8. Select Drive screen shot

You have a choice of RAW format, SMART, E01 and AFF. RAW is very common, but E01 files are predominately used by FTK because they can be compressed and take up less space. To use this feature you must create a case number, evidence number, destination, and create a disk name for your file.

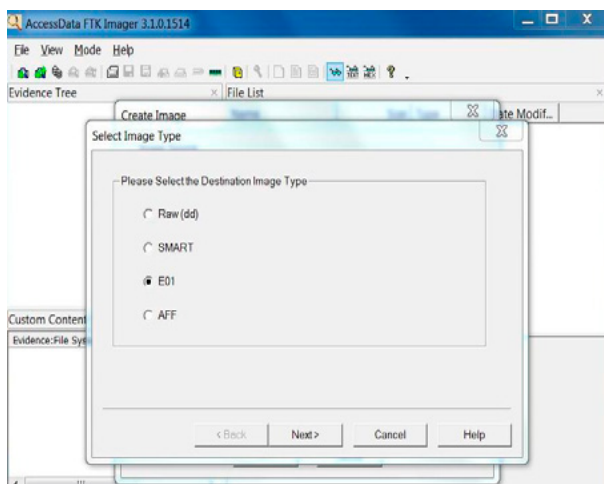


Figure 9. Select Image Type screen shot

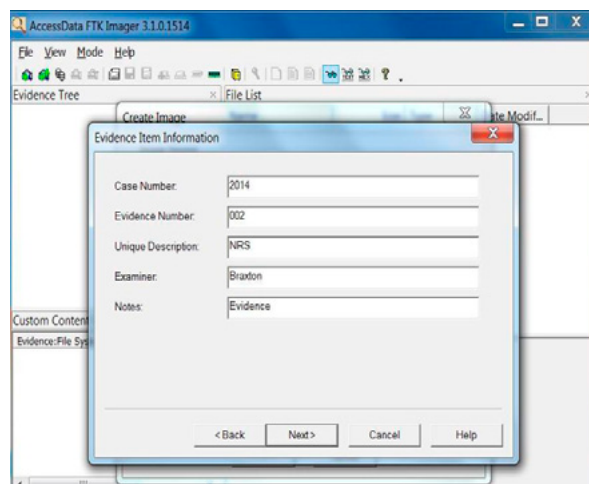


Figure 10. Evidence Item Info screen shot

Now you have the option to set your image fragments and compression. I would use the default of 1500 as the image fragment breaking it into chunks and compression is set at zero allowing you to move from 0=None, 1=Fastest, and 9=smallest. You want to image as quick as possible but you want to create the largest file image as possible so you want to set it at 5 or 6 which will give it a balance between speed and size of the image file. Once you click finish you will see two boxes that have been checked on the bottom for you. One is Verify images after they are created and the other is Precalculated Progress Statistics. These are set as default so you can start the process quickly.

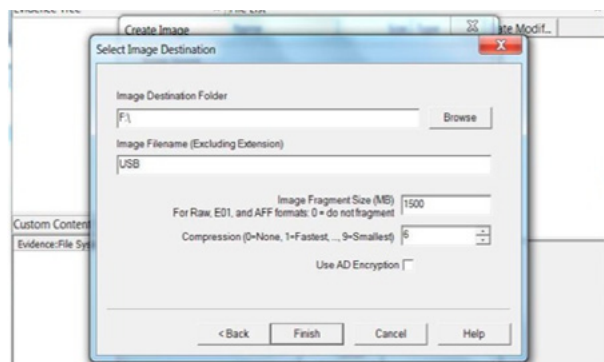


Figure 11. Select Image Destination screen shot

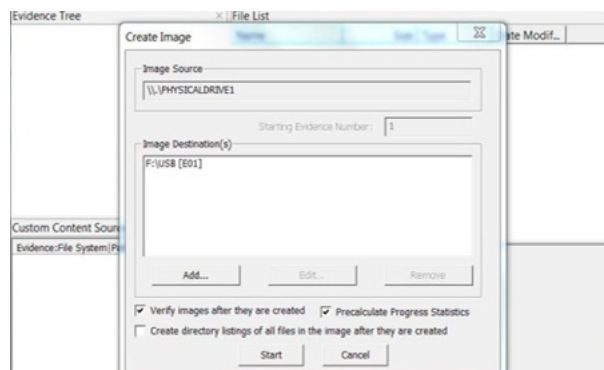


Figure 12. Create Image screen shot

(Creating the Image Process View) Image was created successfully. A status box will show throughout the imaging process to tell you when the image creation is complete and you can click on Image Summary to view the Craetion Log Evedence item.

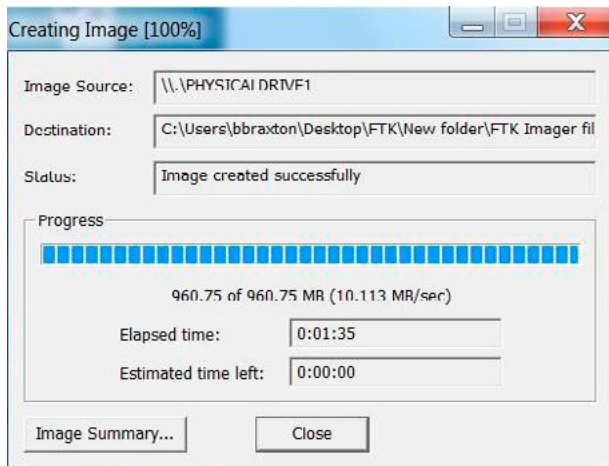


Figure 13. Image Created successfully screen shot

Once the image is complete and the file is on your desktop or file chosen you will have 2 images. One is and E01 card which is the image itself and the other is an E01 text file.

Once you open the text file you will see all of the information about the creation of the drive image. What version of imager created the file, the case information that you entered (the case number and name, the geometry information about the drive, the size the capacity, and some hash values). You can see the hash value of the drive image that was created, there are dates and times that show when that image case started and ended, and there is the verification after the imaging process was completed. That verification rehashes the image drive and then compares them to each other. Once completed you can see that both were verified and they ran two different hashes that verified the data had not been altered at all during the image processing and after the image was created, the image will be identical to the original drive. This is a very important part of verifying the integrity of your data. The E01 card gives you the summary of the imaging process and shows that all the hash values match.



Figure 14. E01 file in notepad screen shot

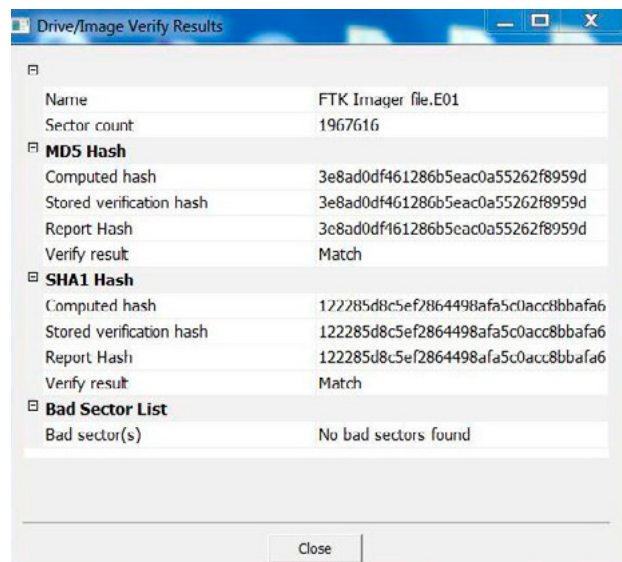


Figure 15. Drive/Image verifying results screen shot

FTK Imager has a couple of other features I would briefly like to touch on. One of them would be being able to mount a drive. This is a great feature if you have a large drive image and you want to take a look at the contents or preview them without actually loading it into FTK and you're on a time constraint. You can mount that image to a drive letter on your computer, just point to the drive image, open it and assign the next drive letter and it will mount it as read only as default on the next available drive. Then from there you click mount and you will see the drive letter available it will launch in, for this instance it is drive G.

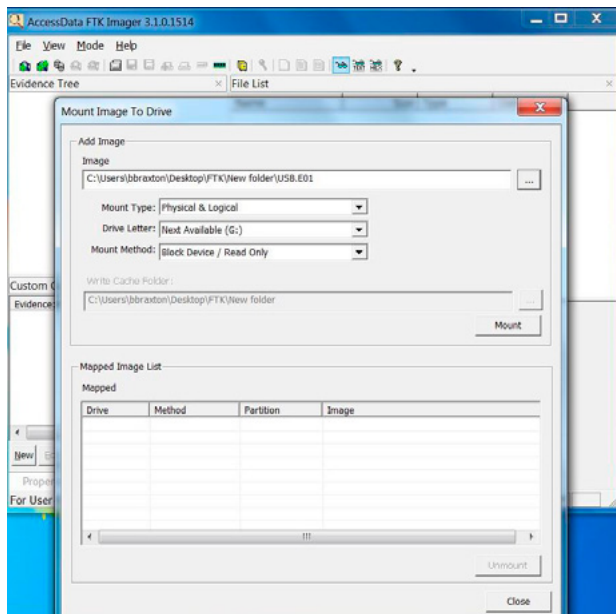


Figure 16. Mount Image to drive screen shot

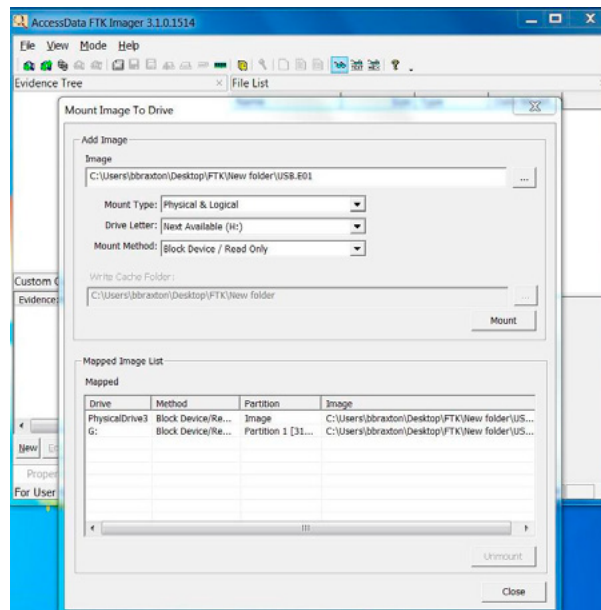


Figure 17. Mapped Image List screen shot

Now you can work through the drive image in a read only matter, you can navigate through the folders, launch files, documents, look at media files and images, and it will still protect the original integrity of the drive image.

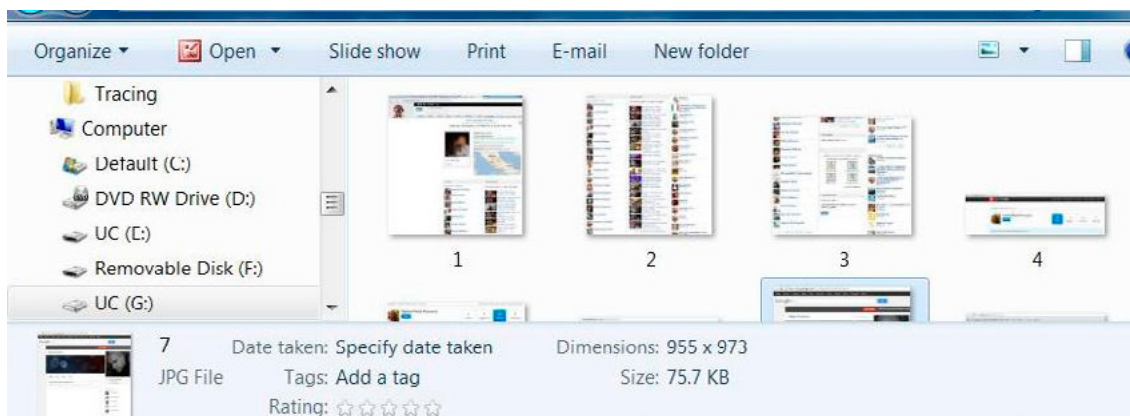


Figure 18. UC G drive view screen shot

Once you close the screen in imager, the drive will be removed and the image will dismount and no longer be available, and the image file will remain in the file folder it was saved in.

There are two other features I would like to highlight in FTK imager. One is the Capture Memory which allows us to image the ram memory of the current running computer. This is good when using FTK Imager-Lite from your thumb drive inserting into a computer that is going to be seized as evidence so that you can capture the contents of the RAM before that computer is shut down and removed to a laboratory or other evidence storage. In this case you simple have to point to where you want to put the file, it will give a default name that you can change and then just hit capture memory. Depending on how large the gigabits are it may take time because it is taking a forensics image of the ram of a running computer so that forensic image can be analyzed later.

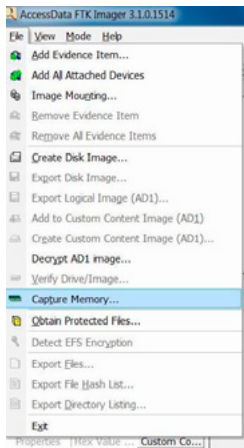


Figure 19. Capture Memory screen shot

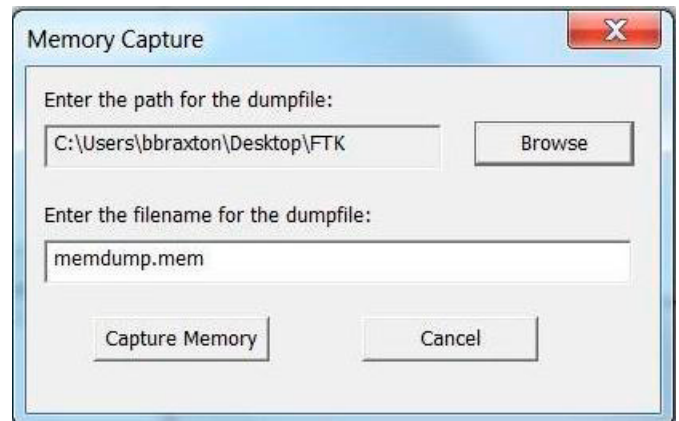


Figure 20. Memory Capture path and file name screen shot

Another feature I would like to touch on would be obtaining protected files this will allow you to export registry files of a running computer. Normally Windows Vista and Windows 7 will not allow access even copy and paste operations to be performed on the registry files of a running computer. FTK Imager is capable of getting past those protections and exporting them for you, you just simple need to choose a destination for the files, choose between the option of minimum files for login password recovery, (which are the minimum files needed to crack the password on the computer) in this case it would be either the SAM registry file and the System registry file, or Password recovery and all registry files. I depends on what evidence you are trying to retrieve but I would choose “ALL” this process is relatively quick and will show on the desktop.

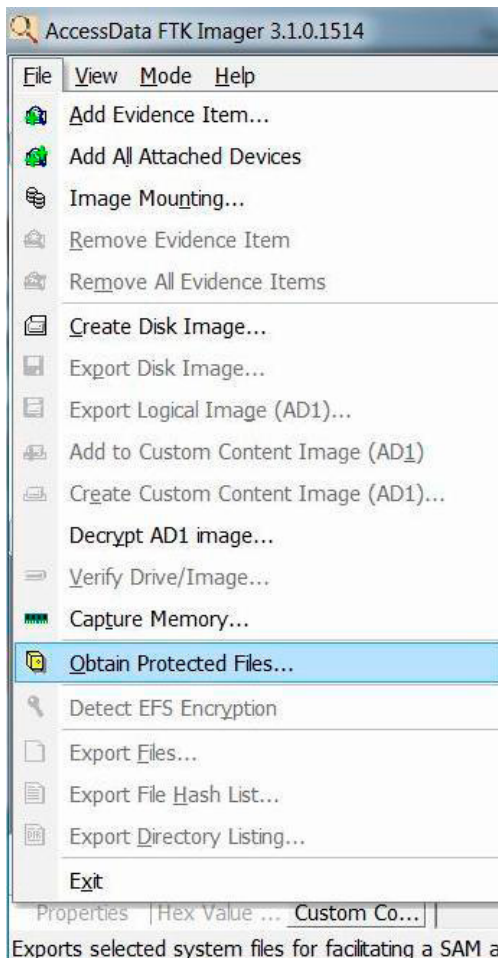


Figure 21. Obtaining Protected Files screen shot



Figure 22. Obtaining System Files screen shot

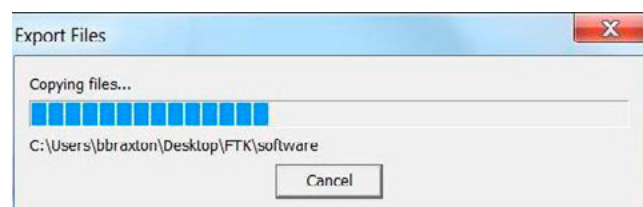


Figure 23. Exporting Files screen shot

Once it is complete you will find all the registry files that were exported. You will see the Security file, default file, system file, SAM file, Memdump file, software file, etc.

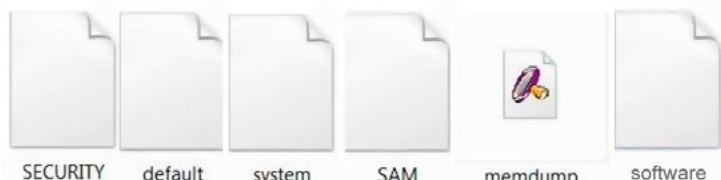


Figure 24. Registry Files that were exported screen shot

Now that you have a forensic image of the drive, it's time to analyze the evidence with the copy of retrieved data including deleted data, files and folders, jpg images, etc. During this time you must maintain the chain of custody when handling the evidence to ensure the integrity of the evidence is not compromised. This will show an audit trail of who accessed the data when it is presented in court. To do this you must follow the digital chain of custody, which includes creating a digital fingerprint by hashing all the data images.

A brief assignment I used Access Data FTK Imager lite from my Lexar 8G USB to create a physical image of the hard drive and captured the memdump and E01 files to the USB device. I then copied them over to my workstation and both hashes matched.

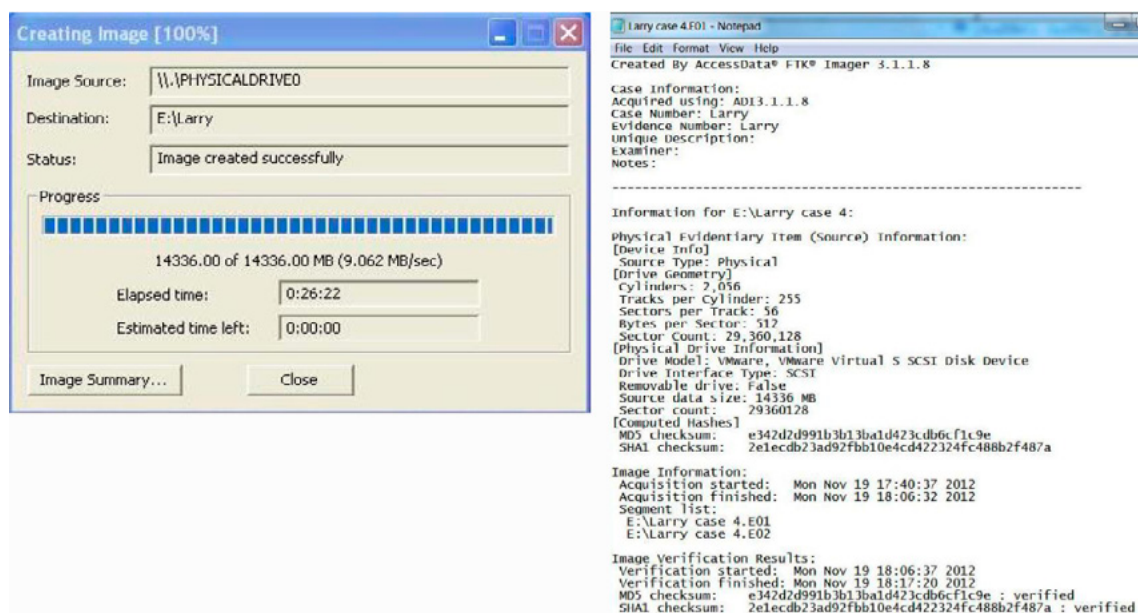


Figure 25. Creating a physical drive from a harddrive using FTK Imager-Lite screen shot

MD5 Matched

<input type="checkbox"/>	Name	Larry case 4.E01
<input type="checkbox"/>	Sector count	29360128
<input checked="" type="checkbox"/>	MD5 Hash	
<input type="checkbox"/>	Computed hash	e342d2d991b3b13ba1d423cdb6cf1c9e
<input type="checkbox"/>	Stored verification hash	e342d2d991b3b13ba1d423cdb6cf1c9e
<input type="checkbox"/>	Report Hash	e342d2d991b3b13ba1d423cdb6cf1c9e
<input type="checkbox"/>	Verify result	Match
<input checked="" type="checkbox"/>	SHA1 Hash	
<input type="checkbox"/>	Computed hash	2e1ecdb23ad92fbb10e4cd42324fc488b2f487a
<input type="checkbox"/>	Stored verification hash	2e1ecdb23ad92fbb10e4cd42324fc488b2f487a
<input type="checkbox"/>	Report Hash	2e1ecdb23ad92fbb10e4cd42324fc488b2f487a
<input type="checkbox"/>	Verify result	Match
<input checked="" type="checkbox"/>	Bad Sector List	
<input type="checkbox"/>	Bad sector(s)	No bad sectors found

Figure 26. Drive/Image MD5 verification results screen shot

Log File

```
[8/21/2011 14:06:38] LogFile Open.
[8/21/2011 14:06:38] Entering OCEntry; Component = <ims> (0)
[8/21/2011 14:06:38] Function = OC_PNEINITIALIZE (0), Param1 = 00000003 (3), Param2 = 00000000 (00000000)
[8/21/2011 14:06:38] Leaving OCEntry. Return=1
[8/21/2011 14:06:38] Entering OCEntry; Component = <ims> (0)
[8/21/2011 14:06:38] Function = OC_INIT_COMPONENT (1), Param1 = 00000000 (0), Param2 = 0018AB34 (0018AB34)
[8/21/2011 14:06:38] Entering unattended install mode
[8/21/2011 14:06:38] No other SMTP servers detected, installing IMS.
[8/21/2011 14:06:38] Leaving OCEntry. Return=0
[8/21/2011 14:06:39] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:06:39] Function = OC_QUERY_STATE (12), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:06:39] Original state is: DEFAULT
[8/21/2011 14:06:39] Leaving OCEntry. Return=0
[8/21/2011 14:06:39] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:06:39] Function = OC_CALC_DISK_SPACE (6), Param1 = 00000001 (1), Param2 = 01F6ABD0 (01F6ABD0)
[8/21/2011 14:06:39] Leaving OCEntry. Return=0
[8/21/2011 14:06:40] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:06:40] Function = OC_WIZARD_CREATED (16), Param1 = 00000000 (0), Param2 = 0017003A (0017003A)
[8/21/2011 14:06:40] Leaving OCEntry. Return=0
[8/21/2011 14:09:02] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:09:02] Function = OC_QUERY_STATE (12), Param1 = 00000001 (1), Param2 = 00000000 (00000000)
[8/21/2011 14:09:02] GetUnattendedModeFromSetupMode iis_smtp
[8/21/2011 14:09:02] SetupGetLineText failed (3758096642).
[8/21/2011 14:09:02] Leaving OCEntry. Return=0
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:10:17] Function = OC_QUEUE_FILE_OPS (7), Param1 = 00000000 (0), Param2 = 032349E0 (032349E0)
[8/21/2011 14:10:17] Leaving OCEntry. Return=0
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:10:17] Function = OC_QUEUE_FILE_OPS (7), Param1 = 00000000 (0), Param2 = 032349E0 (032349E0)
[8/21/2011 14:10:17] GetSubCompAction(): iis_smtp=AT_DO_NOTHING
[8/21/2011 14:10:17] Leaving OCEntry. Return=0
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:10:17] Function = OC_QUERY_STEP_COUNT (9), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:17] Leaving OCEntry. Return=2
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:10:17] Function = OC_QUERY_STEP_COUNT (9), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:17] Leaving OCEntry. Return=0
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:10:17] Function = OC_ABOUT_TO_COMMIT_QUEUE (14), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:17] Leaving OCEntry. Return=0
[8/21/2011 14:10:17] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:10:17] Function = OC_ABOUT_TO_COMMIT_QUEUE (14), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:17] GetSubCompAction(): iis_smtp=AT_DO_NOTHING
[8/21/2011 14:10:17] Leaving OCEntry. Return=0
[8/21/2011 14:10:43] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:10:43] Function = OC_COMPLETE_INSTALLATION (10), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:43] Leaving OCEntry. Return=0
[8/21/2011 14:10:56] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:10:56] Function = OC_COMPLETE_INSTALLATION (10), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:10:56] GetSubCompAction(): iis_smtp=AT_DO_NOTHING
[8/21/2011 14:10:56] Leaving OCEntry. Return=0
[8/21/2011 14:10:56] Entering OCEntry; Component = <ims> (0), Subcomponent = <iis_smtp> (0)
[8/21/2011 14:10:56] Function = OC_QUERY_STATE (12), Param1 = 00000002 (2), Param2 = 00000000 (00000000)
[8/21/2011 14:10:56] Leaving OCEntry. Return=0
[8/21/2011 14:11:15] Entering OCEntry; Component = <ims> (0), Subcomponent = <> (4)
[8/21/2011 14:11:15] Function = OC_CLEANUP (11), Param1 = 00000000 (0), Param2 = 00000000 (00000000)
[8/21/2011 14:11:15] Leaving OCEntry. Return=0
[8/21/2011 14:16:21] LogFile Close.
```

Figure 27. Log File screen shot

By using AccessData Imager using the mount image to drive, I was able to view all profiles. I then used FTK to explore the possibilities of using these items as evidence.

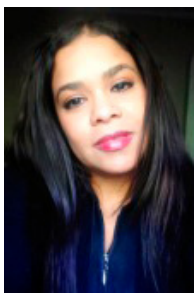
CONCLUSION

These are just the basics of using AccessData Imager and Imager Lite. They are great data imaging tools for investigators and analysis for imaging hard drives and taking a quick look at what's on the computer without leaving a trace they were actually there. The legal compilations come from data and evidence not being properly logged, search warrants that cover every piece of evidence retrieved like imaging hard drives, ensured data integrity, timeliness, accuracy, and forensics techniques training. It is crucial to show objectivity in a court, as well as the continuity and integrity of evidence, demonstrate how the evidence was recovered by showing each process of how the evidence was obtained. Make sure your evidence is preserved to the extent that a third party can repeat the same process you documented and arrive at the same result as that presented to a court. Electronic evidence is very fragile, it can be altered, damaged, destroyed, improperly handled and not well preserved and will render unusable or lead to an inaccurate result.

SOURCES

- <http://www.phillipdixon.net/PDForensics.pdf>
- http://www.downloadplex.com/Windows/Graphic-Apps/Other/accessdata-fts-imager_401591.html
- <http://digital-forensics.sans.org/blog/2009/06/18/forensics-101-acquiring-an-image-with-fts-imager/>
- <http://www.obsidianforensics.com/blog/imaging-using-fts-imager/>

ABOUT THE AUTHOR



Bridgette Braxton works at Jet Propulsion Laboratory/NASA in the Protective Services Division for over 9 years and has received an Exceptional Achievement Medal from NASA for coordinating requirements for offsite NASA Critical Flight Hardware, a Group Achievement Award for the Unified Security Application (USA), SPOT Award from Tracking & Radar Communications, and a Mariner Award for Critical Flight Hardware Exemptions. The Author's education background includes an Associate in Science in Criminal Investigations, Bachelors of Science in Criminal Justice with a concentration in Forensic Psychology (Summa Cum Laude) and two Masters of Science one in Cybersecurity Computer Forensics and the other Cyber Intelligence and holds several professional certifications and is currently enrolled at the University of Michigan for continuation classes in Network Security, Mobile Device Security, Intrusion Detection and Advanced Web Security. The author is also in the process of becoming a partner in Digital Background Investigation Services (DBIS) which analyzes forensic data and recovers digital evidence while preserving the integrity of the electronic evidence for discovery and trial.

UPDATE
NOW WITH
STIG
AUDITING

“IN SOME CASES
nipper studio
HAS VIRTUALLY
REMOVED
the **NEED FOR** a
MANUAL AUDIT”
CISCO SYSTEMS INC.

Titania's award winning Nipper Studio configuration auditing tool is helping security consultants and end-user organizations worldwide improve their network security. Its reports are more detailed than those typically produced by scanners, enabling you to maintain a higher level of vulnerability analysis in the intervals between penetration tests.

Now used in over 45 countries, Nipper Studio provides a thorough, fast & cost effective way to securely audit over 100 different types of network device. The NSA, FBI, DoD & U.S. Treasury already use it, so why not try it for free at www.titania.com



www.titania.com

FTK IMAGER ON THE FLY

by Robert C DeCicco

Practicing computer forensics often times means having to jump on a plane or in a car to get someplace quickly to collect evidence. In part, response to the often reactive nature of the work, agencies and firms have developed fly away kits, mobile labs or other solutions that are prepped and ready to go and can handle a variety of environments or evidence types. More often than not they are above and beyond the necessity of the operation and of course there are issues with the weight, unwieldiness and overkill that often accompanies being 'over' prepared.

What you will learn:

- a convenient tool to collect data in a forensically sound manner without a full kit
- basic interface and features of FTK Imager
- types of situations or circumstances where the necessity to collect evidence soundly occur without notice

What you should know:

- rules of evidence handling,
- foundation principles of computer forensics,
- general operating system architecture
- basic hardware/software operator knowledge

What about when you're not prepared for a collection? What about those instances where you may be only scheduled to attend a meeting or scoping exercise at a client site? Or the investigation has changed and the window of opportunity is closing rapidly that you have no time to call up additional backup, resources or drop ship equipment?

Chances are you have a laptop with you even when attending a meeting. If so for the additional weight of carrying a thumb drive you can have a fully functioning forensic collection software with you that is both defensible and packs a lot of punch.

FTK IMAGER MAIN FEATURES

This tool can create forensically sound images of hard drives, disks, USB drives, NAS devices entire folders, directories or even of individual files from various places within the media storage device. The ability to export files and folders from the created image means that this application can also recover data on its own (in some circumstances). Therefore, it can do more than just allow previewing that data for the sake of preparing recovery procedures which usually involve other sophisticated tools. The main purpose remains disk imaging. It's also free so even if you don't have it with you on a thumb drive you can download it from the web provided you have access. All that in mind it is still simply a tool. Many people can use a chainsaw but you need proper training and time practicing to be able to cut properly and not cause damage and even more time and practice to be able to make something like an ice sculpture. Proper training and knowledge can't be stressed enough. The functionality of the software and the process are only as defensible as

the practitioner in control. The two are co-dependent when it comes to standing up to scrutiny or opposition related to the admissibility or defensibility.

OTHER FEATURES OF FTK IMAGER

Some other built-in features and functionality of FTK Imager include:

- Live Memory Acquisition
- Ability to be run “live” on suspect machines
- Acquisition of protected system files
- Acquisition of Registry hives
- Identification of EFS encryption
- When run live, it can be used to image mounted encrypted partitions
- The ability to mount a variety of forensic image formats as logical, physical and file system volumes (including as write-cached)
- Handle Linux variants
- Handle OSx variants

The User Guide built in to the installed application when you hit F1 provides specific details on all features and how to use them effectively.

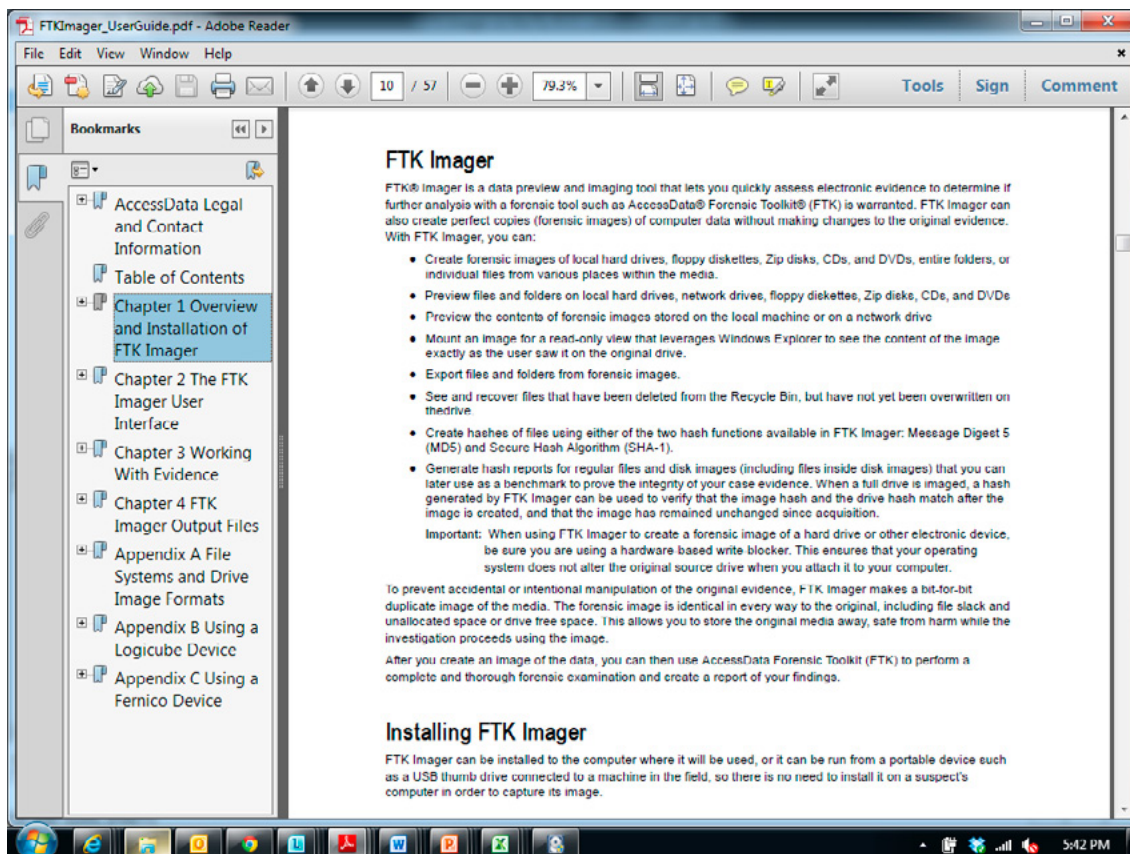


Figure 1. FTK Imager User Guide and Startup Page

Once you’ve decided what you need to collect and have documented appropriately on a worksheet or chain of custody all the pertinent information you can conduct the operation quickly and without altering the source environment. The evidence is ‘containerized’ in a forensic manner and you also have the ability to further encrypt the data prior to transport and to protect any deliberate or inadvertent access to the sensitive information by anyone unintended.

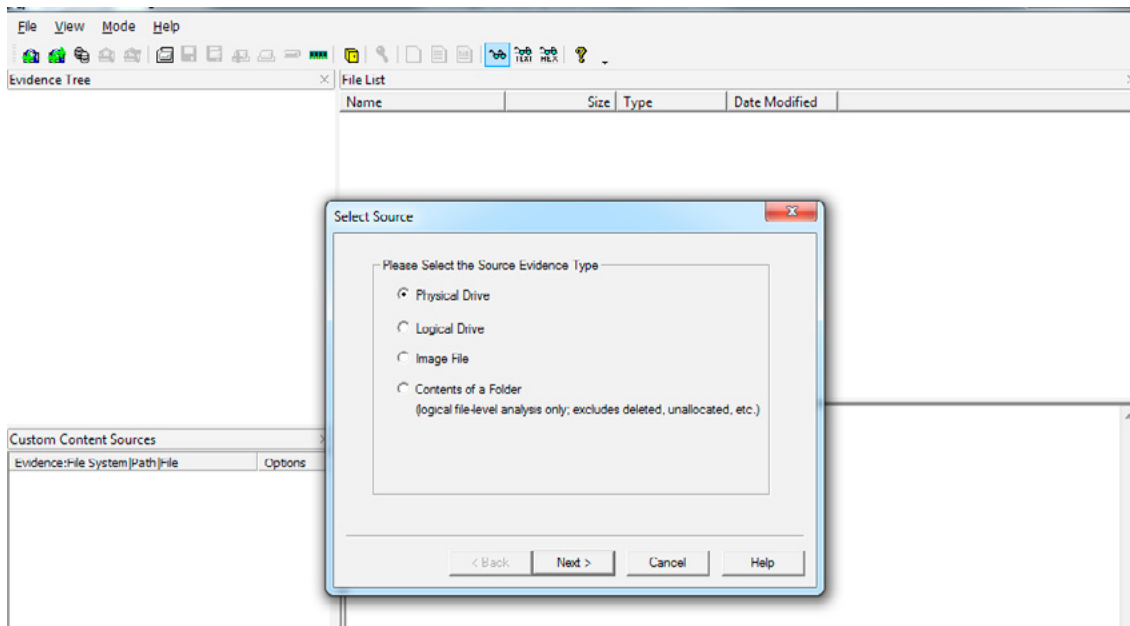


Figure 2. Source Selection - + Add Evidence Item Prompt

If you've hopefully worked with forensics tools like EnCase you'll have a relatively easy time navigating the interface, understanding what is happening and creating appropriate evidence files. Another great feature of FTK Imager is that you can even create EnCase E01's aside from SMART format, AFF or Raw (dd) images, which can be viewed, analyzed or exported from inside of EnCase at a later time. The software affords you the ability to do an entire full disk image physically or select logical extractions from the target digital environment in a simple and intuitive GUI. In addition once you've attached the target, you'll almost immediately be previewing a hard drive (including in-tact deleted files) or when viewing a networked environment or you'll have the ability to navigate through directories on your own or with the assistance of counsel or the user if there is a necessity to do only a targeted collection.

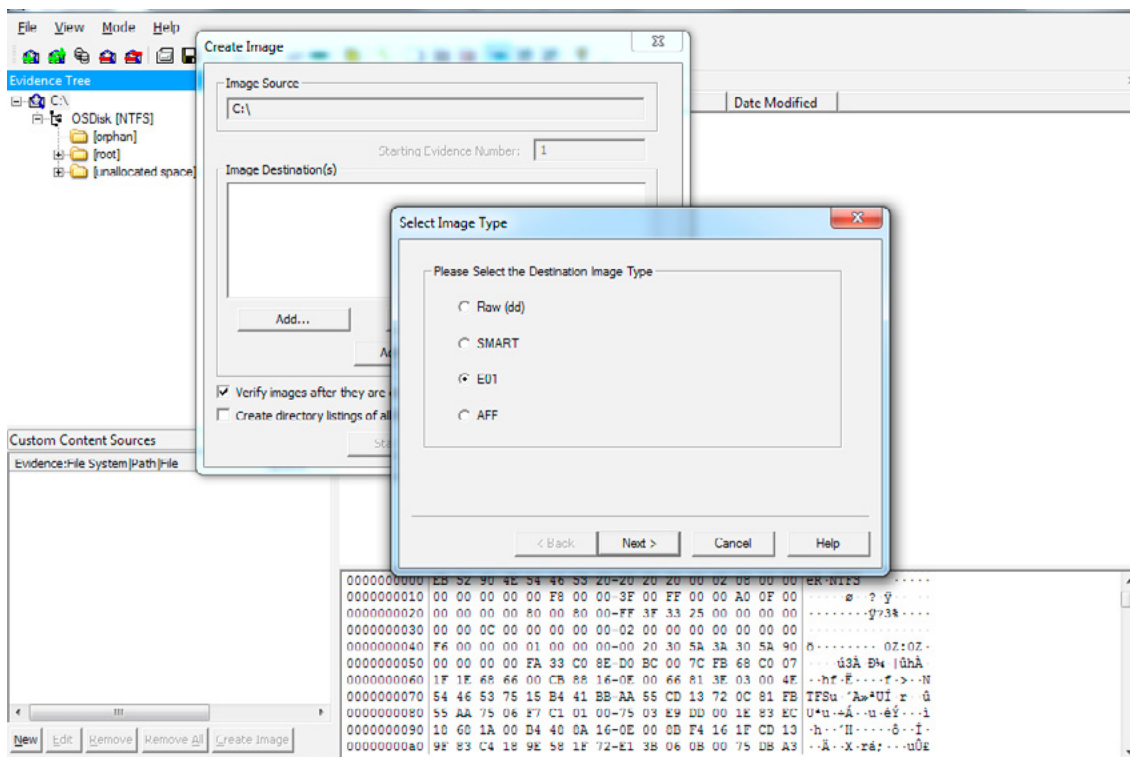


Figure 3. Selecting Evidence Image Type FTK Imager

While your chain of custody or worksheet should always be thorough, the case information block for FTK Imager allows for notes and locked in fields that stay with the image. Much more detail can be provided as a reminder or for the next examiner as opposed to a cloning device.

EVIDENCE ACQUISITION PHASE

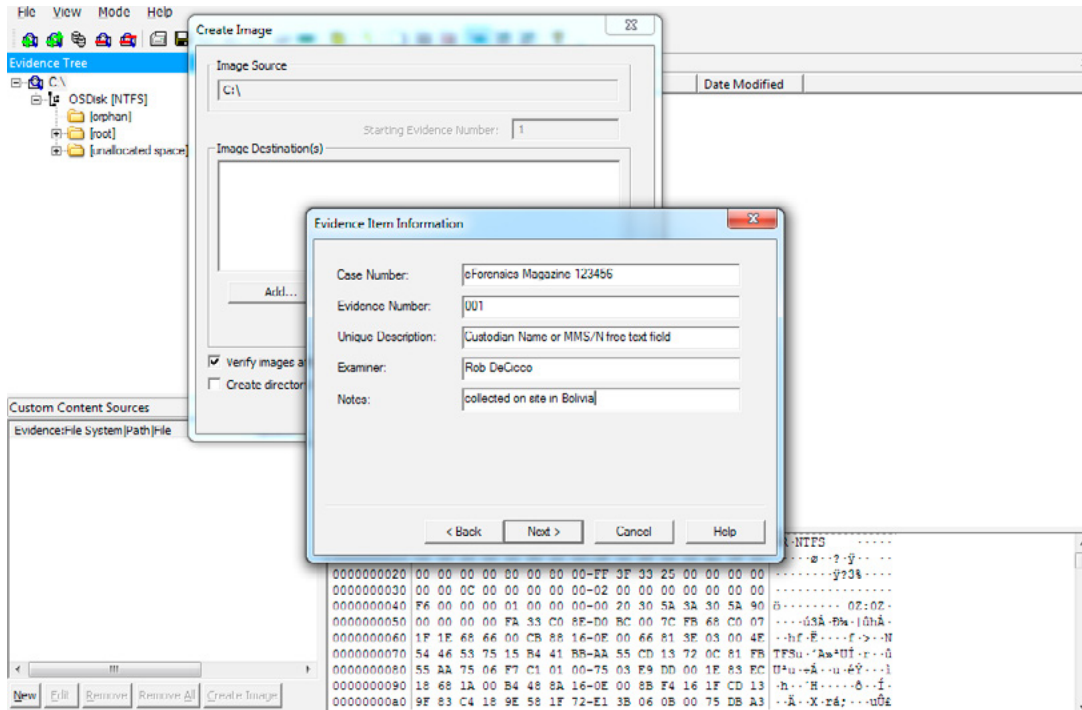


Figure 4. Evidence Item Information Entry Page FTK Imager

After you point FTK Imager to where the evidence needs to go, name the evidence image file itself, select compression and fragment size for the image and the option for whether to encrypt on the fly within FTK Imager.

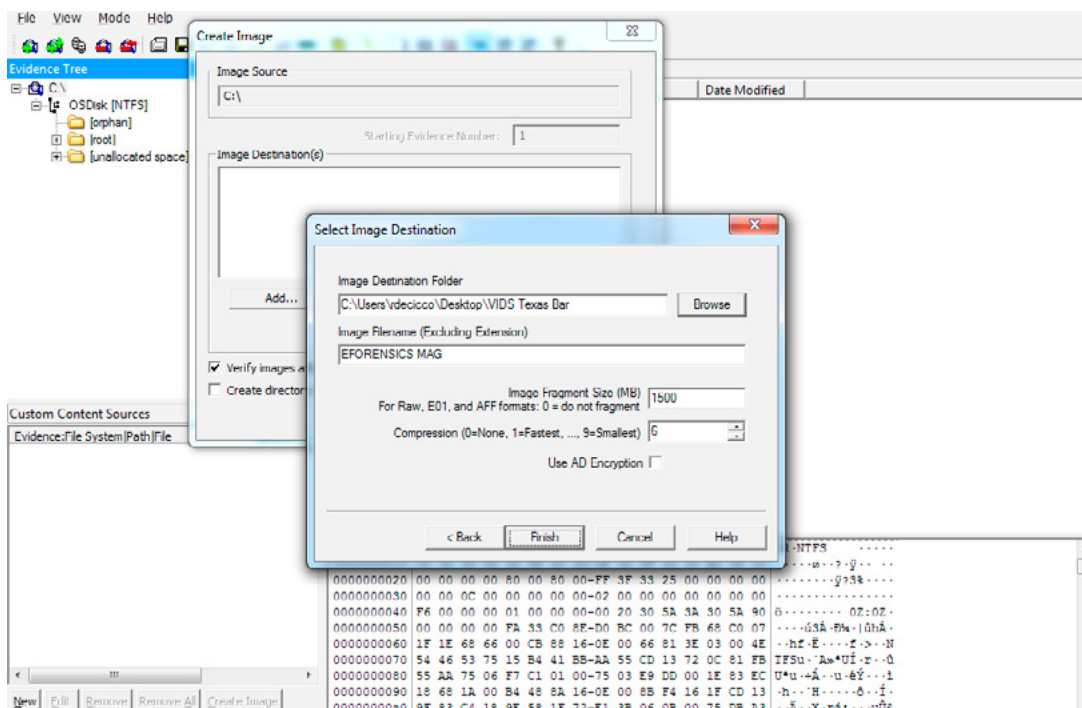


Figure 5. Imaging Final Stage Prompt

Lastly, my favorite and critical feature for FTK Imager is that it also automatically creates a log of the acquisition process and places it in the same directory as the image, *image-name.txt*. This file lists the evidence information, details of the drive, check sums and dates and times the image acquisition started and finished. This information stays with the evidence and is always helpful to have to cross check with field documentation or if the field documentation finds itself separated from the evidence. In the event that sectors are skipped or corrupt or the image fails to verify, that 'bad news' is logged as well.

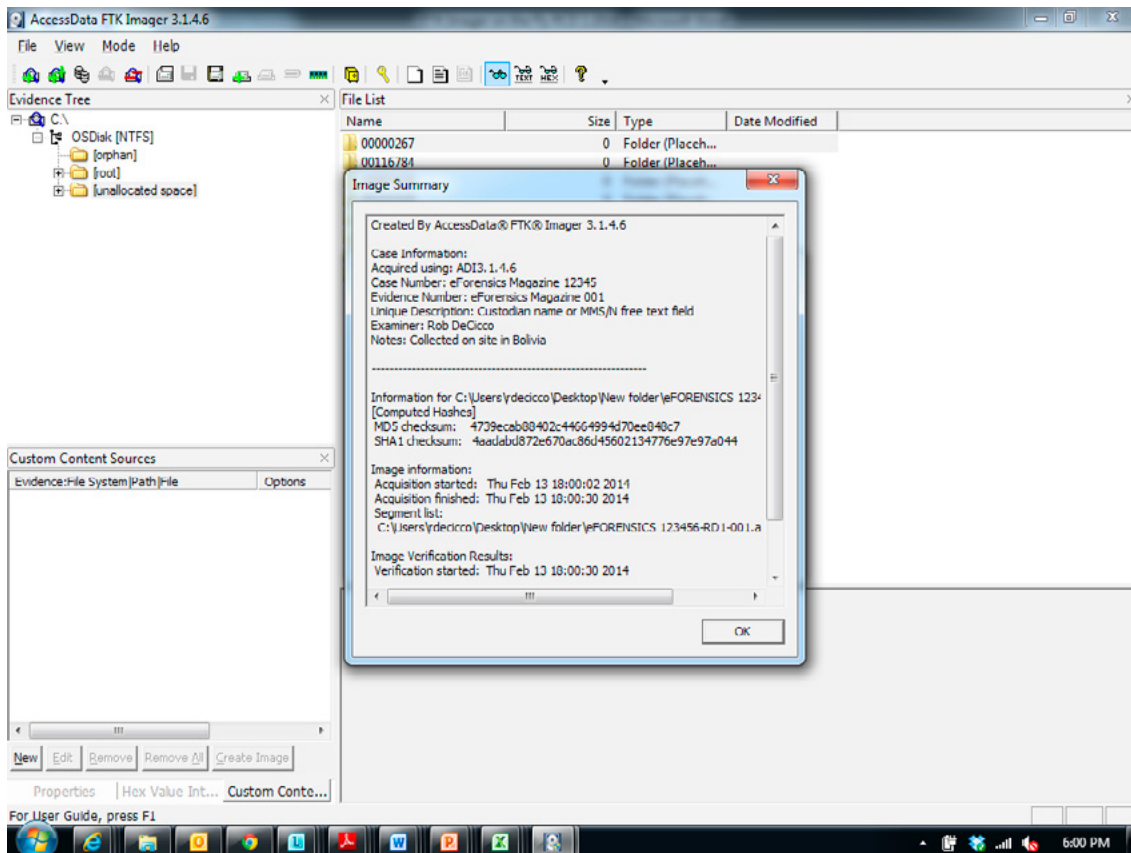


Figure 6. Verification Page / Image Summary FTK Imager

Overall, FTK Imager an excellent tool that gets the job done when in a pinch or even as a tool to add to your comprehensive field acquisition and analysis kit.

CASE EXAMPLE

Some time ago I was scheduled to attend a meeting with outside counsel at a large multinational corporation in Northern California in relation to an investigation related to some very specific, previously identified employees. The meeting was set up for the morning and was just supposed to be with the internal general counsel and 2 of the identified individuals in the investigation. Our role was primarily to conduct interviews and take notes and determine the scope of what data may need to be collected at a later date and reviewed. Because the meeting was so specific and defined, I did not bring any equipment with me nor had I had anyone on standby. In addition, our firm at the time was completely tapped out with personnel in the field in any event and it would have been several days if I needed to get anything to the client location.

Naturally, during the course of the interview the names of 3 of what counsel determined were, extremely relevant individuals surfaced and it was decided that collecting information from all their assigned data sources would be prudent. There was one hitch. 2 of the 3 were heading overseas on assignment for the next 6 months the next morning and would be taking their laptops with them. After a brief conversation with IT management it was determined that the majority of information on their hard drives of their laptops was stored locally and not synched to a server nor backed up and that the investigation would have the need for the review of deleted file space, I needed to image the drives in a forensically sound, complete and defensible manner.

Without the accompaniment of human resources nor equipment and being far from our closest regional lab, I was placed in a position to improvise. I found a 'major electronics retailer' and purchased 2 x 500GB USB external drives for storage and I then employed the use of FTK Imager which I keep on a thumb drive in my laptop bag along with some other handy software tools. During the next few hours while other negotiations took place I was able to obtain complete physical images of 2 x 120GB hard drives from the relevant custodians without having to wait months to gain access at the site, travel to where they were headed nor disrupt their work flow by seizing the machines and/or issuing new ones.

The next day our team was able to add the complete digital evidence to our processing and case review with confidence in the integrity of the data with almost no disruption to the client. Additionally, as issues arose and escalation beyond traditional keyword searching or document reviews we introduced, we were able to conduct more thorough analyses on complete, forensically sound images containing all unallocated space, Registry hives, swap files and operating system files necessary to be able to form opinions with a reasonable degree of scientific certainty and submission to court. This was a stellar example of, for lack of a better description, FTK Imager 'saving the day'.

REFERENCES

Access Data FTK Imager software Version 3.1.4.6

ABOUT THE AUTHOR

Robert DeCicco is a Managing Director and the Leader of the Computer Forensics Practice of the Legal Technology Solutions group at Navigant Consulting. He specializes in providing services related to data acquisition and forensic examinations for a variety of high-profile and confidential clients. His responsibilities span from hands-on examination and analysis of computer data to the day to day management of the forensics staff and the enforcement of policies and procedures. DeCicco has actively managed and participated in the collection and the subsequent analysis and processing of electronic evidence from extremely complex, disparate and sensitive technology environments throughout the world.

Mr. DeCicco is a court accepted expert in the field of computer forensics and has provided expert reports, rebuttals and defended opinions related to computer forensics and electronic discovery in both civil and criminal matters and in both state and federal courts throughout the US and abroad. Mr. DeCicco is a published author and respected speaker on the topic of computer forensics and has authored protocols for electronic information document reviews which have been approved by Special Masters to court and has defended the steps and procedures in court and officially transcribed sessions.



cutting through complexity

Are you prepared?

kpmg.ca/forensic

INTRUSION

ATTACK • THREAT • CYBER SECURITY

TECHNOLOGY • CORPORATE

ELECTRONIC • INFORMATION • COMPLEXITY

DATA ANALYTICS

RISK • INFORMATION • TECHNOLOGY

DATA RECOVERY

COMPLEXITY • ELECTRONIC • INFORMATION

FORENSICS

DATABASE • ELECTRONIC • CONTROL

INTELLIGENCE

INFORMATION • RISK • TECHNOLOGY

eDISCOVERY

COMPLEXITY • THREAT • INTELLIGENCE

INVESTIGATIONS

TECHNOLOGY

COMPLEXITY • THREAT • DATABASE

INTELLIGENCE • PROTECTION

CORPORATE

HIDING INFORMATION THEFT: HOW TO FIND VIDENCE OF DATA THEFT

by Mark Garnett

With the advent of computer systems, new evidence is now available for any investigator to find if an offender has used a computer system to facilitate an offence. With this new found treasure trove of information there is a downside, even though evidence may exist, it can be like searching a needle in a haystack. Not only is data volume an issue, but as computer users become more and more savvy they are finding more novel ways to make a computer examiners life harder. They can do this by destroying or hiding crucial information.

What you will learn:

- Forensic imaging techniques
- MFT file structure
- MFT file entry recovery
- How people attempt to hide and/or delete data
- The importance of file system metadata

What you should know:

- Basic forensic data acquisition methodologies
- File signature analysis
- Investigative techniques
- Data recovery

It is fair to say that most of today's computer users know that when they "delete" a file from a computer system, it is not really deleted. As a result, they have that repository of all answers, Google, available to them that they can use to research ways in which to cover their tracks and prevent computer examiners from finding evidence that may get left behind from any wrongdoing. Whilst this may be the case, computer systems and operating systems are extremely complicated systems and there are generally many exceptions to every procedure used to remove evidence and as a result, it is still common for these procedures to not be one hundred per cent successful.

This article will discuss one such case, a case of stolen intellectual property in which the suspect was under the mistaken belief that he had done all that was necessary to remove any evidence that he had actually stolen intellectual property. This case is particularly interesting as the stolen intellectual property, consisting of electronic documents, was never actually found. All of the critical evidence consisted of information that inferred the previous existence of the intellectual property, but the offender was successful in removing evidence of the documents themselves.

THE SCENE

My client consisted of a large corporate entity that had just received the resignation of a very senior sales executive, who had decided to commence work with a competitor organisation. This particular executive was very well respected within the organisation and was considered an exemplary employee and executive manager. Like all senior executives, he was issued with a laptop computer and a large external USB external hard disk drive on which to store documents. This executive was expected to work away from the office from time to time and as a result, he needed the ability to have a portable storage device. This of itself was no cause for alarm for the business and was considered standard operating procedure.

When the executive resigned, senior management, as a matter of process, decided to review the executive's laptop computer just to make sure that he had not sent any confidential information to his new employer. In addition to this, they also wanted to confirm that he had not stolen or unlawfully removed any confidential documents from the business prior to resigning.

As a result, representatives of my client attended the executive's home to collect all equipment currently issued to him. At this time, the executive returned his laptop computer but not the external USB storage device. The company had failed to realise, at this time, that the executive was actually issued with the storage device and as a result, the representatives that attended at his home did not realise that this piece of equipment was not returned.

The following day, the senior executive contacted his ex-employer and indicated that he had forgotten to return the external storage device and asked if somebody could attend at his premises to collect it.

Both the laptop computer and the external hard disk drive were subsequently provided to me for forensic analysis. It is relevant to note here that the organisation configured all computer systems with the Microsoft Windows XP operating system using the NTFS file system.

THE DATA CAPTURE

My initial brief was to undertake an analysis of both the laptop computer and electronic storage device and determine what, if any, intellectual property had been unlawfully removed from the organisation. I commenced my analysis using FTK Imager, which I used to obtain forensic images of both hard disk drives. FTK Imager is easy to use, robust, lightweight (in terms of computer resource requirements) and very reliable.

Selecting "File->Create Disk Image", I chose the "Physical Drive" option in the subsequent dialog box. In my view, imaging a physical drive in instances where you are undertaking an analysis of a single disk drive system (i.e. laptop, desktop or external storage device) is much more preferable than logical imaging in cases such as these.

In short, physical disk imaging is a process in which a "bit for bit" or exact copy of the entire contents of the hard disk drive is obtained in such a manner so as to:

- Prevent any alteration to the original data;
- Create a true copy of the source drive; and
- Capture all data, including current, deleted and unallocated data.

Current data includes all files that currently exist on the hard disk drive and are "visible" to the operating system. Deleted data includes those files that have been previously deleted by a user or through the normal operation of the computer and have not yet had any portion, including filename entries, overwritten by another file. The unallocated portion of a hard disk drive is space that is marked as available for use by the operating system. This area can contain data that has previously been deleted.

As much evidence relating to the operation of a Microsoft Windows file system is very volatile, it is not uncommon to find critical information about file activity in the unallocated portion of the computer's file system. As I am sure all readers are aware, logical disk images do not capture "deleted" information.

Having said this, there are some instances where physical disk images are simply not necessary or are not desired by the client. Physical or logical imaging of multi-disk storage systems is a matter for each engagement and the merits of each approach should be considered prior to determining what type of image is required.

With respect to image type, I chose “Raw/dd” as this image can be mounted, if required, to behave as an attached disk drive. Mounting images has the advantage of allowing an examiner to navigate the forensic image as if it was an attached disk drive. Whilst not applicable in this particular case, it is good to get into the habit of choosing an image format that provides you the most flexibility in conducting your examination.

After selecting an image format, I completed the necessary evidence information dialog box with the details of my particular case. I have noticed some examiners fill in “placeholder” text into these fields however I cannot stress enough the advantage in completing this dialog box in full with as much information as possible. Whilst an examiner may be able to track a small number of cases, it does not take long for this to become completely unwieldy. Once completed, choose a destination in which to save the forensic image, along with the image fragment size and compression format. These options are open to interpretation and most examiners will have their own preferred size and compression format with which they are comfortable.

As with all forensic matters, the use of a hardware write-blocking device was crucial in obtaining a pristine copy of the source data. A hardware write-blocking device ensures that no information is altered on the source hard drive when it is connected to your forensic machine for the purposes of imaging. Operating systems, such as Microsoft Windows, will routinely alter, however slightly, the data contained on a disk drive when connected to it via USB, for example. Even though the changes may be slight, they may be critical if they inadvertently destroy data that you could otherwise rely on.

In this particular case, there were no issues experienced during imaging process and this was confirmed when completed by selecting the “File->Verify Image” from within FTK Imager. The “Verify Image” option ensures that the stored hash value inside the image matches the computed hash value at the completion of the forensic verification process.

Prior to returning the equipment, other information was obtained such as the accuracy of the clock contained within the laptop. This is an important step if relying on the dates and times associated with files.

THE ANALYSIS

I commenced my analysis by examining the forensic image of the USB external storage device. I noted upon previewing the image that the external storage device contained no user files and had been previously formatted the day prior to it being provided to me for analysis.

The process of formatting a disk for use does not actually delete any “data” contained in files on the disk itself. The process simply recreates a series of system files and discards the old system files that were present before the formatting took place. All of the data that existed on the disk would now reside in the unallocated area of the disk.

During the formatting process, the Microsoft Windows operating system prepares the media by writing a series of system files to the disk that assist the operating system to store files and organise data on the disk. One such file written to the disk at this time is called the *Master File Table* (MFT). The MFT records information about files and directories stored on the disk including the name of the file, the date the file was created and where on the hard disk the file resides. Each file and directory contained on a hard disk has an entry in the MFT.

Each time a disk is formatted, new system files are created and the old system files, such as the old MFT file, are discarded. When examining the actual data contained within the MFT, each entry contained within it is preceded by the characters “FILE” followed by information about the file or directory, such as the file name. This also enables the easy identification and location of MFT entries in unallocated space.

The below screenshot provides an overview of the type of information you can expect to find:

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00335117B0	03	B0	02	00	FC	0F	03	B0	02	00	FC	0F	00	00	00	00	.°...ü...°...ü....
00335117C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00335117D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00335117E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00335117F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	02	00
0033511800	46	49	4C	45	30	00	03	00	09	9A	40	7B	09	00	00	00	FILE0...š@{....
0033511810	05	00	02	00	38	00	01	00	F8	01	00	00	00	04	00	00ø.....
0033511820	00	00	00	00	00	00	00	00	04	00	00	00	00	46	48	02	00
0033511830	02	00	00	00	00	00	00	00	10	00	00	00	60	00	00	00FH..
0033511840	00	00	00	00	00	00	00	00	48	00	00	00	18	00	00	00H.....
0033511850	95	17	04	87	02	C5	CE	01	0B	78	DF	86	02	C5	CE	01	...+.Äf...xB+.Äf.
0033511860	0B	78	DF	86	02	C5	CE	01	A5	3E	04	87	02	C5	CE	01	...xB+.Äf.¥>+.Äf.
0033511870	20	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0033511880	00	00	00	00	B2	06	00	00	00	00	00	00	00	00	00	00°.....
0033511890	A8	94	6A	A4	01	00	00	00	30	00	00	00	78	00	00	00	~"j&....0...x...
00335118A0	00	00	00	00	00	00	03	00	5A	00	00	00	18	00	01	00Z.....
00335118B0	70	7E	01	00	00	00	06	00	95	17	04	87	02	C5	CE	01	p~.....+.Äf.
00335118C0	95	17	04	87	02	C5	CE	01	95	17	04	87	02	C5	CE	01	...+.Äf...+.Äf.
00335118D0	95	17	04	87	02	C5	CE	01	00	00	01	00	00	00	00	00	...+.Äf.....
00335118E0	00	00	00	00	00	00	00	00	20	08	00	00	00	00	00	00
00335118F0	0C	02	48	00	45	00	4C	00	50	00	49	00	4E	00	7E	00	..H.E.L.P.I.N.~.
0033511900	32	00	2E	00	37	00	5A	00	49	00	65	00	6E	00	67	00	2...7.Z.I.e.n.g.
0033511910	30	00	00	00	90	00	00	00	00	00	00	00	00	00	02	00	0.....
0033511920	72	00	00	00	18	00	01	00	70	7E	01	00	00	00	06	00	r.....p~.....
0033511930	95	17	04	87	02	C5	CE	01	95	17	04	87	02	C5	CE	01	...+.Äf...+.Äf.
0033511940	95	17	04	87	02	C5	CE	01	95	17	04	87	02	C5	CE	01	...+.Äf...+.Äf.
0033511950	00	00	01	00	00	00	00	00	00	00	00	00	00	00	00	00
0033511960	20	08	00	00	00	00	00	00	18	01	68	00	65	00	6C	00h.e.l.
0033511970	70	00	69	00	6E	00	73	00	74	00	61	00	6C	00	6C	00	p.i.n.s.t.a.l.l.
0033511980	5F	00	65	00	6E	00	67	00	6C	00	69	00	73	00	68	00	..e.n.g.l.i.s.h.
0033511990	2E	00	37	00	7A	00	69	00	70	00	00	00	00	00	00	00	..7.z.i.p.....
00335119A0	80	00	00	00	50	00	00	00	01	00	00	00	01	00	01	00	€...P.....
00335119B0	00	00	00	00	00	00	00	00	0F	00	00	00	00	00	00	00
00335119C0	48	00	04	00	00	00	00	00	00	00	01	00	00	00	00	00	H.....
00335119D0	3E	09	00	00	00	00	00	00	00	00	00	00	00	00	00	00	>.....
00335119E0	00	00	01	00	00	00	00	00	41	10	EC	F6	50	06	00	00A.iöP...
00335119F0	FF	FF	FF	FF	82	79	47	11	00	00	00	00	00	00	02	00	ÿÿÿÿ,ÿG.....
0033511A00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Figure 1. Type of information you can expect to find

The header for the MFT entry “FILE0” can be clearly seen along with the filename, in this instance, “helpinstall_english.7zip”.

I determined that the disk drive had been formatted the day before being provided to me by examining the date and time that the MFT was created. Generally speaking, the date and time that the MFT was created is a good guide with respect to the date and time that a disk was formatted. This is dependent on the accuracy of the clock in the computer on which the disk drive was formatted.

This of itself was again not a blatant red flag as the senior executive had indicated to his employer that he had not used the external device to store any company files, only personal information, such as photographs and personal documents. As a result he said he reformatted the disk drive to remove any trace of the personal data prior to returning the equipment. Acting within my instructions from the legal team, I commenced a search of the external storage device in order to locate any files that may have been created using Microsoft office or Adobe Acrobat and previously stored on the external hard disk drive.

A very quick “scroll” through the information contained on the external storage device revealed that it did contain “old” data in unallocated space, although from my initial view, I was unable to locate much, if any, “humanly readable” text. The data all looked as if it had been compressed or once formed part of binary files.

I subsequently used my analysis software to look for any previous files on the storage device that contained headers common to documents created using Microsoft Office or Adobe Acrobat. The organisation in question was using a version of Microsoft Office prior to 2010 and as a result, I was not concerned with any documents created using the (then) new Office Open XML format. In particular, the header I was searching for consisted of the hexadecimal bytes “D0 CF 11 E0 A1 B1 1A E1” for Office files and “25 50 44 46” for Adobe Acrobat files.

In addition to this, I also searched for Lotus Notes databases (bytes 1A 00 00 04 00 00) (the organisation in question used Lotus Notes for email) and compressed archives, such as ZIP (bytes 50 4B 03 04) and RAR (bytes 52 61 72 21 1A 07 00). As outlined earlier, I noticed large amounts of what looked like compressed data on the disk from a cursory visual inspection. Expecting to see results when my search was completed, I was surprised when I did not locate any instances of the above file types whatsoever.

As a result, I subsequently undertook a search of the forensic image for file extensions in the event that files may not be present, but file name entries may be, for example in an MFT that existed on the hard disk drive prior to it being reformatted. I used simple search terms for this purpose, such as “doc”, “xls”, “pdf”, “ppt” etc.

I noted that I identified several thousands of instances of filename entries. Upon closer examination, I noted that each of these filename entries existed in, what appeared to be, entries in a previously deleted MFT. The MFT is structured in a particular way and consists of a series of entries in the following segments:

Standard Information
File/Directory Name
Data/Index
Unused Space

Figure 2. MTF Structure

The following figure also illustrates the structure of an MFT entry:

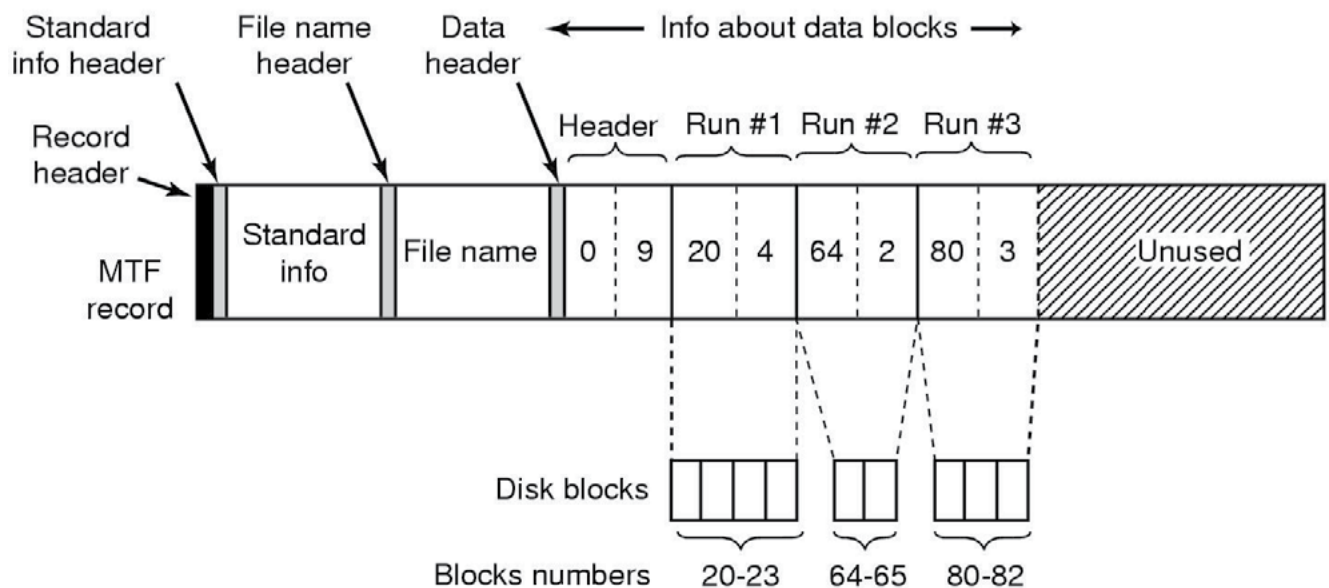


Figure 3. MTF Structure. Source: http://www.cs.bgu.ac.il/~os042/assignments/Ex3/NTFS_background.html

The “standard Information” segment consists of several data items including, but not limited to, the entry header (i.e. FILE), the allocated size of the MFT record (i.e. 1024 bytes) and a flag indicating whether the item is a file or directory. I note that this is not an exhaustive list, the totality of which is beyond the scope of this article. The file or directory name is stored as Unicode text and the data/index information consists of various aspects of the file or directory. For the purposes of this case study, it sufficient to understand that the following relevant information is stored in the “data block” of the MFT entry:

- The file creation date, stored as a 64 bit Little Endian format number. For example, the hex bytes 40 29 AF 60 6C 50 C7 01 would decode to 14 February 2007 at 19:41 hours UTC time;
- The date and time that a file was modified, again stored as a 64 bit Little Endian value; and
- The date and time that the MFT entry relating to this particular file was last changed.

By decoding the MFT entries identified as a result of my searches, I was able to generate a list of approximately 11,000 files that, up until the day before the portable hard disk drive was returned, existed on the device. By cross-referencing the file attribute information outlined above with file listings from the corporate file servers, I was able to match those files which appeared to have been copied from the business servers onto the portable device.

THE RESULTS

Now that it appeared as if the senior executive had acted unlawfully, next came the task of attempting to determine what actions he had performed in order to remove the information from the business. Given that now it was possible to prove that corporate data existed on the portable device, I now had to determine what the senior executive had done in order to “cover his tracks”.

I subsequently undertook a closer review of, what appeared to be, binary data on the hard disk drive. I determined that the information that I was viewing appeared to be compressed video data and sought to extract the data to determine if this was correct. I manually carved entire “chunks” of data from the unallocated portion of the portable hard disk drive and noted that, when saved as a discrete file, was recognised as a video by an MPEG video player and appeared to be recording of a popular British television comedy.

Upon further analysis, the same series of bytes, approximately one hundred megabytes of data, was repeated consistently across the entire length of unallocated space. This meant that, prior to the portable hard disk drive being formatted it contained hundreds of duplicate files of the same television episode.

I also undertook an analysis of the laptop computer provided by the senior executive. Upon an examination of the recycle bin on that computer, as opposed to the portable hard disk drive, I identified a large volume of video file entries with names such as “De785.mpg”, as well as documents created with Microsoft Office and Adobe Acrobat, such as “De621.doc”.

When a file is deleted by a user using an operating system, such as Windows XP on a disk drive formatted using the NTFS file system, by dragging and dropping the file into the recycle bin, the following process is carried out by the operating system:

- The file is moved to the recycle bin by the operating system; and
- When the user empties the recycle bin, the file is renamed using the convention “d” + drive letter of origin + unique index number. For example the file name “De785.mpg” means the file was emptied from the recycle bin it originally existed on the “E” drive and was provided with the index number “785”.

All portable hard disk drives connected to a computer system are given a unique drive letter by that computer system. It is my experience that a portable hard disk drive connected to a computer will generally be given a unique drive letter such as “E”, “F” or “G”.

As a result of this, I was able to conclude that the following series of events had taken place prior to the portable hard drive being provided to me for analysis:

- The portable hard disk drive was provided to the senior executive for business use;
- At various points in time after the senior executive commenced using the portable hard disk drive, he copied several thousand files from the corporate file server onto the portable hard disk drive. This was evidenced by the filenames recovered from the previously deleted MFT;
- Prior to the portable hard disk drive being provided to me for analysis, the business related files were deleted from the portable hard disk drive. This was evidenced by the fact that the business related files no longer existed and that a large volume of entries existed in the Windows recycle bin on the laptop computer relating to Microsoft Office and Adobe Acrobat type documents;
- After the business related files were deleted from the portable hard disk drive, a single video file was copied onto the drive many multiples of times to overwrite the data contained in the previously deleted

business related files. This was evidenced by the fact that the data relating the video was still located in the unallocated portion of the portable hard disk drive and that no data from the documents that previously existed on the device still existed;

- The video files contained on the portable hard disk drive were then deleted. This was evidence by the fact that the video files no longer existed on the portable hard disk drive and that a large number of instances of video files were located in the Windows recycle bin on the laptop computer system; and
- The day before the portable hard disk drive was provided to me for analysis it was reformatted. This was evidenced by the creation date of the current MFT on the portable hard disk drive.

As a result of the actions of the senior executive, his employer decided to question him at which point he admitted to copying vast amounts of information from the corporate file server onto the portable hard disk drive provided to him prior to his resignation. He then indicated that he copied those files from the portable hard disk drive onto another device, which was owned privately by him and using a computer privately owned by him. From this point, the senior executive agreed with the sequence of events as outlined above. Litigation resulted and a judgment was found against the senior executive.

The data stolen by the executive consisted of confidential information corporate information such as policy documents, financial information, customer data, product campaign information, business performance documents and other organisational material. The material was all contained within Microsoft Word, Microsoft Excel, Microsoft PowerPoint and Adobe Acrobat formatted files. The files were all named in accordance with corporate file naming guidelines and so it became an easy task to compare the names of the files that once existed on the external USB hard disk drive and the files that currently existed on the corporate files server. This comparison of names was crucial in identifying the nature of the data that once existed on the external USB hard disk drive.

CONCLUSION

As the old saying goes, “a little bit of knowledge is dangerous”. In this instance, a user, who believed himself to be computer savvy, sought to remove all traces of his unlawful behaviour by overwriting data that had been previously been deleted. He was fully aware that deleting a file from a hard disk drive does not actually delete the file, and that formatting a hard disk drive does not delete files. He mistakenly believed, however, that merely overwriting “previously deleted” data would cover his tracks. Whilst his actions did indeed render it impossible to recover any actual files during this analysis, he was not computer savvy enough to remove all traces of a file’s existence, namely entries relating to files as contained in previously used MFT. The existence of the old MFT and the information contained within the Windows recycle bin on the laptop computer, enabled a detailed timeline to be constructed with respect to the sequence of events that led to the relevant information being destroyed on the portable hard disk drive. This also serves to highlight a very relevant point, and that is that it is not always necessary to recover the actual files that are suspected to have been stolen, only information about those files and a plausible and defensible explanation as to how they came to be deleted.

The most important things I learned during the course of this analysis were to never underestimate the importance of ancillary information when trying to prove that data once existed on a device. Just because you don’t have the files themselves, does not mean you cannot build a robust case outlining a chain of events as to what a computer has, or has not done, on a computer system. In addition to this, persistence, never give up, you may be surprised at what you can find if you take the time to sit back and look.

ABOUT THE AUTHOR

Mark Garnett is a Partner of McGrathNicol Forensic, the leader of its Forensic Technology team and specialises in seizing and analysing digital evidence, data recovery, electronic discovery, electronic evidence preservation and conducting other computer related and fraud investigations.

A qualified investigator and forensic technology practitioner with 14 years experience as a Detective in the Queensland Police Service and over nine years specialist forensic experience, Mark has led a broad range of matters in Australia and overseas involving digital evidence recovery and analysis, network intrusion, asset tracing and misappropriation.



Penetration Testing



HP ArcSight Consultancy



SIEM Deployments



CYBER SECURITY EXPERTS

From security assessment services to complex SIEM deployments, we have the experience to deliver an unrivaled service.

Visit our website to discover how we can help you develop advanced threat detection capabilities within your enterprise

DETECTING EVIDENCE OF INTELLECTUAL PROPERTY THEFT

USING FTK® IMAGER (AND FTK® IMAGER LITE)

by Ana M. San Luis & Robert K. Johnson

Global clients of Alvarez & Marsal (A&M), frequently turn to us when investigating suspicious activity, such as activity suggesting fraudulent activity, data privacy or cyber security breaches, or intellectual property (IP) theft, to name a few. When faced with an issue of suspected or alleged IP theft, experts and investigators at A&M rely on experience, sound methods, and reliable tools to investigate such activity. Two such reliable tools include AccessData®'s Forensic Tool Kit (FTK®) Imager and FTK® Imager Lite.

What you will learn:

- Preliminary steps involved in investigating suspected IP theft;
- How to utilize FTK® Imager Lite to perform a live acquisition of an encrypted hard drive;
- How to utilize FTK® Imager in conjunction with other tools to perform preliminary analysis of hashes, Recycle Bin artifacts, and link files within a Windows 7 environment.

What you should know:

- Basic forensic techniques & terminology (for example, physical vs. logical acquisitions, hashes, live acquisition, live memory capture, etc.);
- Basic understanding of the Windows 7 environment;
- Basic understanding of encryption, deleted files, the Windows Recycle Bin, and link files.

In today's world of constantly evolving technology, there arise a number of options for thieves, embittered and disgruntled employees, or naïve colleagues to participate in the theft of intellectual property, whether intentional or otherwise. Intellectual property, in its most basic definition, is any creation of the mind for which exclusive rights exist¹ to superscript. This can range from ideas, inventions, and creative expressions² to discoveries, designs, and even symbols or phrases. Copyrights, patents, and trade secrets are some examples of intellectual property rights that can be infringed, while proprietary products, software or portions of software, and artwork/music/movies are some examples of intellectual property that can be stolen. IP theft can cost victims their jobs, reputations, and even millions of dollars, depending on what is stolen. Digital IP theft has risen with the evolution of digital technology, and a majority of stolen digital IP can be commonly found within the day-to-day operation of businesses, as illustrated in Figure 1 below.

Common Digital Intellectual Property

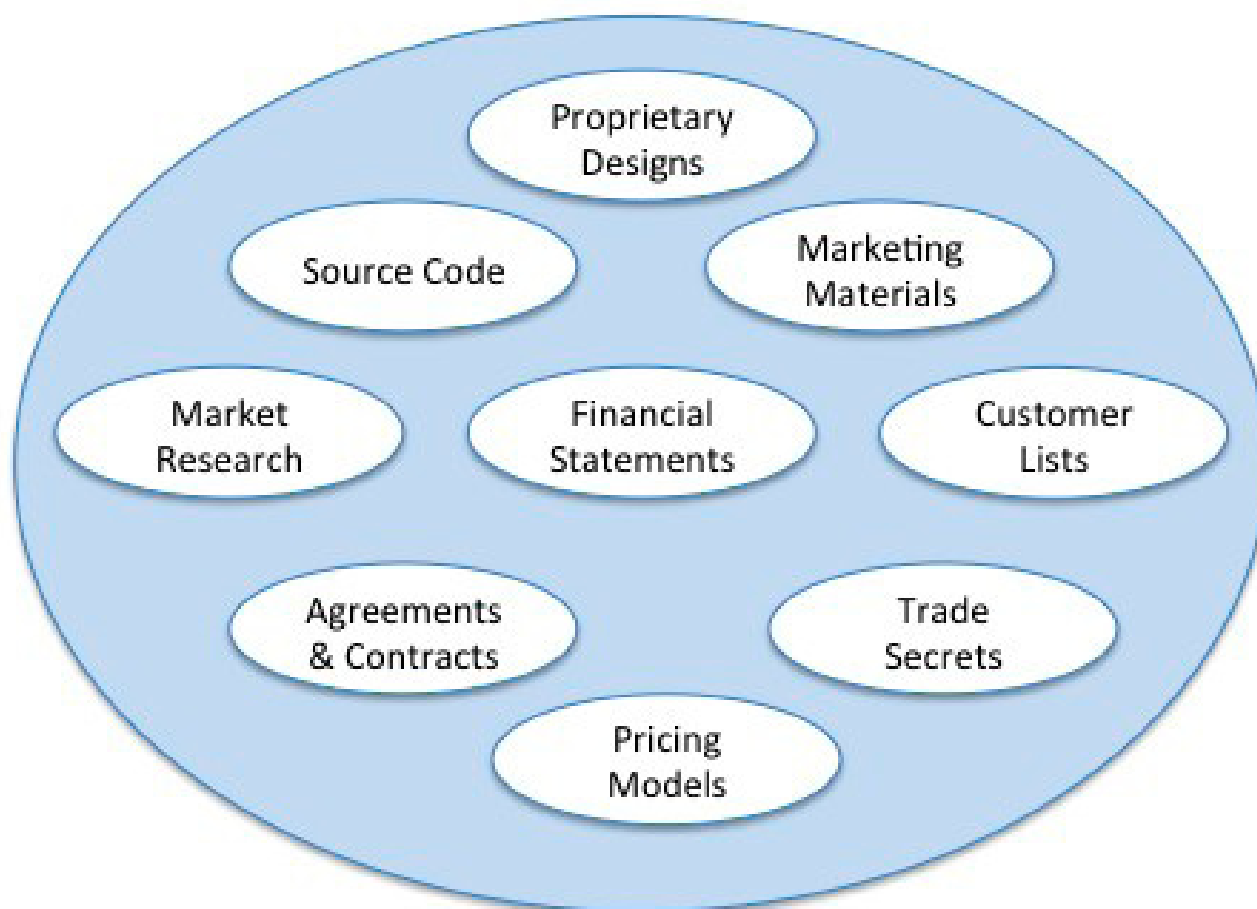


Figure 1. Examples of digital intellectual property commonly found within businesses

Experts and investigators, such as those at A&M, have a number of industry and court accepted tools available at their fingertips to investigate suspicions or allegations of IP theft. Some of these tools allow forensic experts and investigators to examine live running suspect machines or media, while making little to no changes to the suspect machines or media. Two such tools are AccessData®'s FTK® Imager and FTK® Imager Lite, and both are available completely free of cost. These tools allow for some very useful analysis upfront without any software cost.

MOCK SCENARIO

For the purposes of this article, we will be following the investigation related to a mock scenario, which involves a combination of circumstances and situations we have actually faced in the field, surrounding a suspected IP theft matter. While all tools, examination methods, and analyses described below are real and have been utilized in actual investigations, all names related to people and companies in addition to the exact scenario described below are purely fictitious.

SCENARIO

Company A has reached out to us with suspicions of IP theft, alleging that one or more of its employees are stealing proprietary designs for a new line of blue widget machines and are leaking these designs to competitors. Company A believes that the three employees who are mainly suspected of this IP theft are disgruntled employees within their marketing department. Moreover, these three employees should not even have had access to these designs, nor to the software through which the designs are created, edited, and viewed. Company A also believes that the suspected employees may have somehow saved copies of these designs to their machines or to USB thumb drives, prior to leaking IP to the company's

competitors. Unfortunately, Company A does not want to alert the suspected employees of this investigation, and fears that the suspected employees may delete any evidence of their participation in the IP theft if they know they are being investigated. Company A would like us to lead the investigation into the suspected IP theft of their designs in order to confirm or negate their suspicions. Because one of Company A's competitors is expected within the next couple of days to release their new line of blue widget machines, which Company A suspects is significantly based on the designs believed to have been stolen from Company A, Company A would like us to perform a quick forensic analysis. This will allow them to find enough evidence to continue on with the investigation, to issue a preliminary statement to the press, and to seek a possible injunction to prevent its competitor from releasing its new line of machines and making profit off of Company A's own unreleased designs.

PRELIMINARY STEPS

After being hired by Company A, we have several in-depth discussions with trusted individuals within Company A to gain a better understanding of the situation, the suspected employees, and the company's data infrastructure. One of the trusted individuals is John, the head of Company A's Information Technology (IT) department. John informs us that all machines issued by Company A are running Windows 7, are encrypted via TrueCrypt, have two partitions on each machine (a recovery partition and a main user partition), and have a local administrator account, for which only John and two other members of his team have the password. He also mentions that the three suspected employees usually leave their machines powered on, but locked, prior to leaving the office for the evening. John also provides us with copies of the design files, that the company suspects were leaked, and notifies us that only members of their design and engineering team have the IronCAD® software installed on their machines to create, edit, and view the designs.

We decide that the best course of action for the preliminary investigation is to perform a live acquisition and a quick preliminary analysis of the three employees' machines overnight after business hours. This will prevent the employees from suspecting that they are being investigated, will leave the employees' machines running in the same state that the employees left the machines in (and seemingly untouched by us), will allow us to bypass the encryption by imaging a decrypted version of the data, and will allow us to have preliminary findings available to Company A by morning.

We decide to utilize AccessData's free tools, FTK® Imager Lite and FTK® Imager, for the live acquisition and the preliminary analysis, respectively, of the three suspected employees' machines. Because FTK® Imager Lite is a portable tool, it can be copied to any external device and run on a suspect machine, without the tool installing any files on the suspect machine. This will be helpful to us since we want to leave as little trace as possible on the suspects' machines in order to avoid suspicion from the three employees being investigated. We can then use FTK® Imager, in conjunction with some other widely available tools, to perform a quick preliminary analysis while we are on-site overnight.

We prepare each of our target USB drives with copies of FTK® Imager Lite, and ensure that we have a working copy of FTK® Imager on each of our forensic laptops.

Once we are on-site, we decide to accomplish the following tasks before meeting with the trusted individuals of Company A in the morning:

- Live forensic acquisition of the three suspected employees' encrypted machines in an attempt to obtain either a physical or logical image of the machines, including all unallocated space on each partition and all unused space on each hard drive, if possible, on the machines.
- Hash analysis and comparison of the design files John provided us against the files found on the three employees' machines, in an effort to confirm the existence of the suspected leaked design files on the suspected employees' machines.
- Recycle Bin analysis to identify any of the suspect design files that the suspected employees may have deleted and forgotten about in the Recycle Bin.
- Link analysis to identify any link files the system may have created for recently opened files or any shortcuts created by the suspected employees to the design files.

While there are a number of possible ways to proceed with the analysis of a suspected or alleged IP theft matter, we decide to proceed with the aforementioned analysis due to the circumstances surrounding the suspected IP theft and the timing involved with the mock scenario.

Should our preliminary analysis of the above items confirm Company A's suspicions that design files have been misappropriated by the suspected employees, we can perform additional analysis (which will not be covered in this article) at a later date. This analysis will allow us to obtain more comprehensive and in-depth support of Company A's claims, add to our existing findings to see what else the suspect employees may have done or may have intended to do, and include the following:

- USB device analysis to determine possible USB devices which the suspected employees may have connected to their machines, and onto which they may have copied the suspect design files.
- Prefetch analysis to identify any files associated with the startup of the IronCAD® software, in conjunction with the analysis of the Program Files, to determine the existence of the IronCAD® software on each suspect machine. Again, only employees of Company A within their design and engineering team should have this software installed on their machine, and not employees within their marketing department.

MEMORY CAPTURE AND LIVE FORENSIC ACQUISITION VIA FTK® IMAGER LITE

We begin our investigation of the three suspected employees at Company A by capturing the live memory and forensically imaging each of their laptops. Again, in order to prevent the employees from suspecting that they are being investigated, to leave the employees' machines seemingly untouched by anyone, and to bypass the encryption, we decide to perform a live acquisition of each laptop via FTK® Imager Lite.

PHYSICAL VS. LOGICAL ACQUISITION AND THE IMPORTANCE OF ENCRYPTION

Prior to imaging live, we need to decide if we will be able to perform a full physical acquisition or if we will be able to perform only a logical acquisition of each visible partition of each laptop. Typically, with administrator rights, it is possible to acquire either a full physical or logical image through a live acquisition. A logical image will preserve all data within any imaged partitions, but will not preserve any unused disk space not actively assigned to a partition on the drive; whereas a physical image will preserve all data and any unused disk space. The risk with a logical acquisition is that there may be data within the unused disk space, which was previously resident in an earlier created and subsequently deleted partition, which as a result, may be missed with a logical acquisition.

Knowing that the suspect laptops are encrypted with TrueCrypt, we are able to perform a full physical image and capture data on all partitions, as well as the unused disk space on the laptop hard drives. This is not, however, always the case with hard drive encryption, and there are instances when we will need to perform a logical image, as the image may still be encrypted post-acquisition with a physical acquisition. In our case, we have lucked out, as performing a live full physical acquisition of a hard drive encrypted via TrueCrypt while logged in as an administrator will still allow for decryption on-the-fly, and will result in a fully decrypted image.

MEMORY CAPTURE

Using the local administrative (admin) account and password provided by John, we log into the local admin account. We then plug in our target USB device, containing the FTK® Imager Lite software, to the suspect laptop, and note the date/time we plugged in our USB device. This is done so that we do not inadvertently include our own USB device in the list of USB devices plugged into each laptop for a later, more in-depth analysis. We then utilize FTK® Imager Lite to capture the live memory of the laptop for a later analysis of applications currently running to determine whether there is any risk of Trojan viruses or similar. This is accomplished by executing FTK® Imager Lite on the suspect system and selecting "Capture Memory" through the File Menu. From here, we select a destination path and filename, and whether or not we want to capture the pagefile. The image files resulting from the memory capture are approximately 8GB in size, which corresponds to the amount of RAM installed in the laptops. These resulting images can then be analyzed at a later date as necessary without the risk of further altering the resident memory of each laptop.

LIVE FORENSIC ACQUISITION

After successfully capturing the live memory, we then begin the live acquisition of the hard drives installed on each laptop. In FTK® Imager Lite, we select "Create Disk Image" from the file menu. After selecting "Physical Drive" on the next window and choosing the appropriate internal hard drive, we are then presented with a window prompting for image destination and filename. At this window, we select to write to our target external USB drive, in Raw (dd) format, and turn on the options to verify images after creation, pre-calculate progress statistics, and create file listings. Including these options during the forensic imaging will allow us to determine how far along the imaging process we are, give us an ETA of when we

can expect the imaging to complete, and will provide a file listing report of all folders and files found and imaged on each suspect laptop. Additionally, with these settings, the image will be automatically verified to ensure accuracy and completeness of the image at the conclusion of the imaging process.

As shown in Figure 2 below, the physical disk image process captures both the unallocated space within the partition and the unused disk space from the hard drive. This information can be readily queried in the full FTK® application.

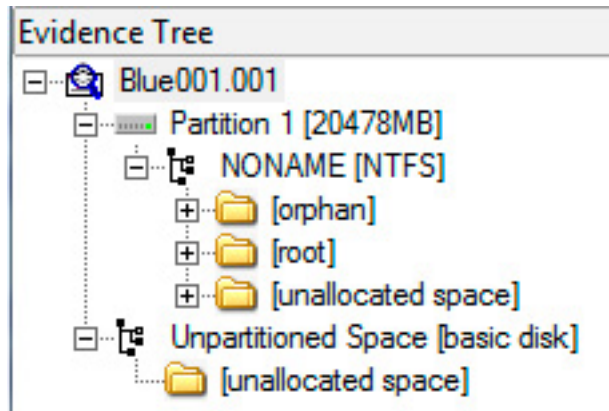


Figure 2. Evidence tree within FTK® Imager showing unallocated space and unpartitioned space, both of which are successfully captured through physical imaging

Following the successful forensic acquisition of the three suspect laptops, we load each image to FTK® Imager by selecting “Add Evidence Item” from the file menu. From there we select “Image File” and point FTK® to the first file for each of the respective images. At this point, we are ready to begin our on-site analysis to see if we can make some quick determinations about user activity, and whether the evidence indicates a strong possibility that Company A’s intellectual property might have been misappropriated.

HASH ANALYSIS VIA FTK® IMAGER & WINMD5

As part of our investigation, we are provided with copies of the files that Company A suspects were stolen. One specific file of interest is an IronCAD® file named BlueWidgetDesign.ic3d. Upon receipt of this file from the trusted individuals at Company A, we create an MD5 hash of this file using a tool called WinMD5Free (<http://www.winmd5.com/>), as shown in Figure 3 below.

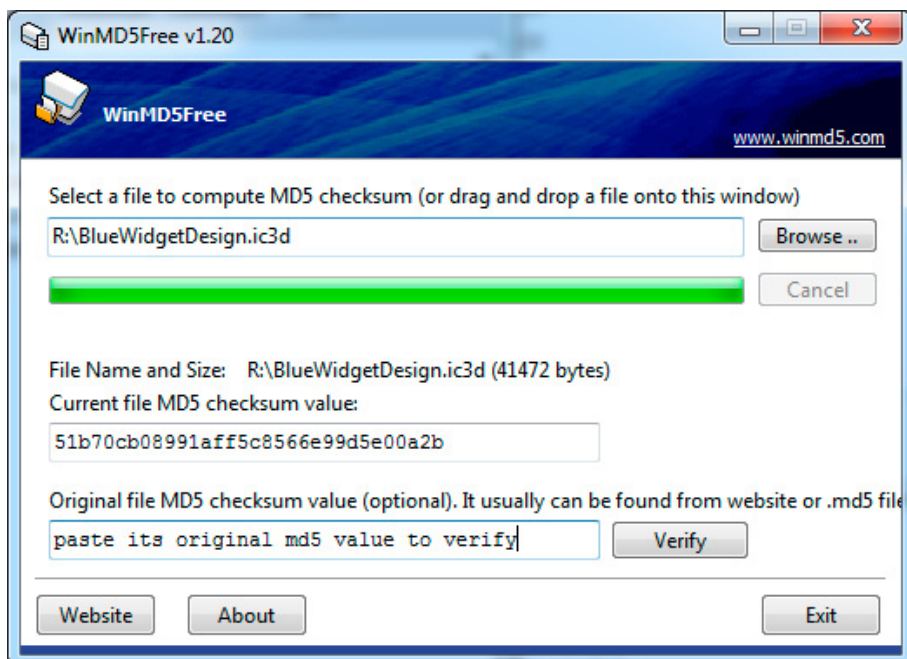


Figure 3. WinMD5 hashing of BlueWidgetDesign.ic3d

Any preferred MD5 or SHA1 hashing tool will work, and if a larger number of files are needed for comparison, we recommend creating an image of the contents of a folder of the sample data through FTK® Imager or Imager Lite, and generating a File Hash List. To compare the hash value of BlueWidgetDesign.ic3d to the contents of the suspect drive, we generate a hash listing of the contents of the hard drives through FTK® Imager. This is achieved by selecting the name of the volume (“NONAME [NTFS]” in Figure 2) in the Evidence Tree in FTK® Imager and then selecting “Export File Hash List...” from the file menu. The resulting CSV file contains the full paths and filenames, and both MD5 and SHA1 hash values for the files captured from the laptops. Comparing the WinMD5 generated hash value (51b70cb08991aff5c8566e99d5e00a2b) from the sample file with the hash listings from FTK® Imager, results in us finding evidence of the BlueWidgetDesign.ic3d file residing in one of the suspect machines’ Recycle Bin as seen in Figure 4 below.

MD5	SHA1	FileNames
		Blue001.001\Partition 1 [20478MB]\NONAME [NTFS]\[root]\\$Recycle.Bin\S-1-5-21-811164276-2822464705-4194092956-1000\SRX7DEM5.ic3d
51b70cb08991aff5c8566e99d5e00a2b	d888d6c8915f43c1e0773b2125baaba80aa06548	

Figure 4. MD5 Hash value from the hash listing generated by FTK® Imager of important BlueWidgetDesign.ic3d file is found on suspect laptop

This discovery confirms for us that, at some point, the suspect had the design file in question on their system, had deleted the file that was then automatically sent to the Recycle Bin, and had forgotten to empty out their Recycle Bin. Further analysis of the Recycle Bin and Link Files will provide more evidence.

RECYCLE BIN ANALYSIS VIA FTK® IMAGER

Having identified a suspect file in the Recycle Bin, we perform further analysis on the file to confirm that in addition to matching on MD5 Hash, the file in the Recycle Bin also matches the filename and file size of the file provided to us by Company A. This will allow us to discover more information about the suspect file, as it existed on the suspected employee’s laptop prior to deletion.

We begin by browsing to the location outlined in the FileNames column from our FTK® File Hash List for the file in question, as previously outlined in Figure 4. Within FTK® Imager, we are able to locate the \$RX7DEM5.ic3d file at the location specified in Figure 4, in addition to a corresponding \$IX7DEM5.ic3d file. Within Windows Vista and Windows 7, the Recycle bin stores two items for each deleted file/folder.

a d v e r t i s e m e n t



The first appears as a file whose names starts with an \$_R followed by a string of characters and the original extension, and the second appears similarly but with an \$_I followed by the same string of characters and the original extension. This \$_R file represents the actual file which was deleted. The \$_I file contains information about the deleted file including the file size, date deleted, and the original full path and filename.

Within the \$IX7DEM5.ic3d file, we focus our attention on bytes 8 through 117. To locate the file size of the deleted file, we look at bytes 8 through 15, represented in hexadecimal as 00 A2 00 00 00 00 00 00. By reversing these bytes (00 00 00 00 00 00 A2 00) and entering them into the calculator within Windows (select the “Programmer” view in Windows 7 or the “Scientific” view in Windows XP), we are able to convert to base 10, or decimal, and determine that the deleted file is 41,472 bytes in size. Alternatively, we can interpret these bytes within the Hex Value Interpreter in FTK® Imager by highlighting bytes 8 through 15 in FTK® Imager, switching to the Hex Value Interpreter in the lower left corner of FTK® Imager, and ensuring that the byte order is set to “Little Endian”. Similarly, to determine the date of deletion, we highlight bytes 16 through 23 within the \$IX7DEM5.ic3d file, switch to the Hex Value Interpreter in the lower left corner of FTK® Imager, and ensure that the byte order is set to “Little Endian” as shown in Figures 5 and 6 below. Through the Hex Value Interpreter, we are able to determine that the \$RX7DEM5.ic3d file was deleted on 3/5/2014 at 9:50PM Local Time, or 3/6/2014 5:50AM UTC.

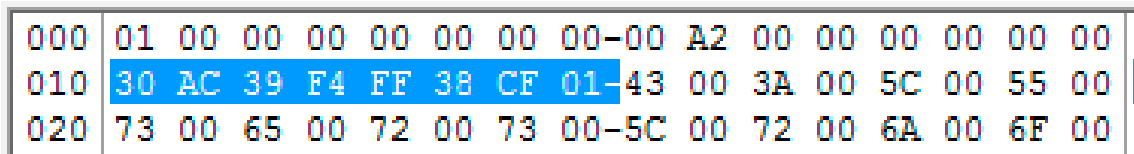


Figure 5. Bytes 16 through 23 highlighted in the \$IX7DEM5.ic3d file through the hex display within FTK® Imager

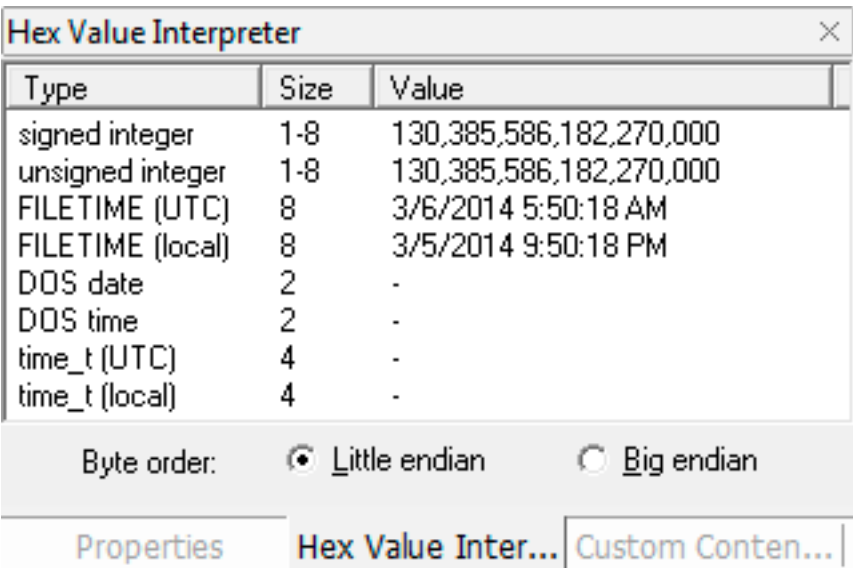


Figure 6. Hex Value Interpreter within FTK® Imager showing date of deletion of suspect file

The remaining bytes of the \$IX7DEM5.ic3d file provide the path and filename of the \$RX7DEM5.ic3d file prior to deletion. In this instance, the path and filename of the \$RX7DEM5.ic3d file prior to deletion is C:\Users\SuspectA\Desktop\BlueWidgetDesign.ic3d. Thus, we are able to confirm that the suspected employee had the BlueWidgetDesign.ic3d file on their desktop, the file size and MD5 hash matched that of the file provided by Company A, and that the file was deleted to the Recycle Bin shortly after the suspected misappropriation.

LINKFILE ANALYSIS VIA FTK® IMAGER AND MITEC'S WINDOWS FILE ANALYZER

At this point, we have shown demonstrably that the suspect file has at one point been on one of the suspects' laptops, and we now move into trying to determine whether the file has been ex-filtrated from Company A for the possible disclosure to a competitor. To perform this analysis, we begin looking at the link files (.LNK extension) found on the suspect's laptop. Examining the link files on a machine will allow us to identify any link files the system may have created for recently opened files, or any shortcuts created by the suspected employees to the suspect design files.

To quickly isolate these link files, we create a Custom Content Source within FTK® Imager. This Custom Content Source, as shown in Figure 7 below, is set to capture all files with a .LNK extension currently existing on the suspect computer. Through this Custom Content Source, we are able to create an image of only the filtered files, and we again select to pre-calculate statistics, verify the image, and create a file listing, as we did with the initial physical image of the suspect laptop, by selecting "Create Image" below the Custom Content Sources.

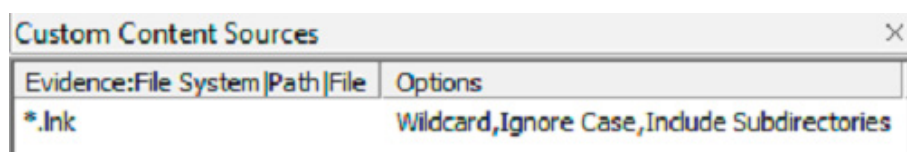


Figure 7. Custom Content Source creation within FTK® Imager

In quickly searching through the file listing of .LNK files generated during the Custom Content Source image, we locate a file named "BlueWidgetDesign.lnk" in a folder on the suspect drive in the following location: "...\\Users\\SuspectA\\AppData\\Roaming\\Microsoft\\Windows\\Recent\\BlueWidgetDesign.lnk". While it is possible to directly interpret the values within the .LNK file from the hexadecimal values, there exist freely available tools which do the job for us. One such tool is MiTeC's Windows File Analyzer (<http://www.mitec.cz/wfa.html>).

In order to utilize MiTeC's Windows File Analyzer (WFA) more readily, we mount the FTK® created AD1 image file containing the .LNK files as filtered through the Custom Content Source by selecting "Image Mounting" within FTK® Imager, as shown in Figure 8 below.

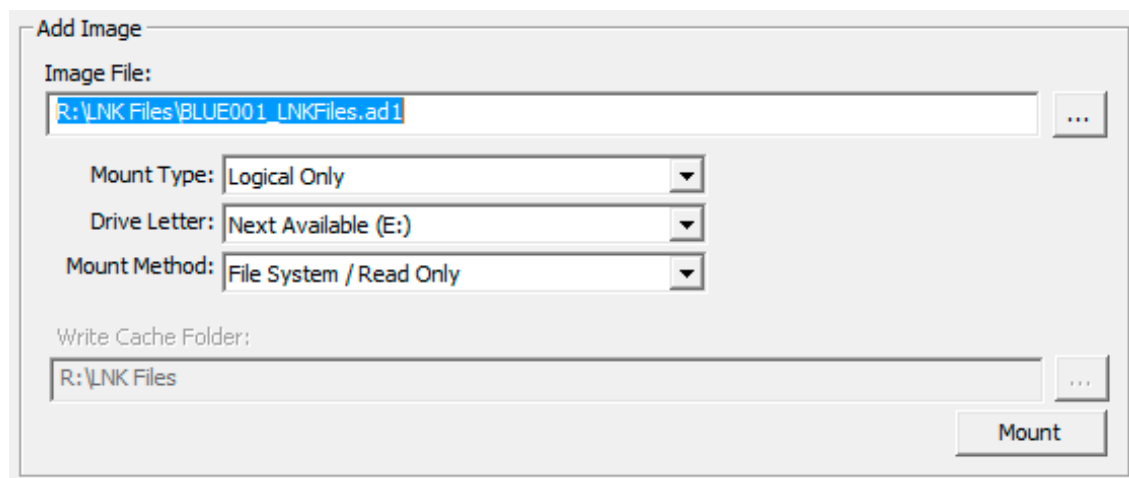


Figure 8. Mounting within FTK® Imager of the AD1 file containing the set of filtered .LNK files

Once mounted, we direct WFA to perform Shortcut Analysis on the folder containing the BlueWidgetDesign.lnk file identified, which is now accessible to applications running on our forensic examiner system. Upon completion of this analysis by WFA, we are able to see that the BlueWidgetDesign.ic3d file was previously opened from a Removable Volume, which typically represents a USB drive, bearing a Volume Serial number of "321A-1213", and a Volume Name of "TRANSFER".

This evidence thus supports the theory that the BlueWidgetDesign.ic3d file at some point resided on removal media and may have been passed to a competitor. Of note, in reviewing the other .LNK files found in the same folder, it appears that two additional files, `SampleCo_2013_Q3_financials.xlsx` and "Company Goals and Directives.docx" were also accessed from this same removable media. These are further items to look into, and it is now time to ask Suspect A for their USB Drive!

CONCLUSION

After completing the imaging of the three suspected employees' machines, while also completing preliminary analysis overnight via FTK® Imager Lite and FTK® Imager, we were able to confirm, in the span of only a few hours, our client's suspicions that one or more of its employees were involved with IP theft of their proprietary designs. We also found evidence of the potential leaking of the designs and other company sensitive information to competitors, of which Company A was not even initially aware. Although we will continue a more in-depth investigation of the suspected employees' machines and external devices, we have resultantly armed Company A with enough ammunition to be able to issue a preliminary statement to the press and to seek a possible injunction to hopefully prevent Company A's competitor from releasing its new line of machines and making profit off Company A's own unreleased designs.

While in many instances with investigations you may need to use a full forensic analysis tool, such as AccessData®'s Forensic Tool Kit, Guidance Software's EnCase®, or X-Ways Forensics, in this scenario, FTK® Imager and Imager Lite, when combined with some freely available tools, allowed us to quickly image encrypted systems, perform preliminary analysis while still on-site, and confirm our client's suspicions of IP theft. In short, when coupled with the right knowledge and complementary toolsets, FTK®Imager and Imager Lite can prove to be very useful.

BIBLIOGRAPHY

1. Wikipedia. (2014, February 26). "Intellectual Property". Retrieved February 28, 2014, from http://en.wikipedia.org/wiki/Intellectual_property.
2. The FBI Federal Bureau of Investigation. (Date of last update unknown). "Intellectual Property Theft". Retrieved March 1, 2014, from http://www.fbi.gov/about-us/investigate/white_collar/ipr/ipr.

ABOUT THE AUTHORS



Ana M. San Luis is a Manager in the New York office of the Forensic Technology Services practice of Alvarez & Marsal, a global professional services firm. A&M professionals draw on their deep skills and experience in forensic technology, business investigations, litigation consulting and expert testimony to provide clients with the solutions they seek to achieve their goals. Ms. San Luis earned her bachelor's degree in Computer Engineering from The Pennsylvania State University and is a member of Women in E-Discovery (WiE). Computer forensics, electronic discovery, and data analytics have been the focus of her career for over 7 years. She has served as a computer forensics examiner and litigation consultant on electronic discovery and investigations involving alleged intellectual property theft, patent disputes, alleged stock options backdating, suspected fraud, alleged sexual harassment, suspected data privacy and cyber security breaches, and class action lawsuits for approximately 150 matters in the US, US territories, Europe, and Canada. She can be reached via e-mail at asanluis@alvarezandmarsal.com.



Robert K. Johnson is a Director in the San Francisco office of the Forensic Technology Services practice of Alvarez & Marsal, a global professional services firm. A&M professionals draw on their deep skills and experience in forensic technology, business investigations, litigation consulting and expert testimony to provide clients with the solutions they seek to achieve their goals. Mr. Johnson earned his bachelor's degree in Computer Science from Vassar College, and is an EnCase Certified Examiner (EnCE), an AccessData Certified Examiner (ACE), and a GIAC Certified Incident Handler (GCIH). During Mr. Johnson's 8 year career, he has focused on computer forensics, electronic discovery, and data analytics, and has provided consulting services for matters based in the US, US Territories, Europe, Canada, and South East Asia. During this time,

Mr. Johnson has served as a computer forensics and electronic discovery litigation consultant in approximately 175 investigations involving alleged intellectual property theft, patent disputes, alleged stock options backdating, suspected fraud, alleged sexual harassment, suspected data privacy and cyber security breaches, and class action litigation. Mr. Johnson can be reached via e-mail at rjohnson@alvarezandmarsal.com.



NIGHT LION[®]
S E C U R I T Y

Information Security Risk Management
24/7 Emergency Incident Response

1.844.HACK.911

www.NightLionSecurity.com

FILE RECOVERY

– PART 01

by **Everson Probst**

One of the core activities of a computer forensics expert is file recovery. Recovery presents possibilities to examine records deleted by users or deleted automatically by the system. This tutorial is about file recovery and covers a few technical issues performed with FTK Imager and Recuva software.

What you will learn:

- Basic concepts about file allocation systems
- How to recover files from a hard disc
- How to use FTK Imager software to virtualize a forensic image
- How to use Recuva software to locate and recover deleted documents.

What you should know:

- specific knowledge is not necessary
- skills with installing programs in Windows environment
- experience with forensic image manipulation

FTK Imager: Free software distributed by AccessData whose main function is to generate forensic copies (E01, AD1, Records and RAM Memory Copies). It also allows browsing of an image in a structured manner without changing it, creating customized images, generating hash lists, etc.

Recuva: Free software distributed by Piriform whose main function is to recover deleted files. It uses the archive system index to recover deleted files and also runs *Data Carver*, but in this aspect, it is not very efficient when compared to Foremost.

RECOVERY BASED ON METADATA

Recovery based on archive system index starts from the principle that the majority of the operational systems keep reference of at least the initial position and size of this file after excluding a file.

These entries remain in the file systems until it is necessary to overwrite them by new-recorded entries.

However, in some file systems such as FAT12, FAT16 e FAT32 systems, just the first cluster address is kept in the index after deleting a file; therefore it is not possible to discover which clusters were specifically used by the file. For this reason, the recovery programs check in the remaining records the size of the deleted file and consider that the file was not fragmented, thus recovering all clusters in the sequel of the initial cluster until reaching the size of the deleted file.

For example: the following figure 1 is a representation of the *Directory Entry*, an important part of the FAT32 file system, where the main information of files in a partition is stored. In our example, we have set 03 parameters only, the ones we need for the recovery process. Thus, we start from the scenario where we have the files presented below in the computer.

	Name	Size	Initial Cluster
Directory Entry 1	File.txt	27kb	100012
Directory Entry 2	Music.mp3	8kb	100019
Directory Entry 3	Pic.jpg	10kb	100021

Figure 1. Example of file entries in a Fat 32 file system

The following figure 2 shows a FAT that serves as a file cluster map. While Directory Entry indicates only the initial position of the file, FAT indicates specifically the cluster sequel used by the file. On the left, we have records about the location of all parts that compound the three files indicated in the table above.

In this table (FAT), the first cluster has one correspondent indicating the next cluster, used by the file in the data area. In case it is the end of file, the value will be 0XF8. Note that the files highlighted in blue and green are fragmented, while the files in red are not fragmented.

FAT		FAT empty		FAT recovered	
ID	Value	ID	Value	ID	Value
100012	100014	100012	0	100012	100013
100013	100019	100013	0	100013	100014
100014	100015	100014	0	100014	100015
100015	100016	100015	0	100015	100016
100016	100017	100016	0	100016	100017
100017	100018	100017	0	100017	100018
100018	100020	100018	0	100018	0XF8
100019	0XF8	100019	0	100019	100020
100020	0XF8	100020	0	100020	0XF8
100021	100022	100021	0	100021	100022
100022	100023	100022	0	100022	100023
100023	100024	100023	0	100023	100024
100024	0XF8	100024	0	100024	0XF8

Figure 2. Example of a FAT 32 structure before and after file deletion and after their recovery

The FAT shown represents the first moment after the user has deleted the three referred files. All clusters are marked with the "0" value, indicating that they are available for using. It is highlighted that only the cluster index in the file allocation table shall have its value changed. Neither the information contained in the *Directory Entry* nor the data itself recorded along the disk will be excluded after deleting the file.

In the FAT represented by the middle table (FAT empty), indicating a supposedly empty partition, we run a file recovery program based on the data present in the *Directory Entry*; the recovery will result in the third moment of the FAT (FAT recovered). Note that the File.txt that was fragmented was incorrectly recovered. It happens because FAT was reset and *Directory Entry* has only the initial cluster reference; the recovery method calculates how many clusters are needed for the file and re-allocates this amount of clusters in a contiguous form. The same happens to the second file. Just the third file, Pic.jpg is correctly recovered because before deletion it was not fragmented.

Why using this method if it is susceptible to flaws? We must consider this recovery method because modern file systems are very efficient in the process of non-fragmentation, which makes this indexing method successful. In addition, we can recover the main properties of a file such as its name, MAC Time, size, etc. despite its content, which is not recovered in full or properly. Finally, in the Computer Forensics environment sometimes it is much more important to know about the file's properties than its content. This premise makes this recovery method very useful.

RECOVERY WITH FTK AND RECUVA

To run the recovery program it is necessary that the drive is mounted. However, according to forensic best practices, procedures shall always be performed on forensic images and never on the original drives. For this reason and aiming to guarantee data integrity, the program FTK Imager shall be used to mount a unit from a Forensic Image. Besides allowing the operation of recovery procedures, when mounting the unit FTK protects the image against recording. Download FTK Imager from this website: <http://www.accessdata.com/support/product-downloads>.

After installing the program, run it. In the window that shall appear, click on the option “File” and “Image Mounting...” as shown in the following figure.

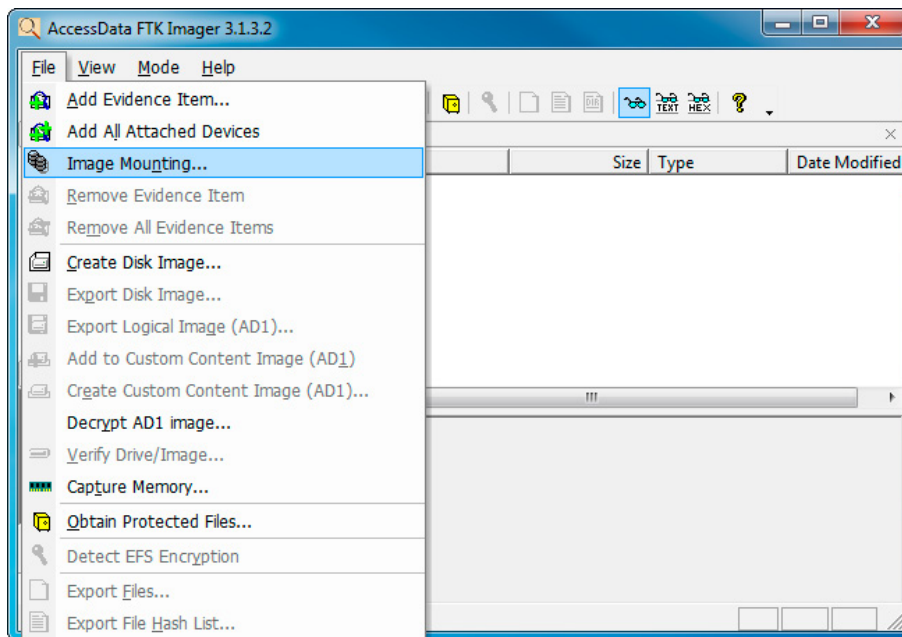


Figure 3. Mounting a forensic image with FTK Imager

In the window “Mount Image To Drive”, choose the forensic image that shall be mounted and select which letter shall be attributed to the unit. All the other setting shall be exactly as in the figure below.

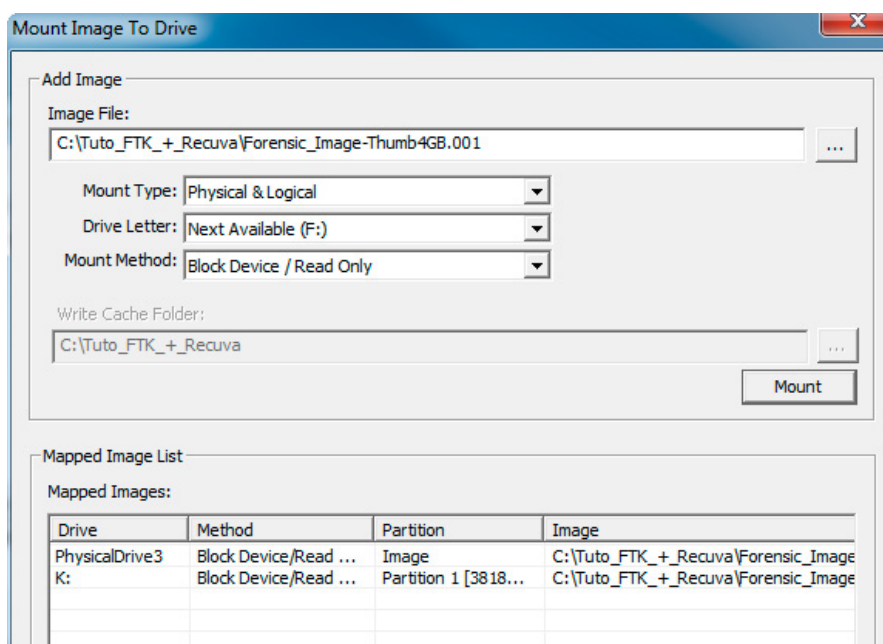


Figure 4. Options for mounting a forensic image

After checking, click on “Mount”. This will make the forensic image to appear as a drive unit protected against recording. Now, download Recuva from this website: <http://www.piriform.com/recuva/download>.

After installing, run the program. In the window that shall appear, click on the option “Next”.

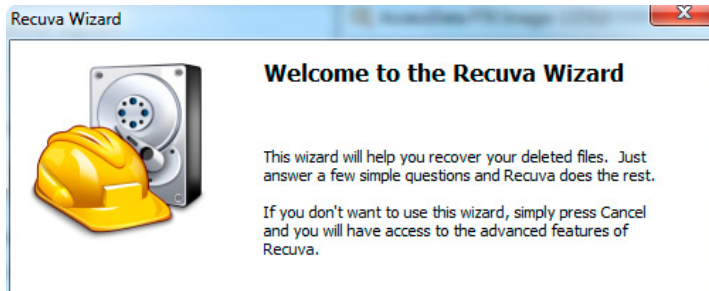


Figure 5. First step of file recovery program

In the next window, you can choose which type of file you want to recover. As the procedure is always very quick and our goal is Forensic, let's choose the option “All Files”.

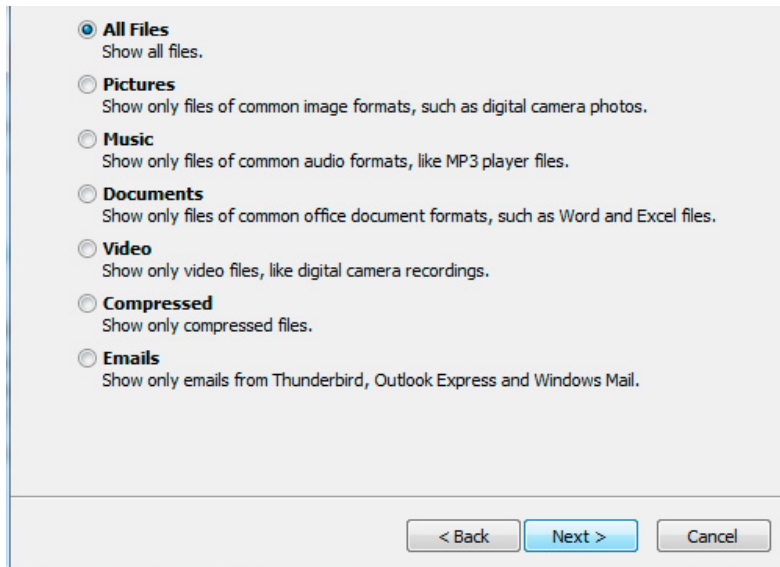


Figure 6. Screen for selecting the type of file you want to recover

The next window is also very important. Here you shall choose the option “In a specific location” and indicate the letter of the mounted unit through FTK Imager. After doing that, click on “Next”.

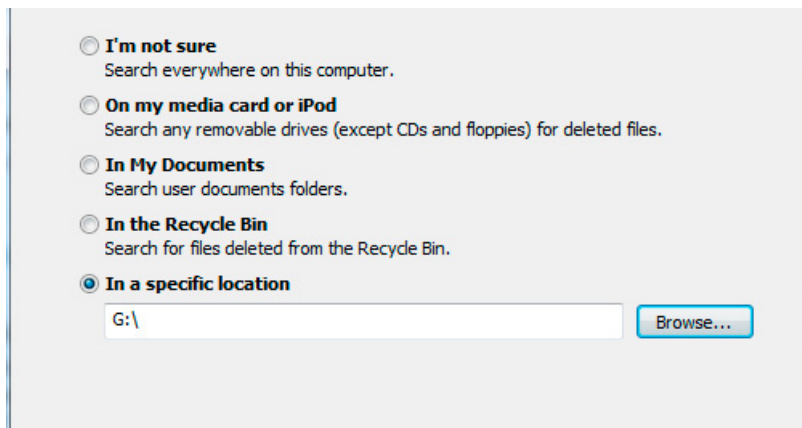


Figure 7. Screen for selecting the disc for examination (newly mounted disk with FTK Imager)

The following screen allows you to choose the option “*Enable Deep Scan*”. This option will extend the form of recovering files. Besides checking archive systems for remaining records, the program will scan all drive searching for headers and foot notes of deleted files which registry have already been overwritten in the archive system. This method is very important and will certainly result in a much bigger volume of files completely recovered, however, besides being very slow, there are other software products much more efficient in this issue, as Foremost that will be detailed in another tutorial. Therefore, leave the option “*Enable Deep Scan*” not selected and click on “Start”.

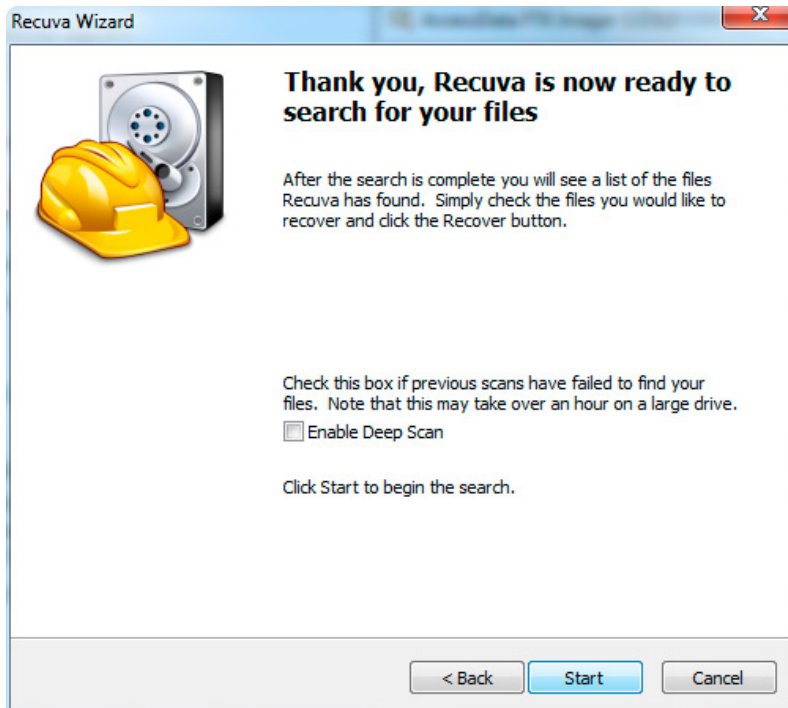


Figure 8. Screen to start the search for deleted files

The procedure is very quick and at the end, a list will be presented with the name of all files that the program located using the method of checking entries of remaining records in the archive system of the computer.

Clearly, as explained in the beginning of this tutorial, this method can result in a series of file names and its metadata, without its complete and whole content actually being recovered. Nevertheless, Recuva knows it and signalizes the file state. The red circle indicates that the file whose data are presented in the list is overwritten by a new file, and therefore cannot be recovered. In the right border of the list, Recuva also shows which new file is overwriting the old file. The yellow circle indicates files which despite incomplete can eventually be opened. The green circle indicates files that have integrity and that can be recovered completely.

Note that until this moment Recuva has only analyzed archive system. To continue, select all files you wish to recover, click with the right button over it and choose the option “*Recover Checked*”. In the window that will appear, you will only have to indicate where you want to store the recovered files and that is it!

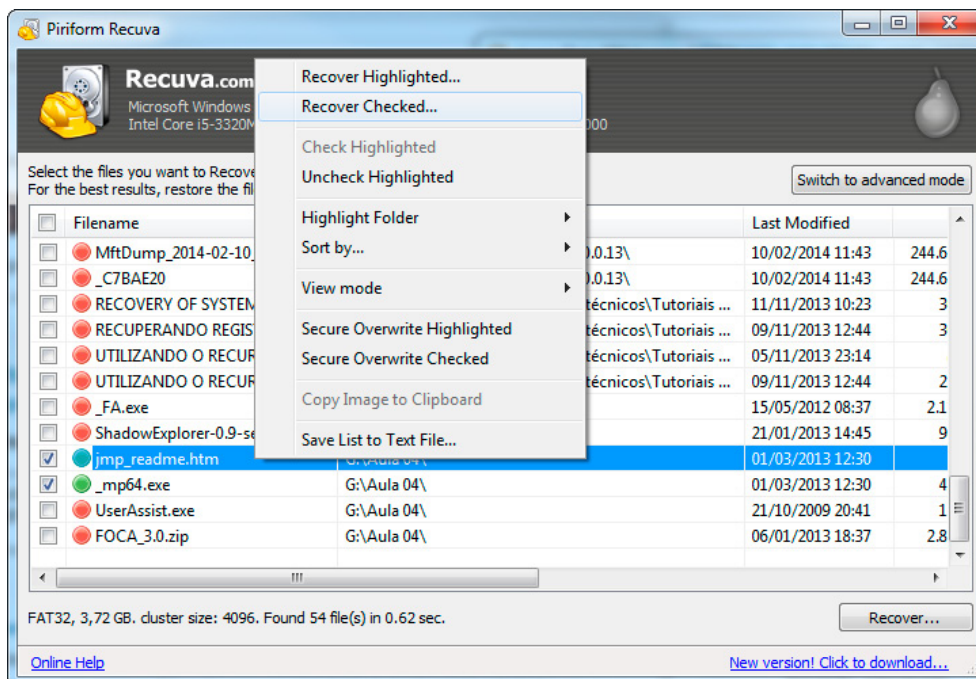


Figure 9. Search result for deleted files

At the end of the recovery, the following message shall be shown. In our example, the two files selected were recovered with success.

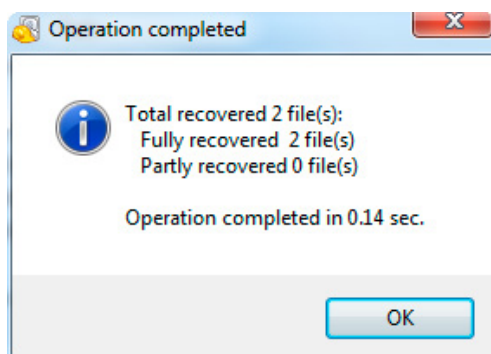


Figure 10. Recovery Status

Note that in the window with the list of files are all the original properties of deleted files. This is very important for Computer Forensics and may help a lot an investigator, even though the content of files has not been completely recovered. Starting from this principle, if you want to analyze all records over what was deleted, just select all files indicated in the Recuva's list, click with the right button on it and choose the option "Save List to text File...". At the end of the procedure, do not forget to return to FTK Imager and click on "Unmount" to unmount the forensic image. That guarantees that the image will remain unchanged and with integrity.

ABOUT THE AUTHOR



Everson Probst is majored in Information Systems and is specialist in computer forensic, disputes and litigation. Guest Professor of the postgraduate course in computer forensics at Mackenzie, he has also taught at Faculdade Paulista de Engenharia – Instituto Brasileiro, Faculdade Paulista de Direito – EPD, Faculdade Impacta de Tecnologia – FIT and Faculdade Getúlio Vargas – FGV, in courses directed to Legal Experts throughout Brazil in partnership with AMCHAM and BSA. Senior consultant in computer forensic and electronic fraud investigations at Deloitte Touche Tohmatsu and member of the Research Committee for Standardization of Forensic Sciences ABNT/CEE-137 (Brazilian Association for Technical Standards) and ACFE (Association of Certified Fraud Examiners).

FILE RECOVERY

– PART 02

by **Everson Probst**

One of the core activities of a computer forensics expert is file recovery. Through recovering, it is possible to examine records deleted by users or deleted automatically by the system.

This tutorial is about file recovery and its technical properties performed with FTK Imager and Foremost software.

What you will learn:

- Basic concepts about how a file is recorded in a hard disk.
- Basic concepts about files fragmentation.
- How to recover files from a hard disc.
- How to use FTK Imager software to create a forensic image.
- How to use Foremost software to locate and recover deleted documents.

What you should know:

- Specific knowledge is not necessary
- skills with installing programs in Windows environment
- experience with forensic image manipulation.

FTK Imager: Free software distributed by AccessData whose main function is to generate forensic copies (E01, AD1, Records and RAM Memory Copies). It also allows browsing of an image in a structured manner without changing it, creating customized images, generating hash lists, etc.

Foremost: Free software that has the function of recovering files based on the Data Carver method. It is capable of recovering files whose record entries are no longer found in the archive system. That makes it a very useful tool to recover older files, despite it is not capable of recovering all original properties of the recovered file.

RECOVERY BASED ON SIGNATURE AND FILE STRUCTURE

Data Carver is a technique used to recover files based only in its signature (header and footnote) or structure. It is used specially when there is no more information about the files in the archive system (FAT, NTFS etc.).

The Foremost program used in this tutorial is based only in the signature check of files. To understand how this method works, it is also necessary to understand how the files are stored in the hard disk.

The main archive systems store data under a minimal unit called Cluster that is a set of sectors (512 bytes in standard configuration). In FAT 32, for example, each Cluster has 4kb in standard configuration. Files bigger than 4kb, i.e., which do not fit into a single Cluster will occupy as many Clusters as needed. In the following example the Tutorial.txt file has 652kb, i.e., it will occupy 163 Clusters of 4kb each ($652\text{kb} / 4\text{kb} = 163$ Clusters).

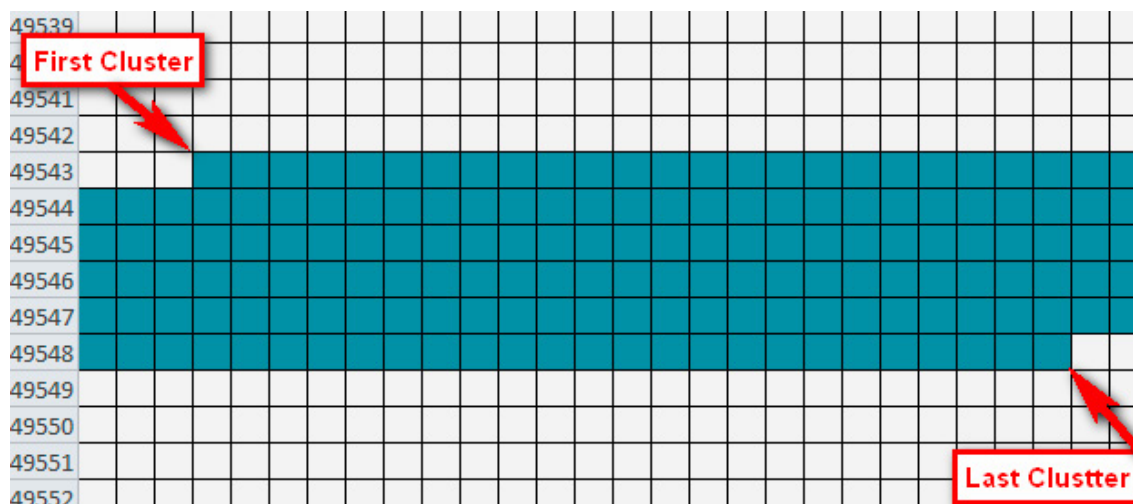


Figure 1. Example of a non-fragmented file (each block represents a cluster on the disc)

Also in the example above, all Clusters occupied by the Tutorial.txt file are sequential, i.e., the file is not fragmented in the disk. Clearly some files end up being fragmented in the moment of its recording, however archive systems avoid file fragmentation and it is with this premise that the main methods of Data Carver work with.

Knowing that the majority of files have specific headers that identify them, the Data Carver algorithms go through the disk searching for these headers. Once a header is found, the beginning of a possible file is considered.

The next task is to know where this possible file ends. For this, two methods are commonly used; the first is for files that do not have footnote. In this case, the algorithm analyzes the initial structure of the possible file and keeps reading the disk from the header until the moment this structure changes. In the place of the structure change, the algorithm considers the possible location of the file end. The second method is based on the location of the file footnote but this only works for files that have unique signatures in its footnotes.

Note that both methods succeeded in locating files recorded sequentially in the disk. For fragmented files there are other methods much more computational expensive and complex.

DETERMINING THE END OF FILE BY STRUCTURAL ANALYSIS

There are many forms of structural analysis. In some cases this analysis is simply based on the symbolic representation used by the file (ANSI, UNICODE etc.) but this is very little efficient. Another form is the entropy analysis. Entropy is the disorder level of a system. In computing, the comparison of entropy levels may be used to determine which parts belong to a same file. In this case, the algorithm can calculate the entropy level of the beginning of file, and from the located header check the entropy level of the subsequent clusters. In the moment a Cluster with a very different entropy level is reached, the end of file is also supposed to be found. Note that even if a recovered file cannot be opened for missing one or other small part, it shall be considered recovered for there are other tools and methodologies for fixing corrupted files.

See the next example. If you open photography in a hexadecimal text editor, you will see that this kind of file is compound by millions of different characters. These characters are responsible for defining image size and resolution, its colors, shapes and all else that can be seen in the computer monitor. This huge possibility of characters guarantees that photography Clusters will have a high degree of entropy.

2014-02-13_155158.png																
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52
00000010	00	00	02	8C	00	00	01	C7	08	02	00	00	00	BA	26	9E
00000020	B3	00	00	00	06	62	4B	47	44	00	00	00	00	00	00	F9
00000030	43	BB	7F	00	00	00	09	70	48	59	73	00	00	0E	C4	00
00000040	00	0E	C4	01	95	2B	0E	1B	00	00	20	00	49	44	41	54
00000050	78	9C	EC	BD	07	74	1C	D7	95	AE	0B	AD	3B	33	F7	8D
00000060	E7	8D	AF	C7	B2	2D	51	C1	62	26	C1	4C	31	93	62	90
00000070	44	52	CC	19	CC	09	89	48	44	06	08	06	E4	9C	73	24
00000080	42	03	E8	6E	E4	9C	73	CE	39	03	04	98	29	C9	B2	28
00000090	C9	96	64	CF	9A	F7	EE	DC	E5	B7	F7	39	55	D5	55	1D
000000A0	80	06	83	AC	99	D7	7B	7D	AB	57	A1	D1	5D	D5	84	0D
000000B0	7D	F8	F7	39	75	8E	D6	DF	84	F5	CF	6F	2F	20	2C	FC
000000C0	C9	A1	D7	5D	F0	7F	CD	C8	5B	3F	6F	7E	C7	31	5F	91
000000D0	FF	F9	93	F1	DB	59	31	EF	F5	F2	9B	19	F8	A7	17	64
000000E0	EE	CB	F2	E6	CB	F3	C1	8B	F3	EB	17	E3	F7	2F	C0	3F
000000F0	2A	E5	DF	D4	E1	FD	9F	88	5F	01	EF	FD	0C	78	F7	05
00000100	F9	5F	EA	F3	CE	6B	E7	97	C0	9C	9F	9E	7F	50	80	3C
00000110	FF	F6	8C	FC	83	3A	FC	EB	AC	78	EB	BF	0E	BF	93	43

Figure 2. Hexadecimal view of a PNG file

If you open a text file with a “txt” extension in a hexadecimal editor, you will see that it is compound by a very limited set of characters (basically letters and numbers), assuring that the entropy degree is very low. See an example of a hexadecimal visualization of a text file in the figure below.

ASL-LICENSE-2.0.txt																
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000000	0D	0A	20	20	20	20	20	20	20	20	20	20	20	20	20	20
00000010	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
00000020	20	20	20	41	70	61	63	68	65	20	4C	69	63	65	6E	73
00000030	65	0D	0A	20	20	20	20	20	20	20	20	20	20	20	20	20
00000040	20	20	20	20	20	20	20	20	20	20	20	20	20	20	56	65
00000050	72	73	69	6F	6E	20	32	2E	30	2C	20	4A	61	6E	75	61
00000060	72	79	20	32	30	30	34	0D	0A	20	20	20	20	20	20	20
00000070	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
00000080	20	68	74	74	70	3A	2F	2F	77	77	77	2E	61	70	61	63
00000090	68	65	2E	6F	72	67	2F	6C	69	63	65	6E	73	65	73	2F
000000A0	0D	0A	0D	0A	20	20	20	54	45	52	4D	53	20	41	4E	44
000000B0	20	43	4F	4E	44	49	54	49	4F	4E	53	20	46	4F	52	20
000000C0	55	53	45	2C	20	52	45	50	52	4F	44	55	43	54	49	4F
000000D0	4E	2C	20	41	4E	44	20	44	49	53	54	52	49	42	55	54
000000E0	49	4F	4E	0D	0A	0D	0A	20	20	20	31	2E	20	44	65	66
000000F0	69	6E	69	74	69	6F	6E	73	2E	0D	0A	0D	0A	20	20	20
00000100	20	20	20	22	4C	69	63	65	6E	73	65	22	20	73	68	61
00000110	6C	6C	20	6D	65	61	6E	20	74	68	65	20	74	65	72	6D

Figure 3. Hexadecimal view of a TXT file

If these two files are sequentially recorded in the disk as shown in the figure below, although computationally complex it is possible to presume which was the last Cluster occupied by each of them and, therefore, to recover both files.

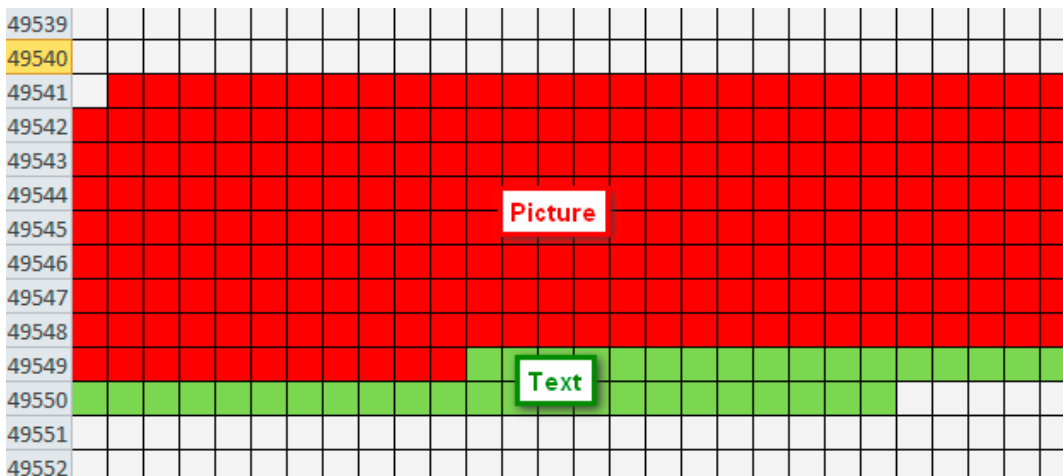


Figure 4. Example of two non-fragmented files recorded in the disc

DETERMINING THE END OF FILE BY FOOTNOTE LOCATION

Luckily, many files have headers and footnotes. This makes the recovery processes easier. In this case, just locate the specific header of a document type and, from its address, locate the next valid footnote.

In the website http://www.garykessler.net/library/file_sigs.html you will find dozens of headers and footnotes of the main types of existent files. Based on the information obtained in it, it is possible to easily recover a file as shown in the example in the figures below.

Imagem_Forenses.dd																
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0018B080	43	42	32	43	39	45	45	37	46	39	31	38	31	34	42	39
0018B090	42	42	42	37	39	37	35	35	31	44	30	36	42	35	30	3E
0018B0A0	3C	44	43	42	32	43	39	45	45	37	46	39	31	38	31	34
0018B0B0	42	39	42	42	42	37	39	37	35	35	31	44	30	36	42	35
0018B0C0	30	3E	5D	20	2F	50	72	65	76	20	39	37	30	37	32	2F
0018B0D0	58	52	65	66	53	74	6D					30	34	3E	3E	0D
0018B0E0	0A	73	74	61	72	74	78					0A	39	39	33	33
0018B0F0	31	0D	0A	25	25	45	4F	46	50	4B	03	04	0A	00	00	00
0018B100	00	00	09	75	50	44	F2	80	05	45	5C	06	0B	00	00	06
0018B110	00	00	08	00	00	00	46	6F	74	6F	2E	70	6E	67	89	50
0018B120	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	00	00
0018B130	00	1D	00	00	00	15	08	02	00	00	00	35	A2	72	35	00
0018B140	00	00	01	73	52	47	42	00	AE	CE	1C	E9	00	00	00	04
0018B150	67	41	4D	41	00	00	B1	8F	0B	FC	61	05	00	00	00	09
0018B160	70	48	59	73	00	00	0E	C3	00	00	0E	C3	01	C7	6F	A8
0018B170	64	00	00	05	F1	49	44	41	54	48	4B	8D	95	79	4C	93
0018B180	67	18	C0	FB	A7	89	72	F6	A0	77	BF	F6	FB	DA	52	DA
0018B190	D2	BB	14	4A	5B	CA	15	6E	6C	39	05	5A	0B	E3	12	4A
0018B1A0	81	82	45	61	D4	8A	28	53	B4	03	04	1D	6E	22	8A	C8
0018B1B0	34	43	44	45	51	74	CE	78	EC	52	07	1E	80	DA	A1	E2
0018B1C0	B5	C4	E8	32	E3	64	46	D4	BD	AC	86	91	EC	7C	D2	3C
0018B1D0	C9	97	A6	BF	F7	D7	E7	79	DE	E7	43	BD	FB	AF	B8	32
0018B1E0	F9	28	AB	C0	42	67	B2	F9	22	29	04	B3	24	32	05	04
0018B1F0	B3	99	1C	2E	95	C1	A0	C1	08	19	82	FC	F0	44	36	97
0018B200	47	24	53	FD	70	38	2F	2F	AF	25	4B	96	2C	5A	B4	08

Figure 5. Hexadecimal view of a PNG file header

The first figure shows the exact position of the beginning of the header of a file with a “PNG” extension. From it, we search the hexadecimal character set that represents the footnote of a “PNG” file, as shown in the next figure.

Imagem_Forenses.dd																
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0018B690	7A	BB	31	A1	BF	3A	77	78	57	FB	BD	69	D7	BD	3B	E3
0018B6A0	AE	5B	D7	9D	2D	AD	A5	96	8A	C9	5B	37	17	AE	3D	B0
0018B6B0	0C	BF	38	76	46	28	0B	56	68	34	CD	4E	E7	A3	C7	73
0018B6C0	72	A8	A3	1B	2C	DD	56	43	6D	B2	AA	3E	2D	C6	96	AC
0018B6D0	2D	D3	4A	37	1B	13	76	AE	48	EB	33	67	B4	16	2E	2D
0018B6E0	CA	4A	D3	86	47	18	72	4C	AE	25	15	F9	85	E6	3D	BD
0018B6F0	7D	7F	BB	48	5F	BC	7A	BB	6F	60	68	EC	EA	D8	FC	B7
0018B700	A8	5D	56	C3	EE	6A	D3	D6	82	94	2D	A6	C4	96	BC	E4
0018B710	B5	29	9A	8E	42	DD	50	53	E5	ED	91	FD	85	C6	2C	91
0018B720	34	04	70	A5	52	59	68	A8	B2	C5	39	37	33	FF	33	50
0018B730	4D	39	F1	ED	25	19	DD	55	C6	7E	7B	F1	A1	75	65	17
0018B740	7B	3B	1E	8F	5F	9E	79	FE	0C	FC	7E	78	E4	04	13	61
0018B750	6A	B5	11	2A	35	80	47	F4	F4	EC	FA	2B	74	EE	AD	B0
0018B760	20	DC	8F	20	FF	0E	12	D8	28	CD	18	A7	6A	61	00	00
0018B770	00	00	49	45	4E	44	AE	42	60	82	50	4B	01	02	14	00
0018B780	0A	00	00	00	00	00	72	51	4A	44	FF	04	89	3A	A9	2B
0018B790	17	00	A9	2B	17	00	50	00	24	00	00	00	00	00	00	00
0018B7A0	20	00	00	00	00	00	00	00	4C	6F	63	EB	43	6F	64	65
0018B7B0	2D	43	6F	6D	70	75	74	65	72	2D	46	6A	6A	6A	6A	6A
0018B7C0	69	63	2D	45	78	61	6D	69	6E	65	72	2D	47	75	69	64
0018B7D0	6B	2D	52	65	66	65	72	65	6E	63	65	2D	47	75	69	64

Figure 6. Hexadecimal view of a PNG file footer

Based on this information it is easily possible to export the data interval between the files header and footnote, saving it as a newly recovered file. This can be done in any hexadecimal editor. However, why to do it manually if we can use FTK Imager and Foremost?

RECOVERING WITH FTK AND FOREMOST

Foremost program was originally developed for Linux Operating Systems. However, this tutorial will be run in Windows. To do this, we will use an emulator called Cygwin. It shall be downloaded in the following website <http://cygwin.com/> and installed with its standard configurations. If there is any difficulty while installing, refer to the technical material developed by developers.

Foremost shall be downloaded in the following website <http://duncanwinfrey.com/wp-content/uploads/2012/05/Foremost-1.5.3-Cygwin-Build.zip>. After downloading the program, put the “foremost.exe” and “foremost.conf” files inside the folder “C:\Cygwin\bin\”. Lastly, download the FTK Imager program from the website: <http://www.accessdata.com/support/product-downloads>. To perform the procedure, we shall generate a forensic image in a RAW format of the device where were the files we want to recover. This procedure shall be performed with the FTK Imager. Run FTK Imager. In the window that will appear, click on the option “Create Disk Image” as shown in the next figure.

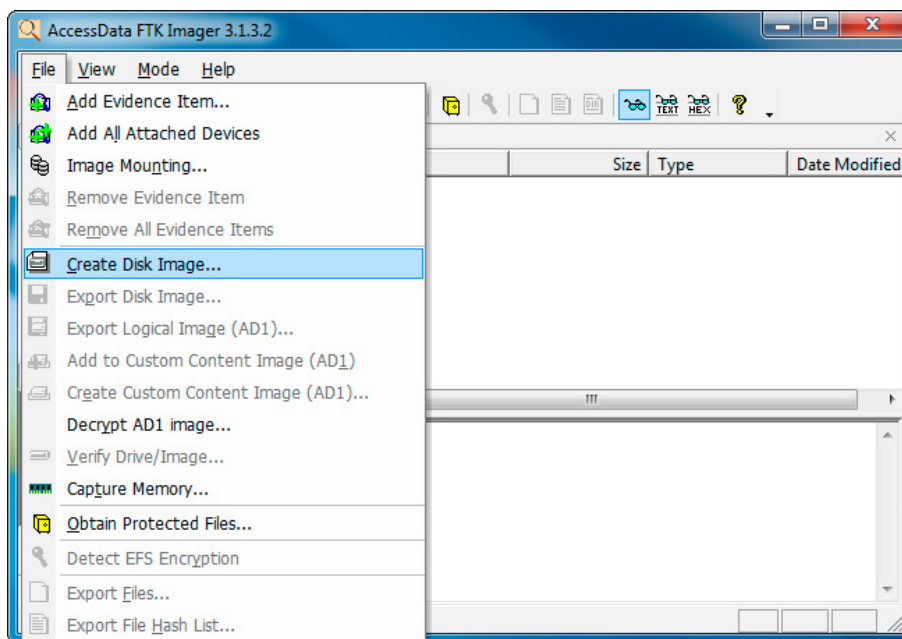


Figure 7. Option for creating a forensic image with FTK Imager

In the following window, choose the option “Physical Drive” and click on “Next”.

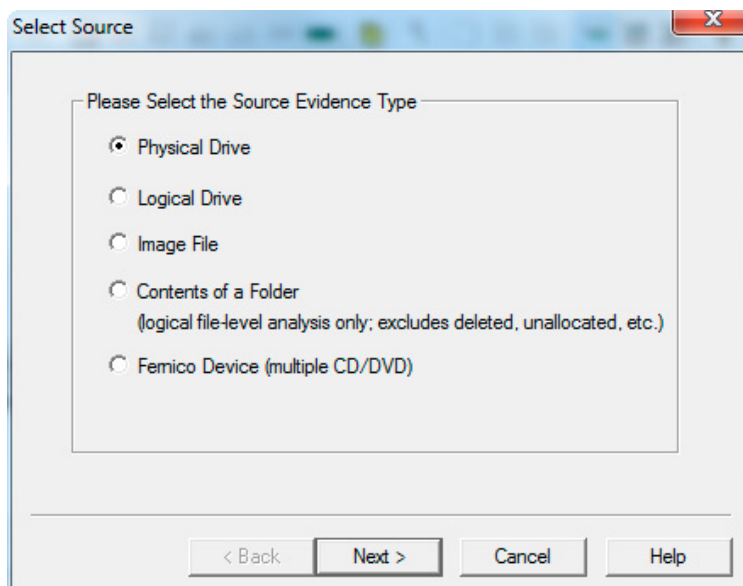


Figure 8. Screen for selecting the copy type

In “Source Drive Selection”, choose the drive from which you wish to recover the deleted files.

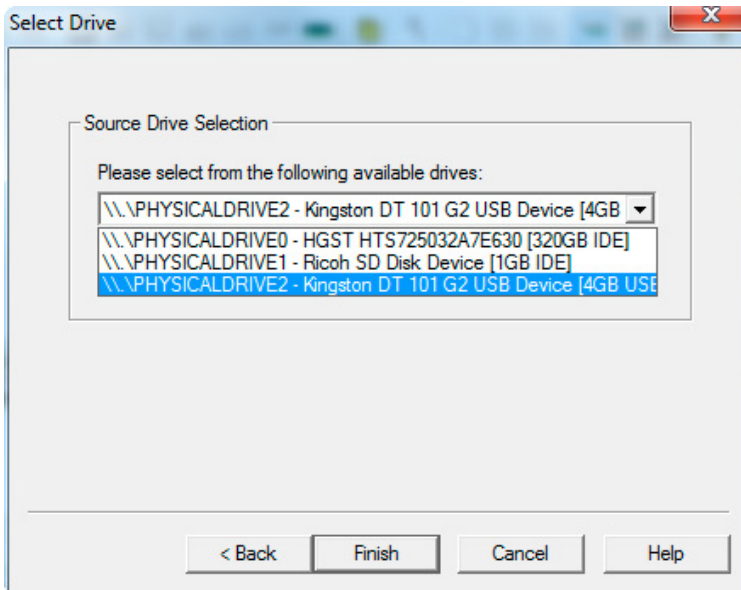


Figure 9. Screen for selecting the disc to be copied

After selecting the source drive, we shall choose the forensic image format. To do this, click in the “Add...” button.

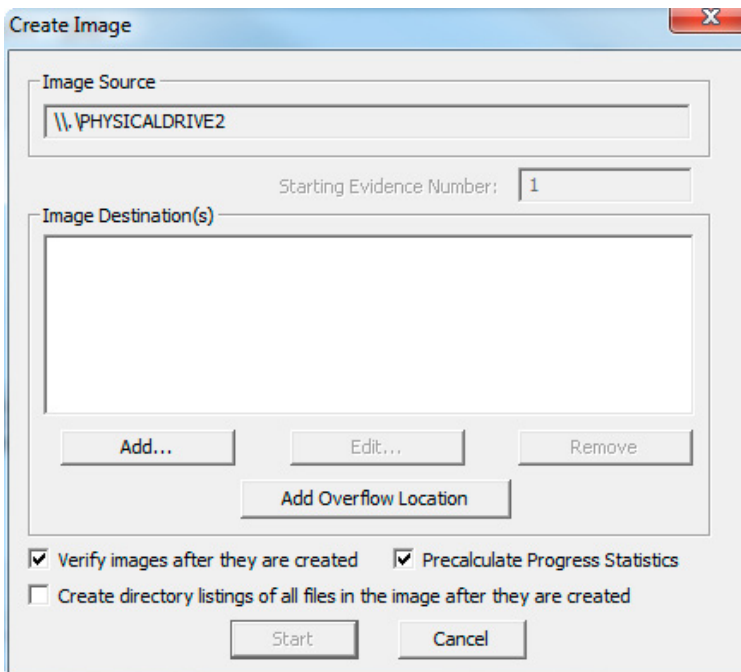


Figure 10. Starting the forensic image setup

In the next window, choose the first option (“Raw”). Do not forget that to recover files from a forensic image it shall be in a RAW format.

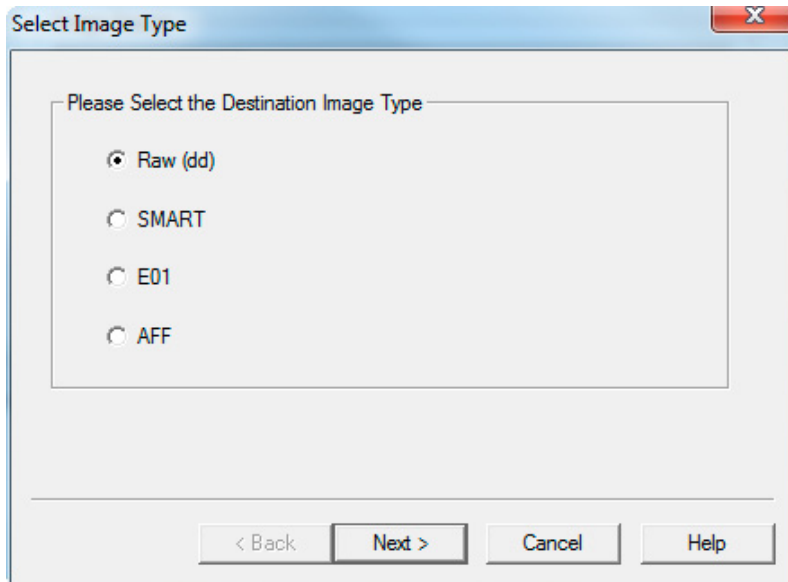


Figure 11. Choosing the output format

Next, you shall type the information about the forensic image. In our example, this is not necessary, so just click on “Next”. The next window shown below is very important; in it you shall choose the folder which will keep the new forensic image, the image name and the fragment size that shall be configured as “0” (zero), as shown in the following image. To conclude, click on “Finish”.

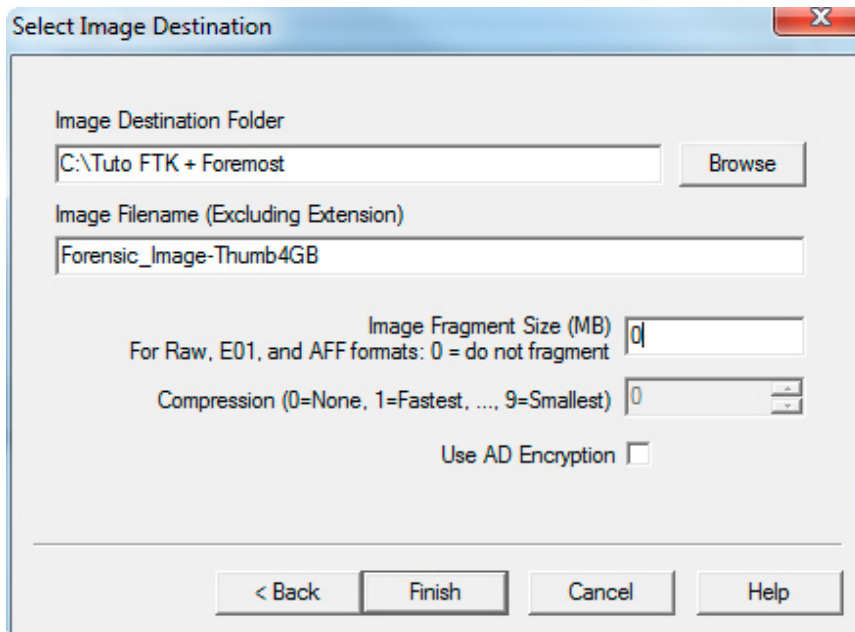


Figure 12. Choosing the destination folder of the image and image name

After finishing all configurations for generation of the forensic image, mark the option “Verify images after they are created” and “Precalculate Progress Statistics” and click on “Start”.

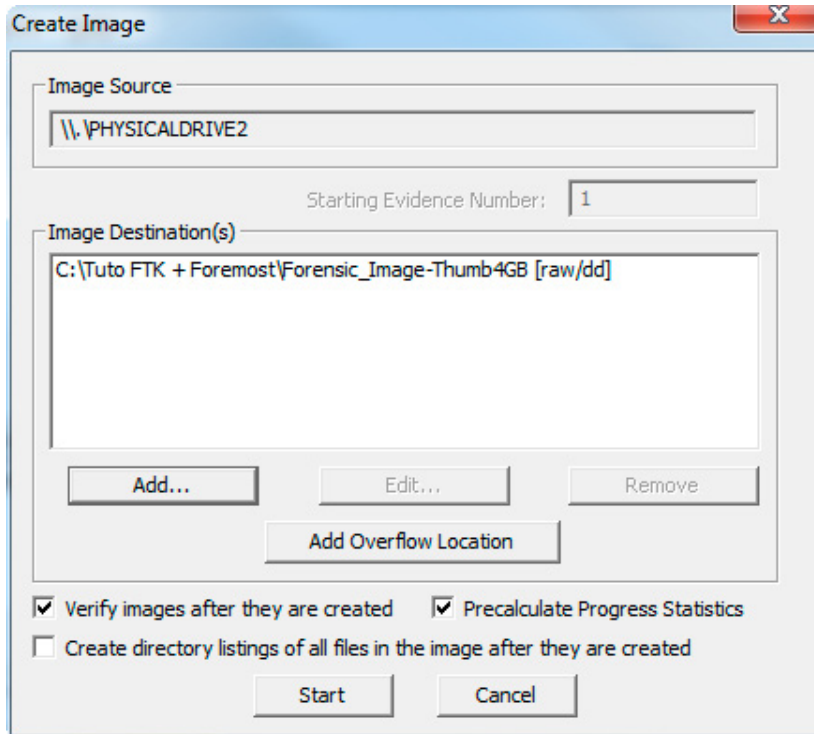


Figure 13. Starting the forensic copy

After the forensic image is generated, a report will be shown indicating if the procedure was performed successfully. Now, we will begin the file recovery procedure. To do this, click on “Cygwin.bat” that can be found at “C:\Cygwin\” after the correct installation of the emulator. The program will begin in a terminal where we shall type the following command:

```
"foremost.exe -c foremost.conf -i /cygdrive/c/<Source/Forensic_Image.dd> -o /Cygwin/c/<Destination Directory>".
```

The parameter `-i` means “Input”, thus after the parameter it shall be indicated the RAW forensic image address just created. The parameter `-o` means “Output” and in the address where the recovered files will be stored shall be indicated. Note that to access the local drive by the emulator it is necessary to use `/cygdrive/` and next the unit letter.

Another important observation is that the output folder indicated after the parameter `-o` shall be just created by the user. No file or subfolder can be created in it before the procedure; otherwise, Foremost will reject the folder.

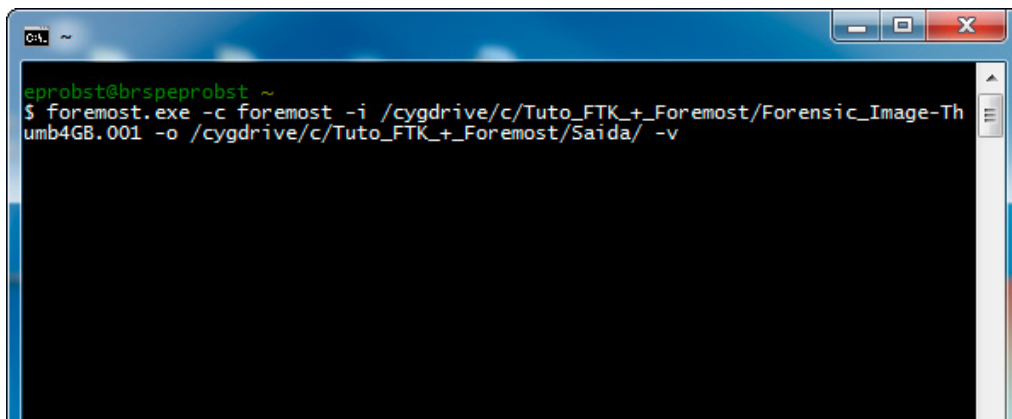
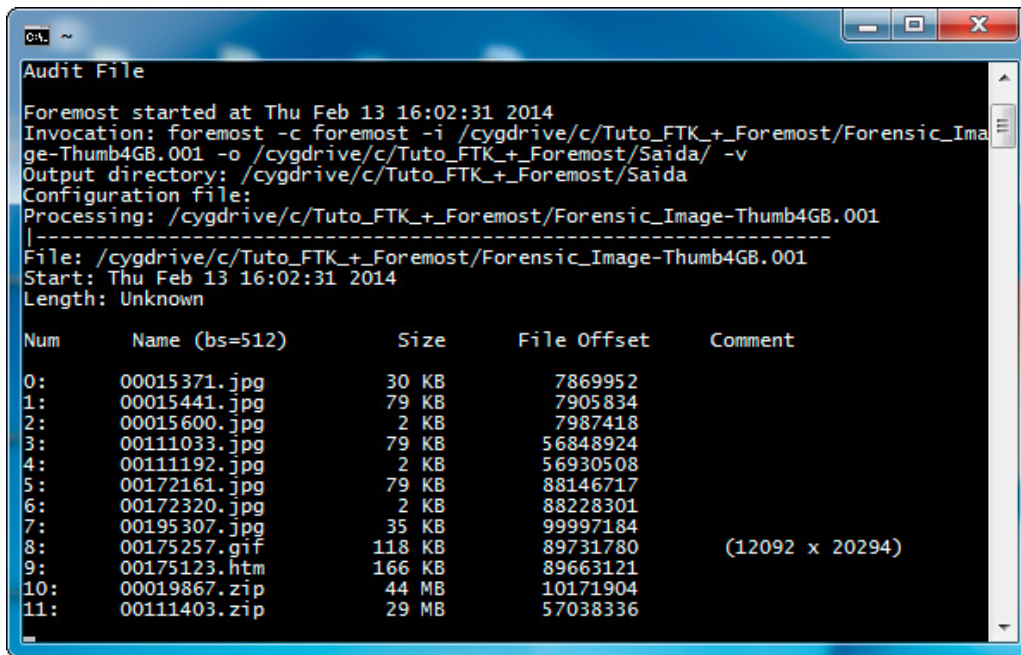


Figure 14. Foremost command to file recovery

If the command is complete, type “Enter”. The result is shown in the next figure.

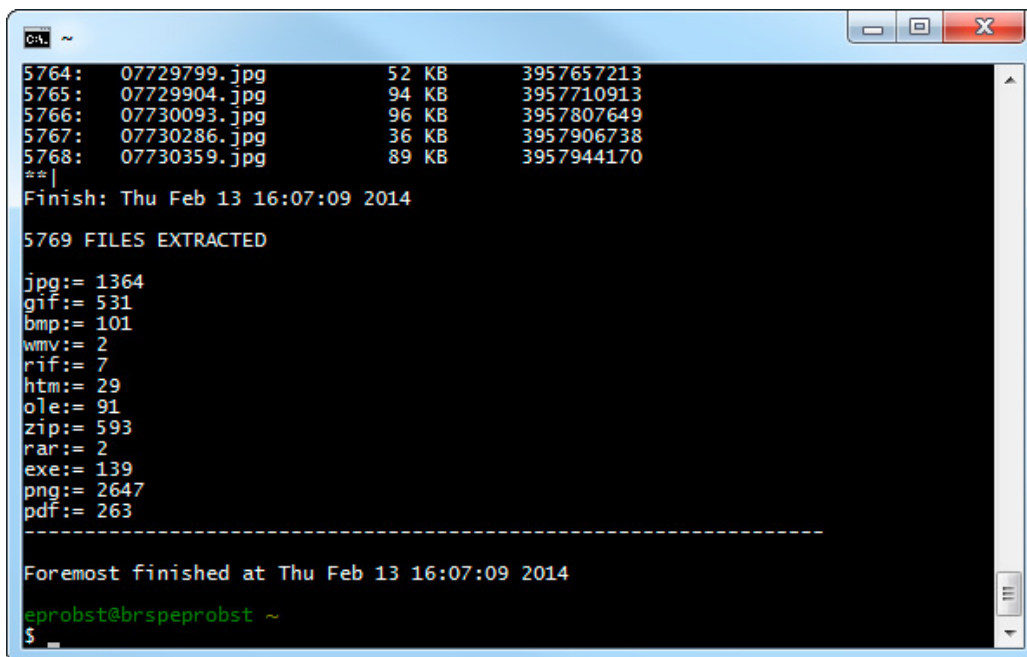


```
Audit File
Foremost started at Thu Feb 13 16:02:31 2014
Invocation: foremost -c foremost -i /cygdrive/c/Tuto_FTK+_Foremost/Forensic_Image-Thumb4GB.001 -o /cygdrive/c/Tuto_FTK+_Foremost/Saida/ -v
Output directory: /cygdrive/c/Tuto_FTK+_Foremost/Saida
Configuration file:
Processing: /cygdrive/c/Tuto_FTK+_Foremost/Forensic_Image-Thumb4GB.001
-----
File: /cygdrive/c/Tuto_FTK+_Foremost/Forensic_Image-Thumb4GB.001
Start: Thu Feb 13 16:02:31 2014
Length: Unknown

Num      Name (bs=512)      Size      File Offset      Comment
0:       00015371.jpg       30 KB      7869952
1:       00015441.jpg       79 KB      7905834
2:       00015600.jpg        2 KB      7987418
3:       00111033.jpg       79 KB      56848924
4:       00111192.jpg        2 KB      56930508
5:       00172161.jpg       79 KB      88146717
6:       00172320.jpg        2 KB      88228301
7:       00195307.jpg       35 KB      99997184
8:       00175257.gif      118 KB      89731780      (12092 x 20294)
9:       00175123.htm      166 KB      89663121
10:      00019867.zip        44 MB      10171904
11:      00111403.zip        29 MB      57038336
```

Figure 15. Real time view of the recovered files information

When Foremost concludes the image checking and locates as many files as it is possible to recover, a summary will be presented indicating the amount of files recovered by type, as shown in the next figure.



```
5764: 07729799.jpg       52 KB      3957657213
5765: 07729904.jpg       94 KB      3957710913
5766: 07730093.jpg       96 KB      3957807649
5767: 07730286.jpg       36 KB      3957906738
5768: 07730359.jpg       89 KB      3957944170
**|
Finish: Thu Feb 13 16:07:09 2014

5769 FILES EXTRACTED

jpg:= 1364
gif:= 531
bmp:= 101
wmv:= 2
rif:= 7
htm:= 29
ole:= 91
zip:= 593
rar:= 2
exe:= 139
png:= 2647
pdf:= 263
-----

Foremost finished at Thu Feb 13 16:07:09 2014
eprobst@brspeprobst ~
$
```

Figure 16. Recovery Summary

The summary can also be found in the used folder as a Foremost output. In there all recovered files will be separated in folders, where each folder stores files of the same extension.

Besides, Foremost will name each file with a hexadecimal numeric sequel. This name is equal to the offset (index) of the first Cluster where the recovered file was located. That happens because Foremost is not capable of identifying the name and original properties of the recovered file since this information

is stored in the archive system. Remember that Foremost does not recover based on registry check of the deleted files in the archive system but based on signature and structure analysis, subject treated in the beginning of this tutorial.

CONCLUSION

At the first part of this article, you learned on how to recover deleted files with a method based on deleted documents information location in the remaining entries of system files. This method is quite efficient when the disc was not formatted after file deletion and when the searched files were recently deleted.

At the second part, we saw another method that consists in analyzing all parts of the disc searching for known files headers and footers. Such feature makes it quite useful when the disc was formatted, which often makes the first method not suitable for use. This is the major difference between the first and the second method. To better understanding, see table below of both comparative methods.

Table 1. Comparison of File Recovery

	Method 1	Method 2
Analyze the file system	YES	NO
Analyze the file structure (header and footer)	NO	YES
Recover files recently deleted	YES	YES
Recover files deleted from formatted discs	NO ¹	YES
Recover files name and property (MAC Time...)	YES	NO ²
Recover metadata of overwritten or corrupted files	YES	NO
Recover fragmented files	YES ³	YES ⁴

REFERENCES

- Brian Carrier, File System Forensic Analysis, Publisher: Addison Wesley, ISBN: 0-321-26817-2
- Rohit Shaw, File Carving, <http://resources.infosecinstitute.com/file-carving/>
- Dave Hull, Fried FAT: A look into FAT file systems, <http://digital-forensics.sans.org/blog/2009/06/24/fried-fat-a-look-into-fat-file-systems/>
- Foremost – Linux man page, <http://linux.die.net/man/1/foremost>
- Recuva, Understanding Windows file deletion, <http://www.piriform.com/docs/recuva/technical-information/understanding-windows-file-deletion>
- Cygwin, Quick Start Guide, <http://cygwin.com/cygwin-ug-net/ov-ex-win.html>

FOOTNOTES

- 1 Other software to file recover, similar to Recuva, are able to locate and recover files from the file allocation table backups. However, most of this software are not free.
- 2 The Foremost is not able to recover file name and properties, as it does not analyses file system entries. However, some documents have information on their internal structure, as Microsoft Office documents.
- 3 The ability of recovering fragmented files depends on the files system of the disc submitted to the recovery process.
- 4 The ability of recovering fragmented files depends on how close the fragments are from one another and the structure of the located document.

ABOUT THE AUTHOR



Everson Probst is majored in Information Systems and is specialist in computer forensic, disputes and litigation. Guest Professor of the postgraduate course in computer forensics at Mackenzie, he has also taught at Faculdade Paulista de Engenharia – Instituto Brasileiro, Faculdade Paulista de Direito – EPD, Faculdade Impacta de Tecnologia – FIT and Faculdade Getúlio Vargas – FGV, in courses directed to Legal Experts throughout Brazil in partnership with AMCHAM and BSA. Senior consultant in computer forensic and electronic fraud investigations at Deloitte Touche Tohmatsu and member of the Research Committee for Standardization of Forensic Sciences ABNT/CEE-137 (Brazilian Association for Technical Standards) and ACFE (Association of Certified Fraud Examiners).

THE OTHERFTK! FORENSICSTHAT KONVICT!

by Christopher M. Erb

Recently released studies have shown 93% of criminal and civil cases in the United States involve some type of digital evidence. Large capacity storage media containing massive amounts of digital evidence and constant changes in newly released software continue to bring challenges to digital forensics. That being said, computer forensic examiners are regularly tapped to process and examine vast volumes of data while removing superfluous rubbish. Today, computer forensic examiners are fortunate enough to have a host of forensic software and hardware products available to them and their respective agencies / corporations. This article discusses the best practices to preserve, examine, and report the results of a digital forensic examination with the use of AccessData's Forensic Tool Kit® (FTK).

What you will learn:

- What precautions do you want to consider when handling digital evidence
- Is the examination authorized and what is the scope of your examination
- What pre-processing options should you consider selecting
- How to logically link a thumb drive to a computer

What you should know:

- What artifacts are relevant to the investigation
- How to explain your examination process and the results you have recovered
- How to effectively prepare, review, and report your results with an investigator or a client

First – a death investigation, an elected official, multiple newspaper reporters, and unauthorized access to a government website; what does this have to do with computer forensics? PLENTY! This case of illegal actions by many players proved to be a disaster for the politician and a serious security breach of a government website. In 2005, the Lancaster City (PA) Bureau of Police investigated an unattended death. A short “blurb” of this investigation appeared in the local newspaper. The brief article contained specific confidential information given to police by the 911 caller who reported the incident to police. As a result of the security breach, there was an extensive secondary investigation involving multiple agencies working jointly on the case. Ultimately, a statewide investigating grand jury was convened. The local elected coroner (*having close ties to newspaper reporters at the Lancaster (PA) Newspaper*

Office,) leaked his user name and password to the Lancaster County 911 computer system. The newspaper reporters used this restricted access to the government website to gather confidential information of crimes in progress for the purpose of preparing stories for print.

Second – young love, murder, social media, plus a getaway. An 18 year old man shoots and kills his 14-year-old girlfriend's parents and flees the state with the girlfriend. David Ludwig was secretly dating 14-year-old Kara Borden. Kara's parents learned of the relationship and informed Ludwig that the situation was unacceptable and would be coming to an end. Ludwig rebelled by shooting and killing Kara's parents in their Lititz, PA home and left the area taking Kara with him. Their whereabouts were unknown and information was limited. What was known was that Kara and Ludwig had computers, mobile phones, and accounts to social media websites. An immediate and intensive investigation quickly began. Among all of the chaos, computer systems and mobile phones were seized and examined. Ludwig was eventually captured. Kara was safe, and Ludwig plead guilty to two counts of first degree homicide. Recovered digital evidence played a major role in this case and Ludwig's decision to plead guilty to both first-degree homicide charges.

WHAT PRECAUTIONS ARE TAKEN TO COLLECT THE DIGITAL EVIDENCE?

In both of these cases, computer forensic examiners met with and educated lead investigators concerning actual and potential digital evidence. Search warrants were drafted, approved, and executed. The warrants were detailed making sure to present all known facts of the case and to connect the facts of the case to the digital evidence. When the respective search warrants were served, proper documentation, and chain of custody were paramount. Items were photographed, inventoried, and placed in sealed and labeled evidence containers or bags.

In many cases body fluids are present. To ensure safety from any harmful blood borne pathogens, and not destroy DNA or fingerprint evidence, latex gloves were used to collect and handle the evidence. Investigators or those responsible for collecting digital evidence must also be cognizant of potential hazards and not become complacent.

Items were taken to the computer forensic lab, where they were submitted for forensic analysis. Along with the physical pieces of evidence, copies of the investigative reports, search warrants, and a detailed lab request were submitted. Upon arrival all items were inventoried, placed in secured storage, and the submitting investigator was provided a receipt of the evidence. Documentation of the status and location of any evidence is a key element when keeping the chain of custody free from question by opposing counsel in future legal proceedings.

WHAT IS THE SCOPE OF EXAMINING THE DIGITAL EVIDENCE?

It is of the utmost importance for the forensic examiner to be diligent in reviewing all submitted documents. A complete review of the search warrant or other legal authority, as well as the lab request MUST be done. Examiners must ensure both documents match with respect to what the case investigator / client is requesting and what you as the examiner are authorized to recover, which is the data requested if it is present. During the examination, should an examiner happen to locate some important artifacts which are unrelated to the case at hand (or outside of the scope of the warrant,) the exam should temporarily stop and the submitting investigator should be notified immediately. For example, if the case being examined is a financial crime and the forensic examiner locates evidence of child exploitation, the legal authority does not contain permission to search for evidence of this "new" offense. Therefore, you cannot proceed with your analysis until the issue is addressed with the application of a second search warrant dealing with the new found crime. Data collected outside the scope of the legal authority can result in an argument for opposing counsel at future legal proceedings. There is a high probability that the recovered evidence will get tossed if a favorable opposing argument is successful.

An examiner should read any and all submitted investigative reports. This will inform the examiner of important details he may locate during the subsequent forensic exam. The examiner will also benefit from knowing the case when contact is made with the submitting investigator concerning the start of the examination and if any questions arise during the forensic examination.

Most importantly, when and if the case goes to court or other legal proceeding, the examiner will be in a position to understand the complexity of the investigation and how his analysis bolsters or corroborates the facts of the case.

WHAT PRECAUTIONS ARE TAKEN TO PRESERVE THE DIGITAL EVIDENCE?

Prior to any forensic examination, all equipment used should be validated and verified to safeguard the integrity of the digital evidence. Forensically-cleaned media is ideally used to store forensic images of the digital evidence. The practice of using clean media helps future legal questions from opposing counsel. The opponent may make an argument by asking questions similar to: “How do we know you did not already have that incriminating file on your computer before you started the exam?” or “How do we know if those files are not co-mingled with a previous case you may have worked on if you did not wipe the drive before examining my client’s computer?” A pre-exam MD5 hash must be obtained. Upon conclusion of the examination, a post-exam MD5 hash must be conducted again to verify the integrity of the forensic image. SHA1 or SHA 256 hashes may be done as well; however, that means more processing time is tacked on to the beginning and end of the exam.

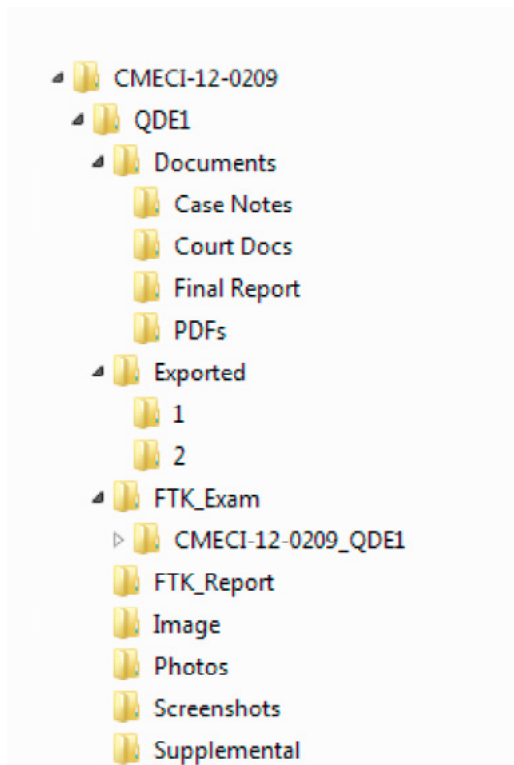


Figure 1. Sample of tree structure

The forensic labs where I work have closed forensic networks with connection to a Secured Area Network or SAN. A directory typically labeled by the case number is created and contains subdirectories for each piece of digital evidence. The subdirectory is titled by its uniquely identified evidence item. The subdirectory also contains additional directories and are typically titled, “Documents,” “Exported,” “FTK Exam,” “FTK Report,” “Image,” “Photos,” “Screenshots,” and “Supplemental.” A sample tree structure is depicted in Figure 1. I have found using directories helps to keep the case and all of the work on the case organized.

The imaging process is crucial! It is the foundation of the forensic examination and must be processed correctly with all of the safeguards in place. Typically, FTK Imager® is my tool of choice. After the media is connected to the exam machine via verified writeblocker, start FTK Imager®. I have made screen shots for each step taken in this process. The screenshots, (Figures 2 through 10) will walk you through every FTK Imager® prompt during the process of creating an image successfully.

During the down time of the imaging process, I check the BIOS time and date, the boot sequence settings and document the information in my case notes.

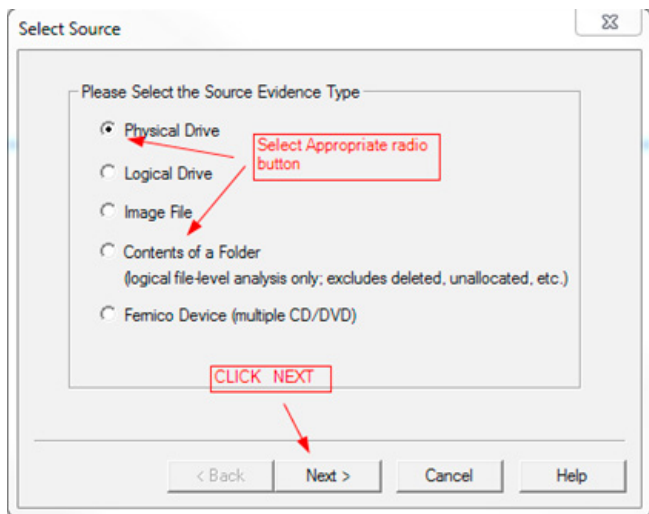


Figure 2. Creating an Image step 1

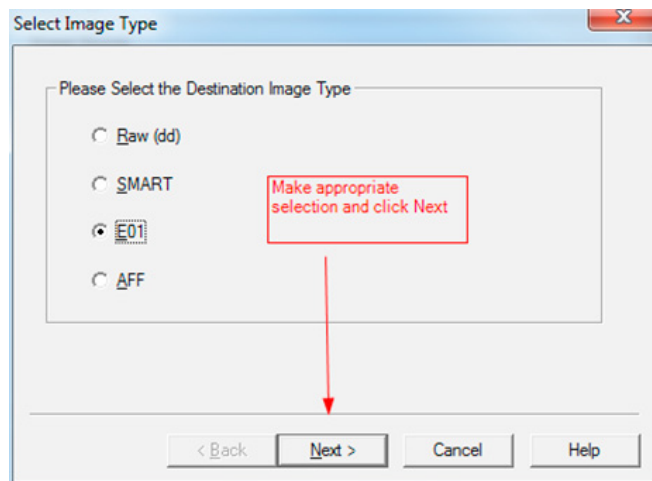


Figure 5. Step 4

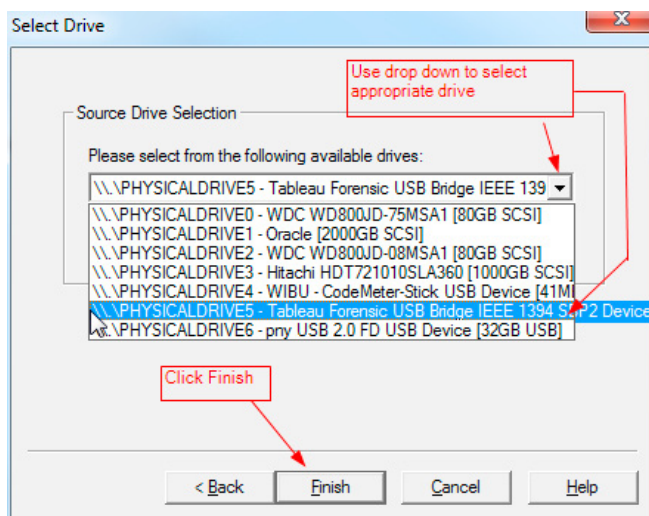


Figure 3. Step 2

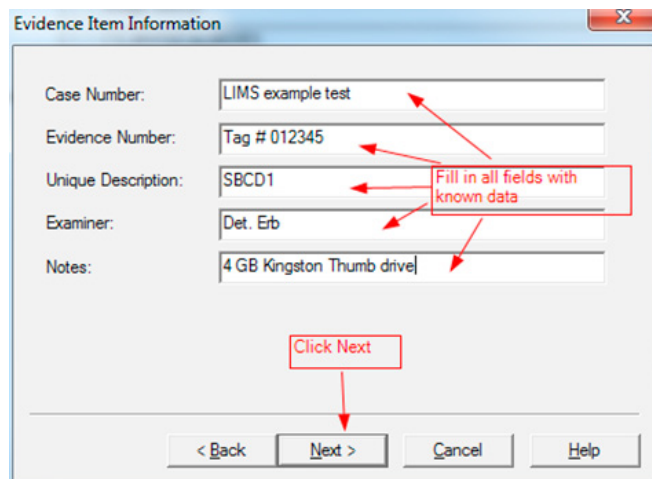


Figure 6. Step 5

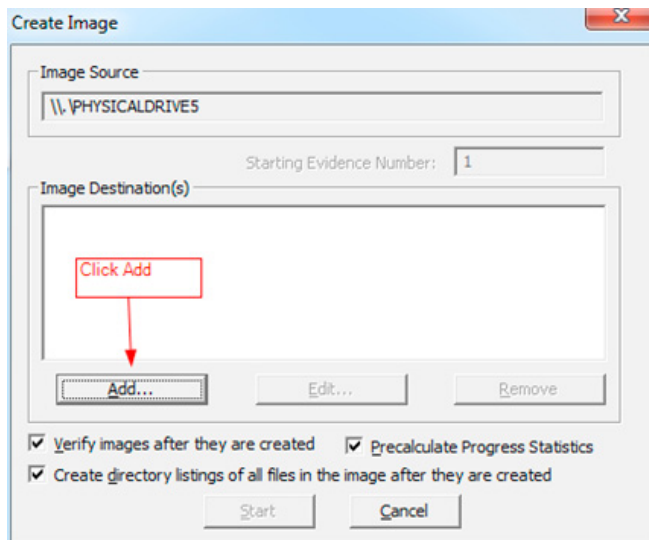


Figure 4. Step 3

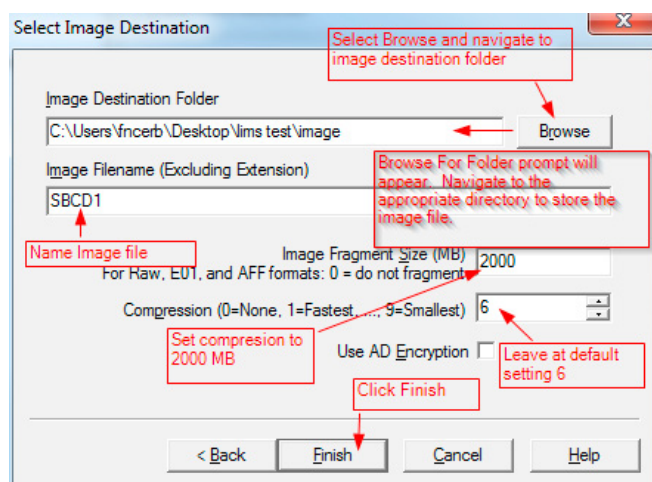


Figure 7. Step 6

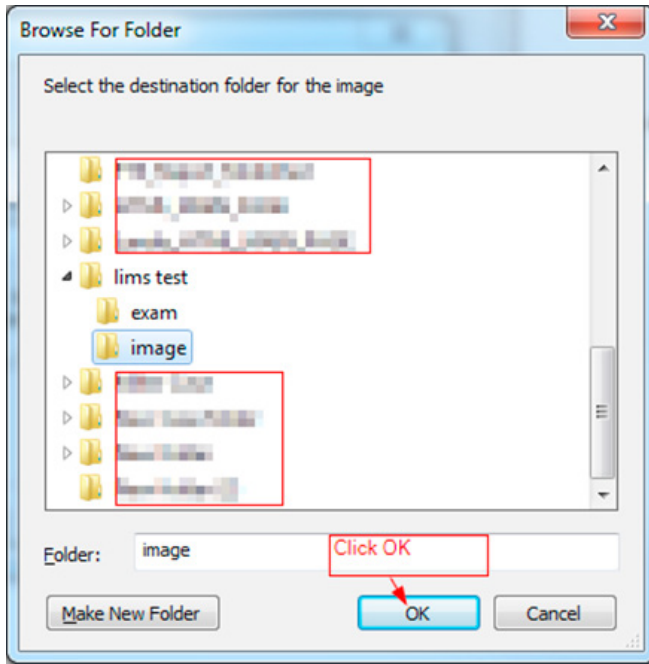


Figure 8. Step 7

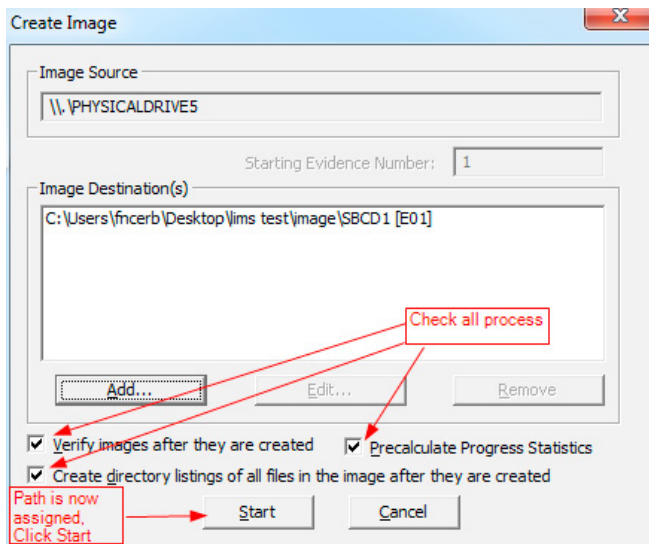


Figure 9. Step 8

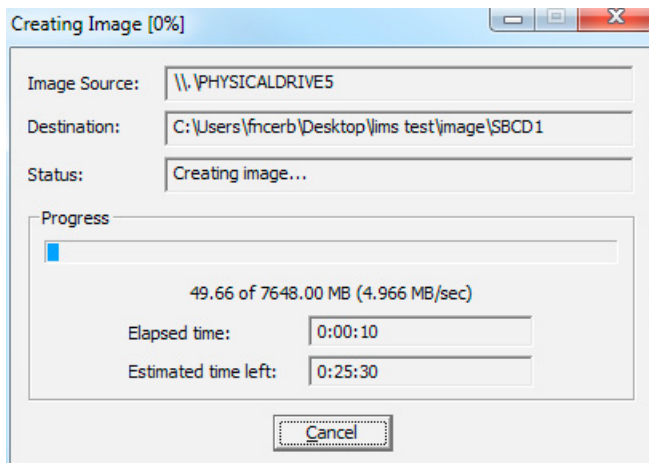


Figure 10. Step 9

Upon completion of the image process, a full directory listing will be created, and the image will self-verify. The verification process can sometimes take as long as the imaging process. When all of these processes are completed, a dialog box will indicate a 100% progress bar. I screen-captured these dialog boxes and incorporate them into a supplemental report as a common practice. I label it "Imaging Process" and store it in the "Documents" directory of the case. Figures 11, 12, and 13 are placed into this report. Notice on Figure 12 there is an "Image Summary" button. If you click this button the .E01.txt content will appear. I also highlight all of the text, copy it, and place the text into my Image Verification supplemental report. Notice on Figure 13 if you highlight any of the column labels, a description appears of its content in a text box at the bottom of the dialog box.

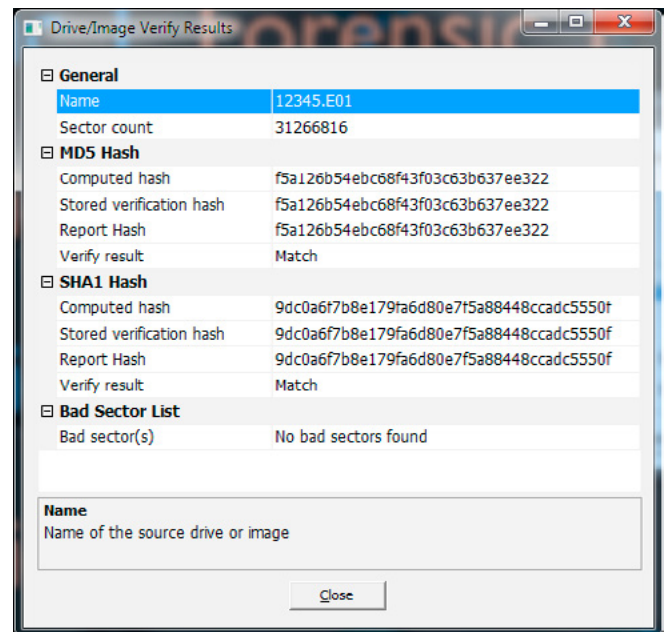


Figure 11. Creating image final steps

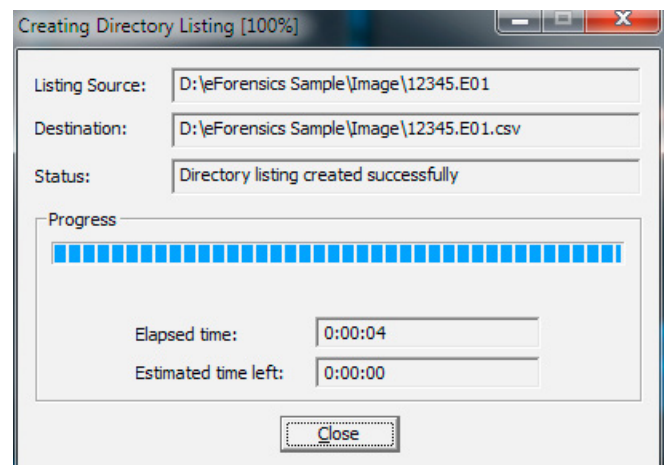


Figure 12. Image Summary Button

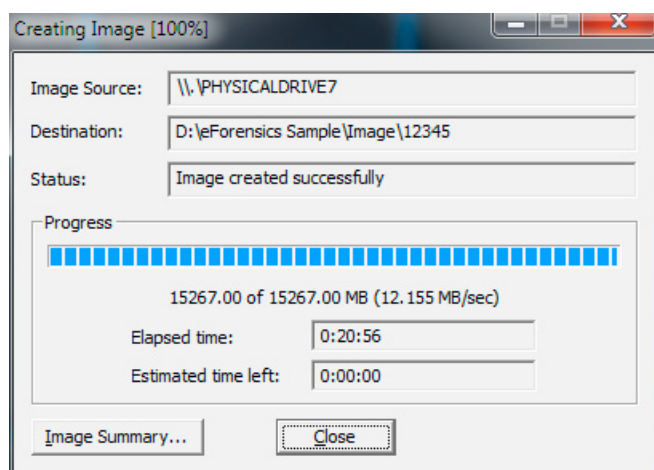


Figure 13. *Creating Image final steps*

Now that the Image process is completed, the next step I take is to close out all of the dialog boxes and close FTK Imager®. Then I power off the writeblocker, disconnect the media and if needed, reinstall the media in its original case and return it to secured evidence. The chain of custody will also be updated with the evidence returned to a secured evidence room / locker.

To confirm the image path and ensure everything is functioning properly, I open FTK Imager® again. I select the source evidence type “Image File” radio button. I navigate to the case directory, “Image” sub-directory, then select the .E01 file for the case, and open the newly created image. I expand the image directory tree structure to locate the following registry files:

- SAM PATH = C:\Windows\System32\config\SAM
- SECURITYPATH = C:\Windows\System32\config\SECURITY
- SOFTWARE PATH = C:\Windows\System32\config\SOFTWARE
- SYSTEM PATH = C:\Windows\System32\config\SYSTEM
- NTUser PATH = C:\Users\?????\NTUser(specific “user generated” profiles)

Once the registry files are located, I right click and select the “Export File(s)” option. I navigate to the case directory, then to the subdirectory “Exported” for the export destination. This accomplishes two things: 1) Ensures connectivity of the forensic network is working properly when the image files are loaded into the FTK software, and 2) The exported files can be processed in FTK Registry Viewer® during down time when FTK is processing the case.

WHAT PRE-PROCESSING OPTIONS SHOULD I SELECT?

The next step in this process is pretty much case specific. After logging into FTK® and creating your new case, you must select the options in which you want FTK® to process. The options an examiner chooses depends on what the case investigator / client needs for their respective investigations. If you already have customized process settings saved for a particular case, they may be selected at this point (*Consult the .pdf user’s guide to build a case processing profile found under the HELP tab.*) Decisions can also be derived from the scope of the legal authority, as well as the lab request submitted with the evidence on the front end of the case. Using Evidence Processing options of FTK v4.1®, I have prepared a screen shot describing the various functions available to an examiner. In Figure 14 below, selections are briefly described in red text boxes adjacent to each of the options. Some of the notations are a matter of personal preference; you may choose the options depending on the needs of your case.

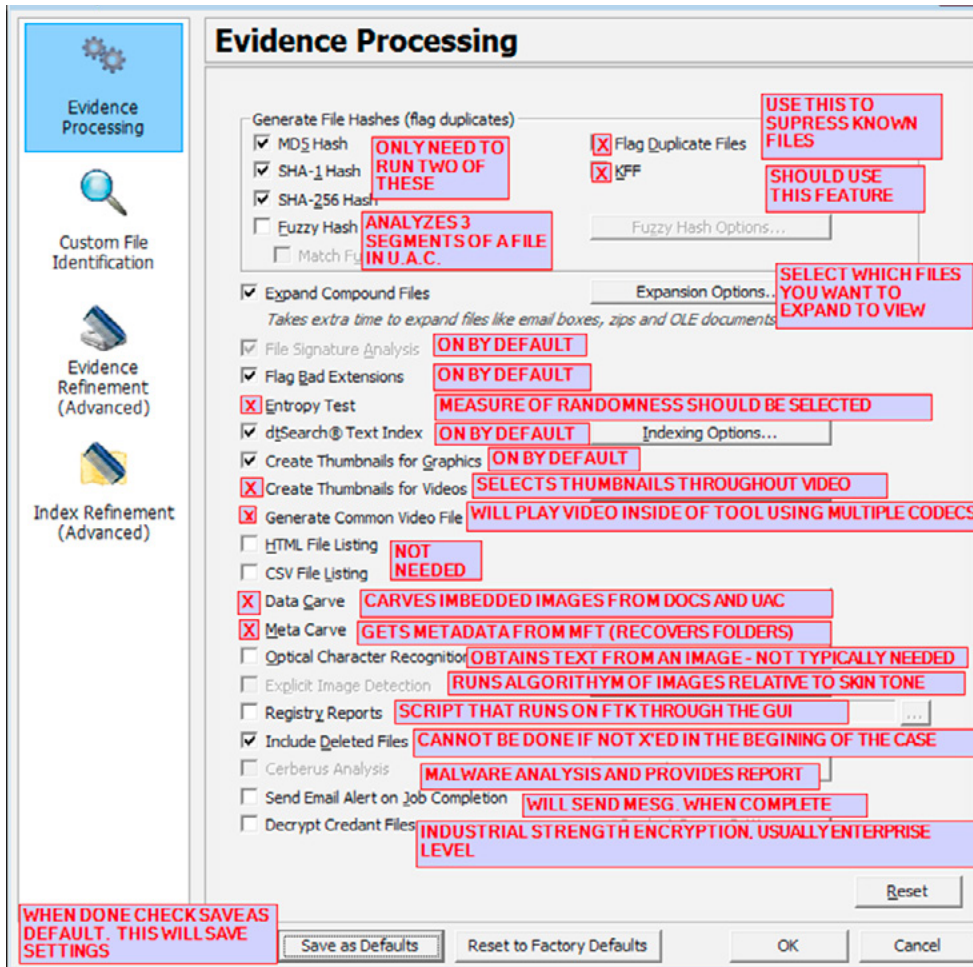


Figure 14. Briefly described selections in red text boxes adjacent to each of the options

In Figure 15 below, the data carving options are selected on a case by case basis. There is an option to select all types if you have unknown variables in your case. This may yield results in a homicide case when you really have no witness statement because your victim is deceased. Always remember to stay within the scope of your legal authority when doing an in-depth examination. If you would like to exclude known files, it is a good idea to select the “Exclude KFF Ignorable box in this dialog screen.

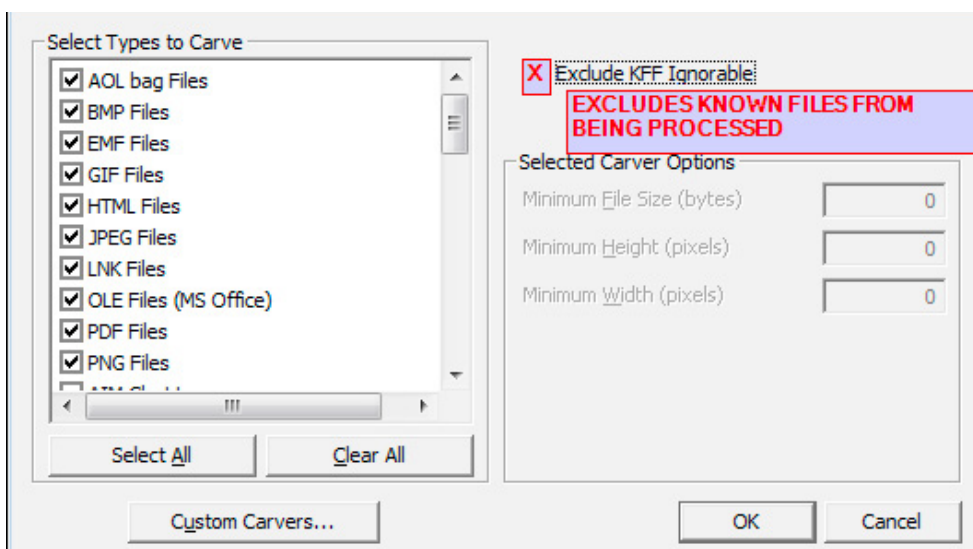


Figure 15. Data carving options are selected on a case by case basis

Figure 16 shows the Evidence Refinement options. This can be accessed by clicking “Evidence Refinement (Advanced)” option button along the left column of the Detailed Options screen.

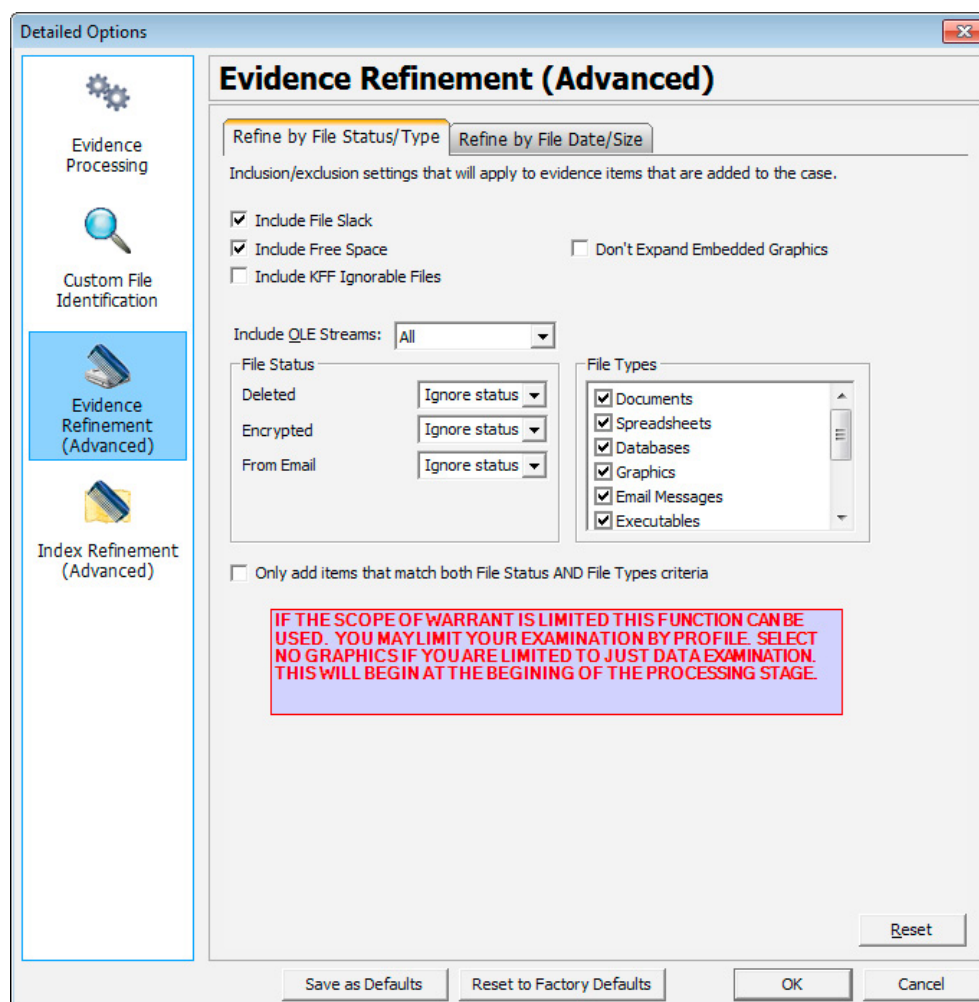


Figure 16. Evidence Refinement options

In the Ludwig homicide case, David fled the area in one of his family's vehicles. It was assumed that Kara was with him immediately following the murders; however, their whereabouts were unknown. Family and friends were not in communication with either of them for days following the murders. Several witnesses, close friends, and family members of the Bordens and the Ludwigs were questioned by police to obtain facts about this case. It was quickly determined for months leading up to the murders, David and Kara communicated in various ways: cell phone calls, text messages, and the social media site Xanga.com. Investigators knew photos were sent to and from one another on their cell phones and as postings on social media sites. Both David and Kara left their cell phones behind inside their respective homes. The phones were collected as evidence with search warrants. Any and all computers, thumb drives, and media cards left behind by David and Kara were collected as well. When this incident occurred, FTK v1.x® was used. Many processing options that are now available could not be selected.

In the second case involving the rogue coroner, I assisted Agent Robert Drawbaugh of the PA Office of the Attorney General. Agent Drawbaugh and I interviewed many witnesses, suspects, and other people of interest. We collected volumes of information over the course of several months. Court orders and subpoenas were authorized and issued by the judge presiding over a sitting statewide investigating grand jury. Throughout the course of this investigation, we were able to pinpoint specific pieces of information we knew had to be stored on digital media. That data would be helpful in corroborating information gleaned from our investigation. The information included email communications to and from all parties involved in this case, internet history, web pages, documents, images, etc. Search warrants were executed in multiple locations. The coroner's residence, office, and the newspaper office were served, and many additional items were collected.

Agent Steve Arter, and other members of the PA Office of the Attorney General's Computer Forensics Unit, as well as Detective Peter Savage of the Lancaster County District Attorney's Office, and Cpl. Jim Strosser of the PA State Police Computer Crimes Task Force helped examine the digital evidence in this case. We worked cohesively to locate and recover data of evidentiary value. FTK v1.x[®] was one of the primary forensic tools used in this investigation. Many of the automated processes available in today's products were not available. In an effort to recover valuable data in both cases, time consuming manual searching of data was the order of the day. Examiners created and entered search scripts in the "Live Search" and keywords in the "Index Search" utilities built into the forensic software. Although it was very time consuming, this process proved to be incredibly effective.

Ultimately, the grand jury indicted the coroner on multiple charges. Agent Drawbaugh and I were co-affiants and criminally charged the elected official with a host of offenses. He eventually plead guilty in a plea agreement. The facts of each of these cases, and in the cases you are assigned to work on (whether they are in a criminal or civil arena), will determine what you are going to process, and guide the pathway to analysis.

OK, back to the steps. Once all of your processing options are selected, click "OK." After a brief refresh, as FTK[®] builds the case database behind the scenes, FTK[®] will reappear. Add the evidence to your case. A job-progress window will appear to inform you of the status or which processes are being performed, (see Figure 17 for an example).

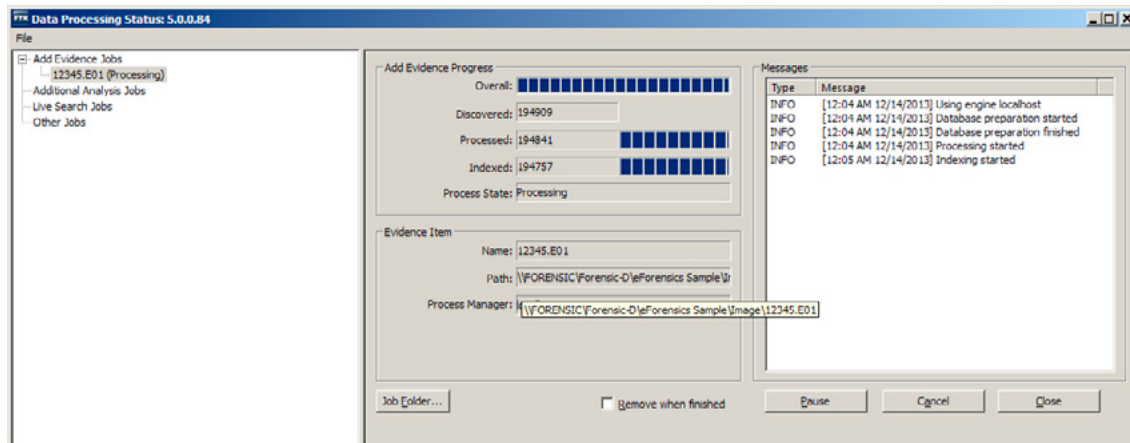


Figure 17. Job-progress window

When processing begins, the image is loaded into FTK's graphical user interface. Processed objects begin to populate designated containers. You are able to review data as the objects are populating; however, constant refreshing occurs. Each time a refresh occurs, the objects you are viewing are interrupted. Ideally, it is best to let the software run uninterrupted. This is where registry viewer comes into play.

LOOKING FOR THE ARTIFACTS

Earlier registry files were exported into the export directory of this case. Open AccessData Registry Viewer[®]. Click the "Open Folder" icon and browse to the exported registry files which have been exported.

The registry files which have been exported previously may now be examined. The data gleaned from a registry analysis may be added to the report generator, and the software generated report can be saved to the forensic network in the case directory. I usually store the information in the created sub-directory; FTK_Exam\FTK_Report\Registry_Reports which was created during the Case Creation (this directory path may vary on your system depending on your preferences).

Although there is a lot of information available during the registry analysis, I typically include the below listed registry information in a registry report to be reviewed by the investigator / client:

- SAM
 - User Accounts including SID numbers
 - Last logon
 - Number of logons

- Number of failed logons, etc.
- SOFTWARE
 - Current Version of Operating System (OS)
 - Registered Organization
 - Registered Name
 - Latest service pack installed, etc.
- SYSTEM
 - Select (note the current control set value)
 - Time Zone
 - Mounted Devices
 - USBStor
 - TCPIP data
 - TCPIP Settings
- NTUser
 - Typed URLs
 - PSSP
 - Intelliforms
 - Recent
 - MRU
 - Prefetch (if applicable to case)

When completed, review the registry reports. Typically, a review of registry files will contain information where some of these important case specific artifacts may be located on the media. This may help guide you to some deeply hidden directory within another directory such as “Program Files” or “Windows”.

After the review of registry information and the evidence is processed in FTK®, view the overall tree structure of the suspect hard drive or whatever digital evidence you are examining. Look for, and note, any anomalies. They can be anything from a questionable filename, to an odd named directory such as “Dell” on an HP machine.

The artifacts are parsed out in their respective containers of FTK’s interface. When talking to your investigator / client, a keyword list of important words, phrases, numbers, or specific file names can be searched using the “Index Search” engine built into FTK®. Results will appear quickly and they will be listed in either allocated or unallocated space on the suspect drive. By expanding your results and selecting them, the content of the file is recovered. It can be bookmarked for purposes of being included in the forensic report generated by FTK’s report wizard. Not in every case, but sometimes you may have to consider manually carving out artifacts from unallocated space. This can be a time consuming task, but it may be required. Custom carvers can be created (consult FTK® Users guide), or you may carve data from the unallocated space manually on the fly. Information in the user’s guide will help you define what to carve out and how to bookmark the data.

Depending on the type of case this examination is involving, consider some or all of the below listed locations and/or procedures (depending on the operating system) in an effort to recover the artifacts for the investigator / client:

- Examination and review of Recycle Bin, or Recycler
- Attempt to match any removable media to the computer system
- Deleted Files
- Keyword searching for specific terms
- Graphics
- Multimedia
- Documents
- Spread sheets
- Databases
- Synced cell phone information
 - Texts – MMS and/or SMS
 - Email
 - Contacts
 - Calendars

- Images
- Call logs
- Voice messages/memos
- Internet History
 - Browsing History
 - Cache
 - Cookies
- Internet searching
 - Note which browser searched for what information and under whose profile
- Internet Bookmarks, or Favorites
- Web pages
- Social Media Chatting/Activity/Buddy listings
- Email & associated attachments
- Listing of other software installed on the system
- Password protected or encrypted files (depending on the version used)
 - Use PRTK®
 - Export word list
 - Use SAM, SYSTEM, and NTUser registry files to build profile
 - Export associated files to the “Exported” directory
 - Drag and drop relevant files into PRTK®
- P2P file sharing or newsgroup activity
- Pagefile analysis
- Manual data carving of relevant items located in unallocated space
- Generation of timeline if available
- .LNK files
- Volume Shadow Copies
- Inclusion of pertinent metadata associated with any of the above
- If there is a need to use proprietary, or third party software to view artifacts, FTK Imager may be used to mount the image file as a read only drive to view the data

The artifacts listed are obviously not a complete list of items of value; however, they will be helpful to your investigator / client. Document procedures or steps taken to recover the data in your case notes!

.LNK files (mentioned above) could be important in your case. Screen-captures of .LNK files can be documented and included into reports. .LNK files can provide information such as a thumb drive being connected to your suspect’s computer. Time and date stamps can be included along with the file name and, more importantly, the volume serial number (VSN). In the examples shown below Figures 18 and 19, you will see how a match can be made. In Figure 18, a screen shot of FTK’s interface is depicted. The “Explore” tab is selected and the Evidence Item “N@#\$%^&P.E01” is highlighted. In the viewing pane on the right, I have selected the “Properties” view tab. You can see the highlighted VSN is F859-FF4F. Through the course of my examination I was able to locate a .LNK file (Figure 19 which is a screen shot of the .LNK file, can be bookmarked and viewed in a .html view). The .LNK file on the computer represents a record created when the thumb drive was suspected of being inserted into a USB port. Both volume serial numbers were a match, confirming the thumb drive had been connected to the computer; probably on the date and time shown on the .LNK file. From this information, along with the hash values matching on this particular file being recovered from the computer and the thumb drive, one can believe with certainty, this file resided on both pieces of media and the thumb drive was connected to the computer. Additionally, the “System File” registry report indicated a match to the manufacturer of the USB thumb drive which was plugged into the computer.

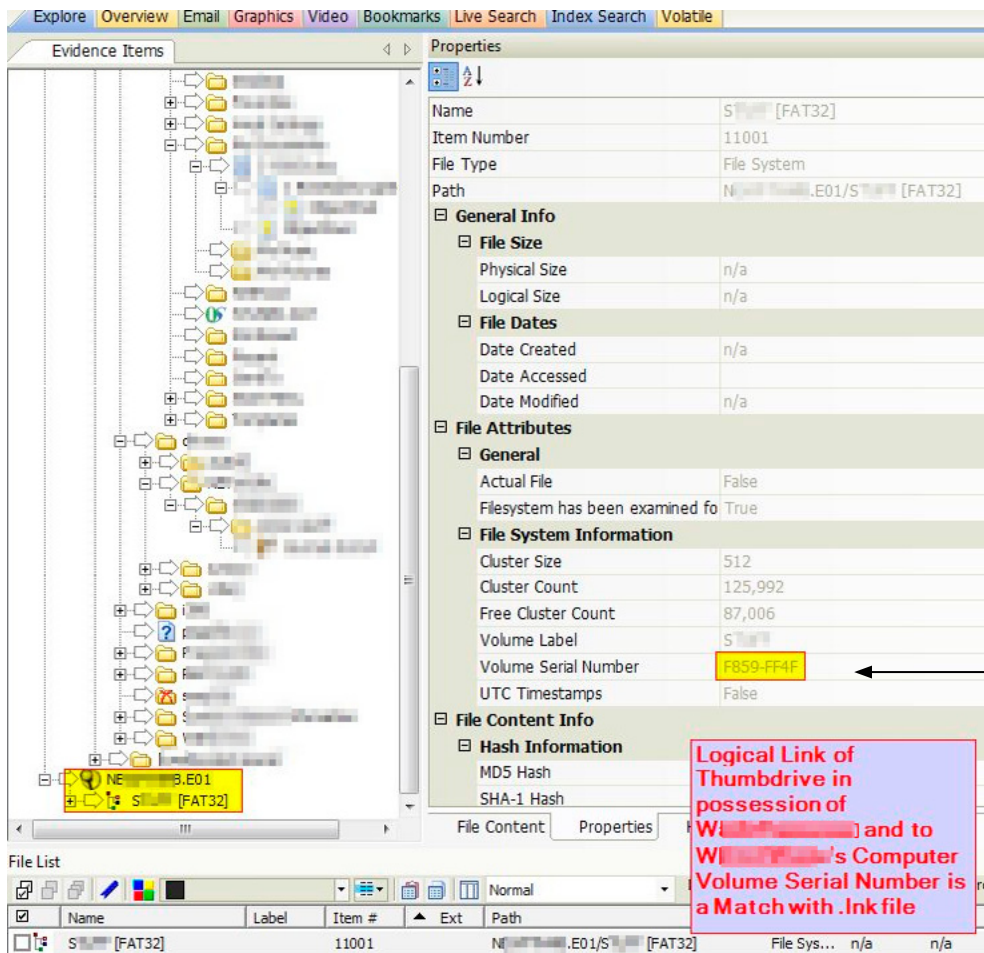


Figure 18. VSN of thumb drive

Shortcut File

Link target information	
Local Path	I:\[REDACTED].jpg
Volume Type	Removable Disk
Volume Label	S[REDACTED]
Volume Serial Number	F859-FF4F
File Size	201197
Creation time (UTC)	5/2/2007 5:49:19 AM +00:00
Last write time (UTC)	3/13/2007 3:38:20 PM +00:00
Last access time (UTC)	5/1/2007 4:00:00 AM +00:00
File attributes	
Archive	
Optional fields	
Working directory	I:\

Figure 19. LNK file from computer

All artifacts of value recovered as a result of the examination may be bookmarked for inclusion on the forensic report generated by the software. Generate the report in .html, word document, or .pdf format. The forensic report will be stored in the "FTK_Report" directory created on the forensic network within your respective case directory.

From the start of the forensic examination to the completion, every step taken must be documented in great detail. Documentation of the examination is just as important as the forensic artifacts you include in the software generated report. Case notes should include a complete description of the evidence, the hardware and software (including versions), and all facets of the actions taken during the examination. The reasons for this are two-fold: 1) During legal proceedings you will need to accurately recall what you did and how you came to find the artifacts you are reporting, and possibly render an opinion. 2) If your examination has to be replicated by a peer reviewer or opposing counsel, the other examiner will need to do exactly what you did to confirm your results.

CASE EXPLANATION AND REVIEW WITH INVESTIGATOR / CLIENT

When the examination has been completed, and you have satisfied the request, contact the investigator / client and advise them you are ready to review the case with them. It is very important the investigator / client understands (sometimes in layman's terms) what you have recovered and how the information plays a role in the case. When reviewing the case with the investigator / client do not be surprised if there is a lack of technological comprehension. Many times examiners take for granted the audience we are engaging knows, or has an understanding of, today's technology as it relates to digital forensics.

It may be necessary for examiners to allocate much of the day to completing a case review simply because you may have to educate your investigator / client as you are showing them the results. Additionally, the results may be plentiful and the investigator / client may feel overwhelmed with large amounts of data. The investigator / client should be familiar and confident with these results to finalize his investigation.

When the review is coming to an end, the investigator / client may have additional questions. Be certain to answer them completely and confirm the exam and the subsequent results are satisfied. Upon conclusion of the case review, confirm with the investigator / client to close the case and finalize your reports. When confirmed, conduct an image verification to ensure the integrity of the original evidence.

IN SUMMARY

When forensic examinations were conducted for both of these cases, examiners followed the aforementioned methodology. In the Ludwig murder case, deleted incriminating instant messages, video files, images, web pages, and emails were recovered. Mobile phone analysis produced voluminous amounts of call logs, text messages, and exchanged photograph images. Collectively all of the data was combined with the facts of a traditional criminal investigation and blended together well. Prosecutors claimed victory in this case, due in part to the abundance of incriminating and indefensible data recovered forensically.

In the case of the coroner, Internet activity and the numbers of visits to the government 911 website were numerous. The newspaper reporter's activity on the "secured" website had been continuous for nearly a year leading up to the investigation. Multiple email communications between the coroner and reporters were recovered, some of which were deleted. Forensics played a major role in this case. Results coupled with investigative efforts lead to an indictment and an eventual guilty plea. The use of FTK and its suite of forensic utilities were instrumental in the successful conclusion of these and many other cases I have worked.

Throughout the course of my career in law enforcement, as well as the private sector, I have found the steps included in this document to be very effective. Keeping current with constant changes in technology is important to examiners who need to be flexible and open to change with the technological advances.

ABOUT THE AUTHOR

I have been in law enforcement since 1989, starting with the Lancaster City (PA) Bureau of Police. I retired as a detective in 2010. I served my last year-and-a-half as a "Task Force Officer" with the F.B.I. – Philadelphia Regional Computer Forensic Laboratory, completing the C.A.R.T. curriculum. Since retiring from Lancaster City PD, I have started my own computer forensic business, CME Cyber Investigations, LLC, located in Lancaster, Pennsylvania. Additionally I am a detective with the Berks County (PA) District Attorney's Office – Forensic Services Unit, where I am assigned as a fulltime computer forensic examiner. I have obtained my ACE and AME certifications from AccessData, as well as a S.C.E.R.S. Certification from the US Department of Homeland Security. Additional profile information of me can be found at <http://www.linkedin.com/pub/christopher-erb/50/79b/829/>

Total Cyber Security Solution

Analyze, Cure, Prevent



TOTAL CYBER SECURITY SOLUTION

Frogteam|Security unique solution allows organizations, companies and security administrators to:

- **Analyze** organization cyber assets (Cloud:Scope).
- **Cure** using Sec:Cure by correlating analysis results with an easy to use fix module (Sec:Cure).
- **Prevent** using Signa:Gen - TCS Cyber Seal is a sophisticated active and live client that is able to detect and prevent different cyber-attacks techniques and vectors.

Three easy steps To Secure Your Assets!

Our total solution enable you to Analyze, Cure and Prevent from cyber security threats and vulnerabilities



Why TCS Cyber Seal is important?

TCS Cyber Seal helps building consumer's trust. With the majority of shoppers' continued concern when providing personal data online - using the Signa:Gen for websites' seal of security will help you concentrate on expanding your business. Signa:Gen - TCS Cyber Seal product objective is to ensure the safety of e-commerce business over the internet. This can be achieved through independent check by the appointed organization which certifies qualified merchant(s) or company(s).



For more information visit our website at: <http://www.frogteam-security.com>

Frogteam|Security Ltd
E-mail: info@frogteam-security.com
Website: www.frogteam-security.com

Corporate Headquarters
1875 Century Park East #700
Los Angeles, California 90067,
United States
Tel: +1 (408) 504-4903

Special Offer for eForensics members
Scan this QR barcode to register
with mobile now and get Special
Offer of 10% discount.



FTK – IMPROVING PASSWORD RECOVERY

by Brian Mork

Password recovery is the most effective way to uncover data and information that suspects try to hide behind encryption, but it can also be very time consuming. In this article we will explore techniques and technologies that can dramatically reduce the amount of time required to recover passwords, and let you spend more time focusing on the actual investigation.

Let's be honest. In our day to day forensics work, it is far more likely for us to encounter a user who has saved all of their passwords in a text file than anyone outside the forensic realm would ever guess. If a suspect hasn't written down their password, it is likely as not to be along the lines of "password" or "123456." On those rare occasions when a "complex" password is chosen, it will often conform to the pattern of "word from a dictionary with a capital first letter, followed by a single number or special character." Those of us who are lucky enough not to spend our days working organized crime cases will find the case where we have to recover a password of any real complexity to be the needle in the haystack.

However, when we do come across the security expert or paranoid individual that has not only chosen an exceptionally strong password but hasn't left a solid trace of it on their system, what are we to do? Brute forcing a 32-character password will take longer than most investigations have time to devote to it, and the evidence that is most closely protected can be critical to establishing the activities of a suspect. This is the time where a well-developed password cracking program and procedures can make the difference in the case. In this article we are going to look at four separate improvements that can dramatically improve the success rate of password recovery efforts: improved wordlist generation, a dedicated Linux password recovery system, effective use of hybrid attacks, and using distributed password recovery operations.

OVERVIEW OF FTK CAPABILITIES

Before we start looking at how to improve our operations, we should have a baseline. For the purposes of this article I have chosen to use AccessData's Forensic Tool Kit (FTK) 5.2 as my baseline. FTK is one of the top suites of tools used by forensic investigators and examiners today to ingest, process, analyze, and track digital evidence. It provides a built-in capability to attempt decryption of a wide range of files, based on code from the standalone Password Recovery Toolkit (PRTK).

NOTE

AccessData provides a robust product in its PRTK and related Distributed Network Attack (DNA) products; PRTK is a standalone password recovery application designed to be run on a single system, while DNA is designed to break up a password recovery operation across multiple systems for greater efficiency. Both products offer substantial improvements in speed and capability over the techniques available within FTK itself. This article has been written to address the case of individual investigators or organizations that need even more advanced password recovery techniques and who are willing to invest some extra manual effort.

I generally separate password recovery efforts into using one of five distinct attack methods:

- **Algorithm Attack** – While it is not often seen, there are cases where a suspect has used an algorithm that has known cryptographic weaknesses. In such cases, an attack on the algorithm itself can sometimes be leveraged to recover the protected content and/or the password.
- **Brute-force Attack** – This method is an exhaustive search of an entire key space. While it is often performed sequentially (“a”, “b”, “c” ... “aaaa”, “aaab”, “aaac”, etc), more advanced approaches will alter the ordering of search patterns based on the probability of their occurrence relative to one another, such as search “the” before “zqx”.
- **Dictionary Attack** – This is the simplest attack method, literally involving a list of potential passwords, each of which is tested to determine if it decrypts the protected content.
- **Implementation Attack** – Implementation attacks rely on poor programming decisions to allow encryption to be bypassed. Common examples of this are when a particular program stores the password (normally XORed with a known phrase or similarly “protected”) within the content that is encrypted. Another example is when protection of data occurs only by having the container used to store the encrypted content requiring a password. For example, a program may store the protected content in cleartext within a file but programmatically require the input of a password to display the content.
- **Hybrid Attack** – Hybrid attacks are also sometimes called rule-based attacks. In the most common variant of this approach, a dictionary of words is used as a seed and a series of rules (“capitalize the first letter”, “add ‘2014’ to the end of the word”, or “substitute ‘\$’ for the ‘s’ character”) are applied to each word, as well as in combination. Hybrid attacks allow for the rapid generation of uncommon passwords from common words by mimicking the way humans often select passwords. For instance, a seed of “password” can generate results such as “p@ssword!”, “Pa55w0rd2014”, or “!4\$\$/0P,c”.

FTK PASSWORD RECOVERY LIMITATIONS

As mentioned previously, FTK includes some degree of password recovery capability within the application itself, but this is limited to basic dictionary attacks. In the normal use case, an examiner will compile lists of words that are frequently used by the suspect. This may be through parsing emails, examining memory dumps, or documents that are accessible. These words are then added to a defined password candidate list and tried in sequential order when a decryption attempt is requested.

This approach is simple to implement, and in many cases will yield a fair degree of success. In my experiences, about 40 percent of user content will be broken from a simple parsing of their email history to generate a word list. What about the remaining 60 percent however?

IMPROVEMENT ONE: IMPROVED WORDLIST GENERATION

The single greatest improvement you can get relative to cost is to improve the manner by which you generate wordlists. There are a number of ways to do this. For example, I personally like building a wordlist in FTK from a Live Search using the following regular expression:

```
|<\w+>
```

This search returns all single words found in the evidence, including memory dumps, unused drive space, documents, and files. Specifically, the search looks for a non-word character (such as a space, tab, or non-printable character), followed by one or more word-characters (lowercase and uppercase letters as well as numbers), and terminated by another non-word character. Using this approach to generate a wordlist for a search actually emulates one of the built-in techniques that FTK will use to seed the PRTK and DNA tools, but you can perform the same function yourself without the need to purchase any additional software.

Another favorite technique of mine for improving wordlist generation is using special interest websites, especially wikis. For instance, I was working a case a few years ago where the suspect had employed strong encryption to protect some files that we believed to be of importance to our investigation, and the usual attacks had failed to turn up a password. I noticed early on that he was quite interested in the show *Farscape* (a science fiction television show from the late 90s to the early 2000s). He made reference to several of the characters in his emails, his browsing history showed active participation in writing fan fiction for the show, and he had even re-themed his entire desktop with a custom *Farscape* theme that he personally developed.

Given the unique names and references typically associated with science fiction, I thought there was at least a decent chance that the password was related to the show, so I used one of my favorite scripts called CeWL (authored by Robin Wood) to spider a few websites related to the show and generate a wordlist that was specific to that universe. My normal syntax for such a search is:

```
./cewl -d 2 -m 4 -w newwordlist.txt -v http://www.example.com
```

The above command will spider a from <http://www.example.com> to a depth of two pages, collect all words of four or more characters, and output the results of that search to a wordlist in the current directory named “newwordlist.txt”.

Several of the passwords fell to that approach, but there was still one large archive that we couldn’t gain access to. In the end, we actually did gain access, and through a similar means. I set up a spider on a website hosting the scripts and processed them to strip out stage directions, indicators of who was speaking, and join multiple words together, so that this:

“Crais: We are somewhat – inconveniently tethered inside a Budong.”

Became these passwords:

- Wearesomewhatinconveniently
- aresomewhatinconvenientlytethered
- somewhatinconvenientlytetheredinside
- inconvenientlytetheredinsidea
- tetheredinsideaBudong

That approach finally broke the archive file, with a phrase that included cursing in a fictional language. Appropriately tailoring and cultivating wordlists can have a substantial impact on your password recovery success rates. In fact, I maintain a personal library of thousands of such lists resulting from investigations and competitions, as well as proactive analysis, everything from wordlists on famous scientists to Pokémon to religious figures and terminology.

While AccessData’s PRTK and DNA offer the capability to parse the entire drive and use the index as a starting point, that capability is unfortunately not present in FTK. Interestingly, most of the capability we just examined in the script parsing is present in PRTK and DNA if all of the words are loaded into a dictionary (they are actually captured in the PP-3-03 passphrase rule), but that approach actually ends up being less efficient in terms of probable recovery times since all permutations will be tried rather than just combinations relative to the source material.

IMPROVEMENT TWO: DEDICATED LINUX BOX

An improved wordlist will greatly increase your chances of recovering a password, but choosing the right tool for the job will also have a substantial impact. Much as no two database connection tools are the same, the same applies to password recovery tools. For instance, when trying to recover the password of an older Unix system that stores passwords in the shadow file using MD5 to represent the password, there is no tool in my experience that performs the analysis faster than Hashcat. If I need to recover the password for a Microsoft Word .docx file, however, Hashcat doesn’t have such a capability, and then I turn to John The Ripper (and office2john).

It is certainly possible to run most of these tools on a Windows machine, but for the lower overhead levels that can be achieved and the greater range of pre-built selections I prefer a 64-bit Linux build, nomi-

nally Kali Linux. At present, I am using two single processor, 8-core 64-bit Kali Linux machines, each with 32 GB of RAM and two dedicated graphics processing unit (GPU) cards.

As one additional note, it is also important to maintain awareness of what type of program is best suited to each type of algorithm. GPU-based attacks are excellent for iterative hashing approaches with a wordlist (where a given plaintext is hashed hundreds or thousands of times as a preventative measure against brute force searches), but are actually disadvantaged when each operation only needs to be performed once. For example, straight MD5 calculations actually spend more time loading the wordlist into the GPU memory than actually performing the MD5 calculation in the GPU, leaving them more efficient as CPU operations.

Over the years, I've built out a simple table to show to new practitioners as a quick reference as to which tool to use. While there are certainly edge cases (notably instances where a GPU-based attack is not available for a particular algorithm), it provides a quick starting point to narrow down to the appropriate tool.

Table 1. Password Recovery Attack Type

Algorithm Type	Attack Type	Processing Unit
Single Iteration (e.g. MD5 or NTLM)	Dictionary	CPU
	Hybrid	GPU
Multiple Iteration (e.g. md5crypt or phpass)	Any	GPU

While FTK doesn't provide any GPU acceleration in its jobs, the components in PRTK and DNA don't actually do so either. At the time of this writing the only decryption tasks in PRTK and DNA that task GPUs are WinZip 9 and Microsoft Office documents. By comparison, oclHashcat (a freeware password cracking tool favored by many in the password recovery domain) supports more than 85 algorithms via GPU.

To put that into more concrete terms, using FTK with the "RockYou" wordlist (a popular collection of leaked passwords that contains 14 million unique passwords) against a md5crypt of a password not in the wordlist (such that all combinations must be tried) takes PRTK 7.4 seconds to complete, while JtR it completes in 1.6 seconds and oclHashcat finishes in a mere 0.9 seconds. All tests were run on the identical hardware using the most recent stable release versions of each program, with Windows Server 2008 R2 (64-bit) for the PRTK test and Kali Linux (64-bit) for the oclHashcat and JtR tests. Put another way, using the Linux password recovery tools resulted in a 68 to 88 percent decrease in recovery time.

ENHANCEMENT THREE: HYBRID ATTACKS

Hybrid attacks are the difference between finding a password of "OldYeller" and "Oldy3113r." While it is immediately apparent to the naked eye the relationship between those two passwords, from a complexity (and entropy) standpoint the latter password is more difficult. Humans have a tendency to take things that are easy to remember, couple that with rules that are easy to remember, and generate seemingly strong passwords from that approach.

In certain instances, this holds true. If we take the first letter of each word in the phrase "My yellow watermelon could have sold for \$8 on Tuesday but I held out for 10 trucks on Friday" we get "Mywchsf\$8oTbIhof1toF". The first phrase is easy to remember, the second one is a fairly strong password. Fortunately for us, most people are more likely to follow the "Oldy3113r" approach, and those types of rules are very easy to model.

Most of the tools that are used for password recovery at this point allow for the use of some rules or permutation engine, and the intent is the same for each: provide a way to specify how to translate human rules into all the variations that may be caused by them. While the built-in rules are an excellent starting point, I always like to perform modifications of my own, such as the following:

- *Create a rule set that substitutes one special character for another:* This rule comes and goes in popularity over the years, but it's a constant for my efforts. One of the most common permutations I come across is one where a simple substitution such as a "!" for a "l" then gets shifted by one character to

the right. For example, the password “Hello” is changed to “He!!o” and then to “He@@o”. Using a special character substitution allows you to catch these changes in a combinatory manner, because the first expansion (from a built-in rule) will trigger the second (from a manually created rule).

- *Create a rule that appends the same special character to the start and end of the seed word:* This is a personal favorite due to the number of times I’ve heard it explained to me. “Well, I just used my daughter’s name and added an asterisk on either side of it” is a typical response. When people are required to inject special characters they often do so at the beginning, the end, or both. This rule catches the third case.
- *Create a rule that appends every 4-digit number (0000 through 9999) to the end of each seed word:* Whether it is the year they were born, the year their grandfather came back from the war, or the year they plan to retire, years are frequently added to basic passwords in an attempt to meet the complexity requirements of a system or quickly pad the length.
- *Create rules based on patterns of character types, not just words:* Be aware that suspects may avoid using common patterns, but that patterns may still exist. If you see an email signature from a computer criminal that is signed with “eViLgUY” then build out a rule to combine words and invert the normal cases. If they sign it “eViLgUy” that can be templated and captured in a rule as well.

One last part of the hybrid attack that is often overlooked is recombination. If you are working a case and find multiple passwords, do some analysis to determine if a pattern exists in the way passwords are being chosen, and build rules around that. If you recover four passwords, and looking at them you see that they are all one or more letters followed by a random six digit number, just knowing that information greatly reduces the key space you have to search. For example, this rule would state that the shortest password is seven characters long, but a search JUST of the seven-character space would only require 52 million attempts with this knowledge (52 options for the first character, and 10 options for each of the six remaining) versus 6.9 trillion (95 options per character for standard ASCII printable characters) without it, a reduction of 99.99 percent.

Here again, PRTK and DNA offer a similar capability to the tools readily and freely available in a dedicated Linux environment such as Kali Linux, and even offer some built-in rules. Both PRTK and DNA also offer a full featured rule editor that is substantially more user friendly than the syntaxes used by most open source tools. If you have these tools, keeping and maintaining a list of custom rules can greatly increase the efficiency of your key space searches.

ENHANCEMENT FOUR: DISTRIBUTED RECOVERY OPERATIONS

As Aaron Allston once famously said, “when all else fails, complicate matters.” Dan Kaminsky applied it to computers a little more succinctly when he said “sometimes, brute force *is* the elegant solution.” Either way, there comes a time when all the wordlist tailoring, tool selection, and approach optimization still won’t reduce the search time to a reasonable approach. When that happens, the only solution left is to literally throw more computing power at it.

A distributed recovery operation involves multiple systems acting in concert to parallelize the tasking at hand. If it would take one system eight hours to search a particular key space, then four systems working in parallel will be able to search that same space in two hours. There are many ways to accomplish this (and vendors willing to sell you their approach), but I’ll focus on just one of these with another common password recovery tool: John the Ripper (JtR).

JtR includes two features that make it relatively easy to manually parallelize operations: fork and node. The fork option allows JtR to create multiple processes to handle a single command. For instance, if JtR is told to process a wordlist that contains 10,000 entries and the fork command is specified with a value of “4” then four separate processes will be created, and each will be given a unique set of 2,500 entries from the total wordlist. The node option tells JtR how to split up tasking evenly between roughly equal systems, and pairs with the fork option. Each command executed by the node option specifies a number (or range of numbers) for the nodes to be interpreted by that command. For instance, on my two machines I specify the following to split up a common task (here, using JtR to crack MD5s between the two systems):

```
# On first machine
./john --wordlist=./star-trek.txt --rules --format=md5 --fork=8 --node=1-8/16 hashes.txt
# On second machine
./john --wordlist=./star-trek.txt --rules --format=md5 --fork=8 --node=9-16/16 hashes.txt
```

Of course, each system requires the wordlist (star-trek.txt) and list of MD5s to crack (hashes.txt). The result is that on each system there are eight instances of JtR executed, and each instance is given 1/16th of the wordlist as a seed. Since I am also using the JtR rules engine to generate permutations on my wordlist, the result is a distributed, hybrid attack across two machines. Not bad for two quick commands!

SUMMARY

In the end, it all comes down to determining how much time you are spending on password recovery operations, how much appetite you have for hardware, and what the costs/benefits are to improving the efficiency of password recovery. Improving the efficiency of operations can be as simple as a few hours' worth of effort, but once you start down the path you may quickly find yourself searching out new wordlists, dictionaries, rule set ideas, and tools. Password recovery is a deeply rewarding experience, both professionally in our investigations as well as personally in the hunt for that elusive bit of evidence that can provide concrete proof of circumstantial findings. Following the four steps laid out in this article will give you the basis, but it's up to you to take your work to the next level.

ABOUT THE AUTHOR

Brian Mork is an information security professional currently working for Alliance Data Systems, Inc., an S&P 500 company, is the leading global provider of data-driven marketing and loyalty solutions. Alliance Data and its combined businesses is a leading global provider of data-driven marketing and loyalty solutions serving large, consumer-based industries. His previous work experience includes serving in the United States military and as a Department of Defense contractor, specializing in information security and software development. He has been a speaker at multiple information security/hacker conferences, is the founder and technical lead of a Capture The Flag team which has won several competitions, and has authored a distributed password cracking system.

a d v e r t i s e m e n t

Big Data is Getting Bigger

Don't Get Buried.



Forensic Toolkit



Heat Map Visualization

Stay on top
of big data with FTK™

+1 800 574 5199

Fax: +1 801 765 4370

sales@accessdata.com

Forensic Toolkit™ (FTK) includes Visualization tools to help you reduce the big data analysis process by allowing you to visually understand relevant evidence, spot patterns and trends and determine areas where closer examination is required.

Recommended

Keep your PC at peak performance!

It incorporates Wise Registry Cleaner, Wise Disk Cleaner and many other useful features like System Slimming, Startup Manager, Disk Eraser, etc.

The software has amazing scan & clean speed, which makes it outstanding from similar products. You'll love the clean and user-friendly interface and the pragmatic efficiency!



WiseCleaner

Wise Care 365 2

- ✓ Over 15,000,000 downloads worldwide.
- ✓ Clean & Tune up Windows system.
- ✓ Protect digital privacy effectively.



5-Star Reviewed by
Top Download Websites

Official Website for More Information:
www.wisecleaner.com/wisecare365.html



Support system:
Windows XP, Vista, Win7/8
(both 32-bit and 64-bit)



Dr.WEB®

since 1992



Dr.Web 9.0

for Windows — the rapid response anti-virus

1. Reliable protection against the threats of tomorrow
2. Reliable protection against data loss
3. Secure communication, data transfer and Internet search



© Doctor Web
2003 — 2013

www.drweb.com

Free 30-day trial: <https://download.drweb.com>

New features in Dr.Web 9.0 for Windows: <http://products.drweb.com/9>

FREE bonus — Dr.Web Mobile Security:
<https://download.drweb.com/android>

