

Computer Forensics Applied to Windows NTFS Computers

Anders Svensson

Master thesis¹

Stockholm's University / Royal Institute of Technology
Kista, Stockholm, Sweden

April 2005

¹ This thesis corresponds to 20 weeks of full-time work

Abstract

The interest for computer forensic has increased the last couple of years. This is because criminals have moved to the digital world, using computers and computer networks to commit crimes. Computer forensics is a relative new science that is under rapid development and the need for computer forensic investigators will increase in the near future. This thesis has been written to give an introduction to the world of computer forensics and explain how to apply it to Windows computers.

This report is the result of an extensive literature review and should give the reader the needed information to begin analysing computers. The focus of the research study has been on four specific topics; computer forensic process, computer forensic techniques, NTFS file system and forensic science applied to Windows computers. The first two make up the theoretical framework for the thesis. The latter two has used the theoretical framework and developed it further by applying it to NTFS computers.

Computer forensics involve people from several different areas of expertise; legal advisors, technical experts, forensic specialists etc. Therefore all people involved should be aware of the forensic process, since they all may be of different opinions on how to conduct the investigation.

There are many sources of evidence on a NTFS computer that may give important information of what actions have been taken on the computer, what software applications have been installed and how the computer is configured. Recovering deleted files is central in that job, whether it is to gather information about specific actions taken or to recover a specific file of interest.

Computer forensic tools are of great help when locating and extracting evidence, but there is always a risk that a tool produces errors or that a tool has limitations making the output erroneous. The investigator therefore needs to have a deep understanding of the file system that is run on the computer.

Evidence collected is used in an attempt to reconstruct the crime so that the main questions (who, what, when, how, where and why) can be answered. Knowing where to look for evidence may give some extra clues that could be invaluable when trying to reconstruct the crime.

Sammanfattning

Intresset för datorutredningar har ökat de senaste åren. Detta beroende på att brottslingar har gått över till att utnyttja datorer och datornätverk i den digitala världen för att begå brott. Datorutredningar är en relativt ny vetenskap som är under snabb utveckling och behovet för nya utredare kommer att öka den närmaste tiden. Denna uppsats har skrivits för att ge en introduktion till datorutredningar och hur sådana utredningar kan appliceras på Windows datorer.

Denna rapport är resultatet av en omfattande litteratur granskning och syftar till att ge läsaren tillräckligt med information för att kunna påbörja att analysera datorer. Fokus för detta forskningsarbete har varit på fyra olika områden; arbetsgången för datorutredningar, tekniska metoder för datorutredningar, filsystemet NTFS och applicering av metoderna på Windows datorer. De två först nämnda områdena utgör det teoretiska ramverket för uppsatsen. De två sistnämnda områdena utvecklar teorin i det teoretiska ramverket ytterligare genom att applicera det på datorer med NTFS filsystem.

Datorutredningar involverar personer från olika expertisområden; rättsliga rådgivare, tekniska experter, utredningsspecialister osv. Alla personer som involveras bör därför vara medvetna om hur utredningen går till, eftersom det kan finnas meningsskiljaktigheter om hur en utredning bör gå till.

Det finns många källor med bevismaterial på en NTFS dator som kan ge viktiga upplysningar om vad som har gjorts på datorn, vilka program som finns installerade och hur datorn är konfigurerad. Återskapande av raderade filer är en central uppgift i detta arbete, oavsett om det gäller att inhämta information om vad som har utträttats på datorn eller det gäller att återskapa en specifik fil som är av intresse.

Datorutredningsverktyg är till stor hjälp för att hitta och extrahera bevis, men det finns alltid en risk att verktyget genererar felaktiga resultat eller att begränsningar i verktyget påverkar resultatet. Utredaren behöver därför en djup kunskap om det filsystem som datorn utnyttjar.

Bevismaterialet används sedan för att försöka rekonstruera brottet så att de grundläggande frågorna (vem, vad, när, hur, var och varför) kan besvaras. Genom kännedom om var man bör leta efter bevis kan eventuellt några extra pusselbitar hittas som kan vara ovärderliga i försöken att rekonstruera brottet.

Acknowledgements

This thesis has been conducted as the last part of the Master of Science program in Information and Communication System Security at the Royal Institute of Technology (KTH) in Kista.

I would like to recognise and thank Matei Ciobanu Morogan who has been my advisor at the university. Matei has helped and supported me throughout the writing of the thesis. His comments on the work has increased the quality of the thesis and also made it possible for me to complete it within the desired time frame.

Table of contents

1. INTRODUCTION	11
1.1. BACKGROUND	11
1.2. WHO SHOULD READ THIS THESIS?	11
1.3. RESEARCH PROBLEM	11
1.4. RESEARCH QUESTIONS	12
1.5. RESEARCH PURPOSE	12
1.6. EXPECTED RESULTS	12
1.7. RESEARCH METHOD	13
1.7.1. Literature review:	13
1.8. RESEARCH LIMITATIONS	13
1.9. RESEARCH VALIDITY AND RELIABILITY	14
2. THEORETICAL FRAMEWORK.....	15
2.1. HOW IS DIGITAL EVIDENCE PROCESSED?	15
2.1.1. Policy and Procedure Development	15
2.1.2. Evidence Assessment	15
2.1.3. Evidence Acquisition	16
2.1.4. Evidence Examination	16
2.1.5. Documenting and Reporting	17
2.2. FORENSIC ANALYSIS OF COMPUTER SYSTEMS	17
2.2.1. Forensic value and forensic quality	17
2.2.2. Computer disks and file systems	17
2.2.3. Data files and data areas	18
2.2.4. Live and dead systems	19
2.2.5. Boot process	19
2.2.6. Imaging hard disks	20
2.2.7. File recovery, data extraction and data reduction	21
2.2.8. Analysis of extracted data and crime reconstruction	23
3. APPLYING COMPUTER FORENSIC ON WINDOWS NTFS SYSTEMS	25
3.1. INTRODUCTION TO NTFS	25
3.1.1. NTFS disk structure	25
3.1.2. File names	29
3.1.3. Multiple data streams	30
3.1.4. File creation and deletion	30
3.1.5. Time and Date on NTFS systems	30
3.2. FORENSIC ANALYSIS OF NTFS SYSTEMS	32
3.2.1. NTFS and forensic investigations	32
3.2.2. Where is evidence found?	32
3.2.3. The importance of metadata files in computer forensics	32
3.2.4. File recovery	33
3.2.5. Recycle Bin	33
3.2.6. Internet Activity (Index.dat files)	35
3.2.7. Shortcut files (.lnk)	39
3.2.8. Thumbnail files (thumbs.db)	41
3.2.9. Registry entries	42
3.2.10. Printer spooler files	44
3.2.11. Additional locations of evidence	45
3.2.12. Future file systems and computer forensics	46
4. COMPUTER FORENSICS TOOLS	49
4.1. DEFINING COMPUTER FORENSIC EXAMINATION AND ANALYSIS TOOLS	49
4.1.1. Layers of abstraction	49
4.1.2. Categorizing computer forensics tools	51
4.2. COMMERCIAL COMPUTER FORENSICS TOOLS	52
4.2.1. Encase (Guidance Software)	52

4.2.2.	<i>Forensic Toolkit (AccessData Corporation)</i>	52
4.2.3.	<i>Winhex (X-Ways Software Technology)</i>	52
4.2.4.	<i>X-Ways Trace (X-Ways Software Technology)</i>	53
4.3.	OPEN SOURCE COMPUTER FORENSICS TOOLS.....	53
4.3.1.	<i>The Sleuth Kit (TSK)</i>	53
4.3.2.	<i>File Date Time Extractor</i>	53
5.	DANGERS OF MISUSE OF FORENSIC ANALYSIS TOOLS	55
5.1.	POTENTIAL MISUSE OF FORENSIC ANALYSIS TOOLS.....	55
5.1.1.	<i>Information theft</i>	55
5.1.2.	<i>Supervision</i>	55
5.2.	AVOIDING MISUSE OF FORENSIC TOOLS.....	56
6.	RESEARCH FINDINGS	59
6.1.	RESULTS	59
6.2.	LIMITATIONS	59
6.3.	RELIABILITY AND VALIDITY OF THE RESEARCH	60
6.4.	DISCUSSION	60
6.5.	FUTURE WORK	62
7.	REFERENCES	63
7.1.	BOOKS AND PER-REVIEWED ARTICLES	63
7.2.	WHITE PAPERS	63
7.3.	INTERNET SOURCES	64
7.4.	COMPUTER FORENSIC TOOLS	64

1. Introduction

1.1. Background

In the last decade there has been an enormous increase of computer usage. The development of digital equipment and the availability of computer networks have had a great impact on businesses today. A lot of transactions that earlier was done by regular mail is today conducted through automated processes on the Internet. This shift has made corporations dependent of computers and computer networks. In the past information was stored in large archives as paper documents. Today information is stored electronically in databases and often made available over networks. All of these changes have made a lot of the work easier for companies but the downside is that companies (and private persons) are more prone to attacks from cyberspace.

Criminals and the crimes they commit have always followed the development of new technologies closely; as soon as a new technology is developed the criminals adapt to it and use it to commit crimes. Technology advancements have had a positive influence on business opportunities but where businesses can make money there is a potential for criminals to make money as well. Criminals have now entered the digital world and more and more crimes are conducted with the use of computers or other digital equipment. Forensic investigators have therefore also been forced to enter the digital world and the need for computer forensic investigators has increased in the past years.

Companies have opened their eyes for computer forensic and many, often larger, corporations have started up their own computer forensic teams within the company. The market for computer forensic tools has expanded rapidly and several computer forensic tools have been developed and refined. There are a few software packages that have been widely used by computer forensic investigators, but the licence fees are quite expensive. Therefore a lot of open source tools have been developed that might be used by the public, free of charge.

1.2. Who should read this thesis?

This thesis is intended for security personnel, students or other people that are new to computer forensics. The reader should preferably have some knowledge of computers and Windows operating systems in particular.

1.3. Research problem

More and more literature has been written in the last couple of years. Even though a search on the Internet gives thousands of hits on the subject, it is hard to find a good reliable introduction to computer forensics. Most papers focus on specific problems that often are too complex to understand for people new to computer forensics.

There is also few “complete” research papers i.e. that takes up the forensic (investigative) work, forensics applied to computers and the technical background of the file system that is to be investigated. To conduct a sound computer forensic investigation the examiner needs to have good knowledge of both the computer forensic process and the underlying technology. Papers focused on the computer forensic process often prerequisites good knowledge of

operating system and the file system. Papers focused on forensics examinations of a particular file system often prerequisites good knowledge of the forensic process. There is a need for an introduction to computer forensics that takes up both of these areas.

Tools and techniques used in computer forensics are undergoing a rapid development. There are a few well accepted tools that successfully have been “proven in court”. These tools have undergone extensive testing to prove that they do what they are supposed to do. In addition to these few “accepted” tools there are several tools that have been developed by companies trying to enter the market and others developed in open source projects. Many of the tools are good, even though they may not be as complete as the market leading ones. The problem with most of the tools is, as with most software applications, the lack of a scientific approach to development by the development team [Car03]. Therefore there is a risk that errors are introduced that makes it hard for an investigator to draw unambiguous conclusions based on the results presented by the tool.

1.4. Research questions

As described above computer forensics is a rapidly evolving science and there are many problems that need further research. The problem area developed above will form and steer the focus of the thesis and the specific problems identified will hopefully be answered by this report. Four distinct questions have been formulated related to the problem area:

- What is computer forensics; what techniques are used to secure evidence on a computer?
- What information regarding NTFS is important for a forensic investigator to be aware of?
- How is computer forensics applied to Windows computers, and NTFS file system in particular?
- How are errors introduced by a tool and how does such errors affect computer forensic examinations?

1.5. Research purpose

There is a need for education in computer crime and computer forensic, not only for forensic specialists but for security personnel as well. Computer forensic training is offered by several companies, but the training is often intended for future forensic specialists.

The purpose is to give the reader an understanding of computer forensic tools and techniques applied to Windows computers. The reader should also get an understanding for the limitations of the tools and techniques presented. This master thesis shall hopefully give the reader the knowledge and skills needed to begin analysing Windows computers and to assist forensic teams during computer forensic investigations. That implies both to be aware of the forensic process as well as knowing where to look for evidence on a Windows computer.

1.6. Expected results

The result of this master thesis is to give a complete introduction to the computer forensics, and the thesis should encourage the reader to approach computer forensics in a scientific

manner. The four questions formulated above are the central questions that are to be answered by the thesis.

The report shall result in an introduction to the computer forensic process and describe how to apply computer forensic to NTFS computers. The theory will be presented and visually described using various computer forensic tools. Some of the tools used in the report will also be described briefly in a separate chapter covering computer forensic tools.

1.7. Research method

This thesis is a theoretical study of the subject and will involve technical details about computer systems and forensic science. An extensive literature review has been chosen as research method for the thesis.

1.7.1. Literature review:

In order to understand the underlying technology and how that knowledge could be used in computer forensics, an extensive literature review is to be done.

The literature review is focused on four different topics; computer forensic process, computer forensic techniques, NTFS file system and forensic science applied to Windows computers. The forensic process and forensic techniques shall form the theoretical framework for the thesis. In the analysing part of the research the theoretical framework is to be used and further developed by applying it to Windows computers utilizing NTFS file system. Understanding the underlying file system is important in computer forensics, but there is a lot of information that is not interesting from a forensic point of view. Therefore an introduction to NTFS will be given in the beginning of the analysing chapter.

Main sources for gathering information will be:

- Published papers from accepted and peer-reviewed journals.
- Printed books
- White papers.
- Internet pages/sites

Computer forensic is a relative new science and a lot of information on the Internet is subject to misinterpretations and is therefore often incorrect. Data gathered from Internet pages/sites will be thoroughly scrutinized and the primary use of the information will be to introduce new ideas that could be further researched.

1.8. Research limitations

The thesis will be limited to cover the NTFS file system, even though FAT is still widely used on Windows computers. FAT is an older file system and NTFS is the file system preferred today for computers running Windows 2000 or XP. FAT and NTFS has many similarities but there are too many factors that differs and that affect the tools and techniques used in computer forensic investigations. The workload of covering both file systems would be too extensive, and would force the discussion to be more general. Therefore only one file system will be covered and the decision was made to cover NTFS. There are two reasons for the choice; Windows 2000 and XP are the newest operating systems and NTFS the preferred file

system, and the amount of information about computer forensics and NTFS is less than for FAT file system (the need for more information is greater for NTFS).

The method that is to be used is an extensive literature review. One can argue that a deductive approach using experimental research could contribute with a lot of empirical data to analyse. One problem with the experimental research approach is that all computer forensic tools found for Windows operating system is only available in evaluation versions. Since this project is not financed by any company the licence fees was out of the budget. Using the limited evaluation versions would affect the research study and the results thereof negative and is therefore no choice. There are some Linux tools that are free of charge but using such tools prerequisites deep knowledge of Linux, which I don't have. Using those tools may introduce new problems that are out of the scope for the research study. Since a computer forensic examination is highly dependable of the examiner's knowledge of the underlying technology the reliability of the results would also be uncertain.

Another, even bigger, problem is that experimental research requires a lot of preparations and would take time from the literature review, and as previously mentioned that will make the discussion more general. Therefore I have decided not to use experimental research, and instead dig deeper into the subject using an extensive literature review.

When conducting a computer forensic investigation there are rules and regulations that have to be followed. Evidence presented in court must have been acquired in a forensically and legally accepted way, but it is hard to establish what that means. If investigations are conducted cross-boarder that is even harder, since laws are applied differently in different countries. The legal aspects of computer forensics are important and could/should be further studied as a topic of its own. The legal aspects will not be covered in this report.

1.9. Research validity and reliability

Literature and secondary sources of information are associated with two major problems; *Comparability* and *Reliability*.

Comparability problem has to do with that previous data may have been gathered for other purposes, and may not be comparable with the data collected for the current problem.

Reliability problems imply that the previous data may be incorrect, due to misinterpretations or flawed analytical reasoning. Methods for gathering the empirical data may also lack a scientific approach or mistakes may have been done

To counter comparability and reliability problems it is important to understand the reasoning behind any reports or articles. The majority of the data that is to be used in the research will be gathered from accepted and peer-reviewed journals or printed books on the subject. Researchers and authors of white papers, and other secondary sources will be especially checked upon.

2. Theoretical framework

Forensic analysis of crime scenes have gone from securing physical evidence such as fingerprints and DNA to secure digital evidence. Technology advances has resulted in more sophisticated crimes using computers and other digital equipment. Sometimes the computer or digital equipment is not used to commit the crime but digital evidence can be found that help investigators tie a suspect to the crime and the crime scene. Forensics investigators therefore need to secure digital evidence as well as physical evidence. Digital evidence can be found in many different devices like computers, scanners, printers, digital cameras, cell phones etc.

Chapter 2.1 will cover how digital evidence is processed. Chapter 2.2 describes how computers are searched for evidence and how the evidence found is to be analysed.

2.1. How is digital evidence processed?

Computer forensics involves experts from different areas of expertise which all have different approaches to examining a crime scene. Therefore a lot of aspects must be taken into consideration when conducting a computer forensic investigation. The process could, as described in [NIJ04], be divided into five steps; Policy and procedure development, Evidence assessment, Evidence acquisition, Evidence examination, and Documenting and reporting. These steps are discussed briefly below, for a more complete description see [NIJ04].

2.1.1. Policy and Procedure Development

Computer forensics units involve personnel with different skills and knowledge; forensic investigators, technical and legal experts etc. These different experts may have different opinions of how to conduct the investigation and therefore policy and procedures must be established. Administrative issues, funding, support from management and other important issues must also be dealt with. All these aspect are to be covered by the policy and procedure development that includes Mission statement, Personnel requirements, Administrative considerations, Service request and intake, Case management, Evidence handling and retention, Case processing, and Developing technical procedures [NIJ04]. Policy and procedure development is out of the scope of this report, for further information see [NIJ04].

2.1.2. Evidence Assessment

The second step is to assess the evidence and decide how to proceed with the investigation. There are a lot of different laws and regulations that may inflict on the investigation and therefore it could be wise to contact legal advisors to assist the forensic investigators when assessing evidence.

Before touching anything the crime scene may have to be searched for physical evidence such as fingerprints, DNA etc. Evidence can be in many different forms; printouts, notes on paper, current output on the computer screen, different types of digital equipment etc. Thus it is important not only to look for computer equipment since a lot of evidence would be missed. All sorts of evidence need to be assessed.

There may be evidence in several different locations, for example log files on a local server or mail servers at an Internet Service Provider (ISP) are often good sources of evidence.

Computers, external hard drives, digital cameras are others. Information that should be examined and documented are e-mail accounts, ISP used, aliases, network configuration, system logs, and passwords and so on.

The forensic investigator needs to decide whether the examination should be done onsite or if the evidence is to be transported and examined at a forensic laboratory. Factors such as time needed for the examination and business impact of confiscating computing equipment (servers, databases etc.) should be considered.

During the evidence assessment phase the forensic investigator need to determine how to document, package, transport and store the evidence so that it is not damaged. There might be a need for continuous electrical power for an electronic device, and some components must be protected for electromagnetic interference etc. For more information about packaging, transportation and storage of electronic evidence see [NIJ01].

2.1.3. Evidence Acquisition

The third step is to acquire the digital evidence in a manner that protects the evidence from being altered, damaged or destroyed. It is also important to document information regarding hardware configuration and internal components such as model, size, settings, drive interface etc.

The suspect's hard disks are imaged for later examination, using specialised computer forensic tools. The image of the hard disk and the configuration information are acquired through controlled boots. This will ensure that the suspect's operating system is not booted which would alter data and contaminate the evidence. Imaging and examination of the computer will be covered in greater detail in chapter 2.2.6.

2.1.4. Evidence Examination

Evidence examination is the process of extracting and analysing the digital evidence. Using the disk image created from the original source disk the data are recovered in two different ways; *physical* and *logical extraction*.

Physical extraction is not dependent of the file system; data is recovered from the binary data located on the physical disk. There are several ways to hide data and preventing it from being listed by a logical extraction. In a physical extraction; cluster by cluster on the disk is searched and the hidden data might be found and could help the investigator secure evidence of criminal activity.

Logical extraction is in contrast to physical extraction dependent of the file system on the computer. Logical extraction can reveal file names, file sizes, directory structure, time stamps and other useful information. In case of a highly fragmented disk files may be hard to recover using physical extraction methods. Logical extraction however may be able to recover fragmented files residing on the disk. During a logical extraction harvesting metadata information could also reveal important information that could be useful in an investigation.

When data extraction is complete the analysis phase begins. There are three major categories of evidence to look for [WP01]:

- Inculpatory evidence: that which supports a given theory.
- Exculpatory evidence: that which contradicts a given theory.
- Evidence of tampering: that shows that the system was tampered with to avoid identification.

There are a lot of conclusions that can be drawn from file names, time of last access, hidden and password-protected files, applications installed, Internet history and so on, see chapter 2.2.8 for more details.

2.1.5. Documenting and Reporting

Documenting and reporting is the final step although it is important to notice that documenting is an ongoing process throughout the examination. The documentation should be complete and accurate i.e. include every action taken during examination, all the findings, the name of the case investigator, and results etc. Documenting and reporting is out of the scope of this report, for further information see [NIJ01].

2.2. Forensic analysis of computer systems

Forensic analysis of computer systems is performed with specialised computer forensic tools, but in order to find and preserve the integrity of the evidence the investigator must be aware of how a computer and its file system works. This chapter will go through some of the basics of computer systems and forensic analysis.

2.2.1. Forensic value and forensic quality

Every piece of evidence has a *forensic value*, which describes the possibility to draw conclusions from the evidence [Buc04]. Time stamps has a high forensic value since time stamps makes it possible to reconstruct the order of actions on the computer.

Forensic quality refers to how believable the information is [Buc04]. Information like time stamps is considered to have a high forensic value, but it doesn't say anything about the forensic quality. If there is a possibility that the evidence has been tampered with, the forensic quality decreases.

2.2.2. Computer disks and file systems

The computer hard disk is used for storing non-volatile data such as program files, system files and user-created files etc. Non-volatile data refers to digital data that remain in memory even if the power is turned off. Volatile data on the contrary is lost when the computer is turned off or the power is lost. Volatile data is for example found in the computer RAM.

A hard disk may be divided into logically separate *partitions*, each formatted to a *volume* using a *file system* such as FAT or NTFS (Windows computers). Files are handled and stored differently on different file systems and this report is focused on NTFS used by Windows 2000 and XP operating system. A hard disk may contain up to four primary partitions which all can have different operating systems and different file systems.

A hard disk is divided into *sectors* which normally are 512-byte in size (determined by hardware). Two or more sectors form a *cluster* and thus the cluster size is always a multiple of the sector size. Cluster size varies between different file systems and on NTFS it is possible to

manually set the cluster size when formatting the volume. Larger clusters can make the disk blocks more manageable but with increased waste of disk space [Sol00].

A File stored on disk allocates as many clusters as needed to fit the entire file. Allocated clusters always belong to a certain file and cannot be split between two files. If for example a small file is saved to the disk but do not fill the entire cluster with data, the unused space of that cluster can not be used for storing any other data as long as file exists. This unused portion of the cluster is called *slack space* [Cas00]. The last cluster allocated by a file will always leave a little bit of slack space, since it is highly unlikely that the file fills the entire cluster. Therefore an increased cluster size will result in increased waste of disk space, because the average size of the slack space will be larger [Sol00].

Disk space not currently allocated by any file is called *unallocated space*. This does not mean that unallocated space is “empty space” on the hard disk. There is often a lot of information that could be found in unallocated space like deleted files or fragments of deleted files [Cas00].

2.2.3. Data files and data areas

During a computer forensic investigation the file system is searched for evidence. Evidence is often found in files, but there are other data areas that may contain evidence like slack space and unallocated space. One way to divide files and data areas is into the following four categories: user-created files, user-protected files, computer-created files and other data areas [NIJ01].

User-created files are files that the user is somewhat aware of; it could be files downloaded from the Internet and saved on disk or files created by the user himself. Files in this category are amongst others:

- Address books.
- E-mail files.
- Audio/video files.
- Image/graphics files.
- Calendars.
- Internet bookmarks/favorites.
- Database files.
- Spreadsheet files.
- Documents or text files

User-created files have high forensic value and are therefore important, and a lot of evidence may be found. If the suspect is engaged in illegal activity it is possible that he/she tries to protect the “illegal information” from being disclosed. This can be achieved by the use of encryption or by using steganography and there are other ways as well. These kinds of files are referred to as *User-protected files*. Some other examples of User-protected files are:

- Compressed files.
- Misnamed files.
- Encrypted files.
- Password-protected files.
- Hidden files.
- Steganography

Computer-created files are files that the computer system creates during normal operation, and are usually a rich source of information of what have been done on the computer. Some of the files are possible to delete and there are a lot of tools that could aid a criminal to erase tracks left on a computer. Even if such tools are used there are often some traces left and there

may also be possible to find evidence of that such tools have been used. Examples of computer-created files include:

- Backup files.
- Log files.
- Configuration files.
- Printer spool files.
- Cookies.
- Swap files.
- Hidden files.
- System files.
- History files.
- Temporary files

Other data areas refer to files and data areas not covered by the first three categories. The following are some examples:

- Bad clusters.
- Computer date, time, and password.
- Deleted files.
- Free space.
- Hidden partitions.
- Lost clusters.
- Metadata.
- Other partitions.
- Reserved areas.
- Slack space.
- Software registration information.
- System areas.
- Unallocated space

Note the difference between free space and unallocated space. Free space is disk space not allocated by any partition, unallocated space is disk space not allocated by any file on a partition.

2.2.4. Live and dead systems

When the investigator is to confiscate a live system there are some issues to consider before cutting the power. A live system refers to systems that are up and running where information may be altered as data is continuously processed. Dead systems are systems that are switched off and no data processing is taking place [Cas01]. To retain the integrity of the data it is often considered appropriate to cut the power supply to the computer, but this will have other implications.

There is a lot of information of evidentiary value that could be found in a live system. Switching it off may cause loss of volatile data such as running processes, network connections and mounted file systems. In contrast leaving a computer running may cause evidence to be altered or deleted. The investigator therefore needs to decide what alternative is best in a given situation. Another approach is to use specialised tools to extract volatile data from the computer before shutting it down, but this is out of the scope of this thesis.

Other aspects may also be needed to be taken into consideration. Confiscating servers and other computers may have a severe effect on the business that depends on the system, e.g. an online bank is highly dependent of their banking system. Often the business may have been the victim of a crime, and confiscating computers for a forensic examination may cause additional harm to the company. In some cases, to minimise the economical impact for the company, the forensic examination is better done on the live system [Cas01].

2.2.5. Boot process

As soon as the power switch on the computer is turned on, the computer starts the boot process which will load an operating system and make the computer ready to use. The start-up of a computer includes the following steps:

1. The BIOS and the CPU initiate the power-on self test (POST).
2. The BIOS finds the boot device
3. The BIOS loads the contents of the first physical sector of the hard disk into memory (the Master Boot Record, MBR).
4. The BIOS instructs the CPU to execute the MBR code.

The POST is run after a CPU reset (resetting or starting the computer) to test that system components work as expected. The BIOS looks for a boot device which normally is the hard disk but changing the BIOS configuration can make the computer to look for another device such as a CD. When the BIOS has located the boot device it loads the MBR into memory, supposed the device is a hard disk, and transfer the control to the CPU which executes the MBR code.

The MBR contains a partition table and this table is used to locate the primary partitions, which can be up to four in total, on a hard disk. Using extended partitions may allow additional partitions, but that is not covered in this report. During the execution of the MBR code the partition table is scanned for bootable partitions. If the BIOS finds more than one (in case of a so called multi-boot system) a menu is usually presented to let the user choose which to boot from. Normally there is only one bootable partition and when located the boot sector is loaded into memory. The boot sector is located at the first logical sector of a partition and contains executable code (the bootstrap code) and information that the file system needs to access the volume. The MBR code then transfers the control to the bootstrap code which allows the computer to boot the operating system.

If the BIOS finds a bootable CD or floppy disk the first sector (boot sector) from the CD or floppy is copied into memory and the control is transferred to the bootstrap code contained in the boot sector. The bootstrap code then loads the operating system.

By changing the order that the BIOS looks for boot devices, another operating system may be loaded even though there is one located on the hard disk. During a forensic examination of a computer; the forensic investigator often boots the computer with an operating system that does not write to the hard disk. This allows the investigator to make a copy of the source disk, called a disk image without changing anything on the original disk. The disk image is then used in the examination to locate evidence. The operation of disk imaging is further described in the next chapter.

2.2.6. Imaging hard disks

Imaging a hard disk means that information stored on the source disk is copied to another storage area. The source disk could be cloned onto another disk or copied into a file. A hard disk image is a bitstream copy of the entire disk and since cluster by cluster is copied from the disk all data including information in slack space, unallocated space and free space will be copied to the image file. Volatile data, such as information in RAM, will not be copied though. It is important that the storage medium used for the image has been wiped of all previous data. Overwriting all data once with a known pattern is generally sufficient [Cas04]. By using a clean storage medium the risk for mistaking previous data as coming from the source disk is minimized. Before making the image a cryptographic hash value is calculated of the original disk, using MD5 or SHA1 algorithms. The cryptographic hash value could be used to verify that the forensic image created is identical to the source disk.

When examining a computer the information on the hard disk could be retrieved in two ways; an image of the disk is created or only the data that are of interest is copied. The latter alternative will only copy the files, which means that metadata files, slack space and unallocated space will not be copied. This could be enough for the purpose of the investigation but in some cases an image of the disk(s) is preferred.

The following steps are the general procedure when collecting the entire contents of a computer [Cas00]:

1. all volatile data should be collected
2. the computer should be shut down
3. the computer should be booted using another operating system that bypasses existing one and does not change data on the hard disk(s)
4. a copy of the digital evidence from the hard disk(s) should be made

When creating an image of the hard disk it is important, as the third step above explains, not to let the computer boot its own operating system. When a computer runs its bootstrap code some information is altered and this is not accepted when conducting a forensic analysis of a suspect's computer. Therefore the forensic investigator needs to bypass the original operating system through a series of controlled boots to ensure that the computer loads the preferred operating system from floppy disks or CD-ROMs. During the controlled boots all configuration information should be documented.

The first step is to unplug the hard drive from the motherboard to prevent alteration of data if the controlled boot fails. The next step is to verify the booting sequence so that the preferred operating system will be booted. The boot sequence may need to be altered and that implies changing the BIOS configuration information [NIJ04].

A second controlled boot is done to ensure that the forensic operating system is loaded from the floppy/CD-ROM. It is important that the hard disk still is disconnected in case the boot process is not successful [NIJ04].

If the operating system boots as expected a third controlled boot is performed, this time with the hard disk reconnected. The disk configuration information (CMOS/BIOS) should be captured and documented. Disk configuration information includes logical block addressing (LBA); large disk; cylinders, heads, and sectors (CHS); or auto-detect [NIJ04].

There are several imaging tools used for capturing the data from hard disks to create an image. All of these tools must use some kind of write protection to prevent altering data [NIJ04]. Some tools use hardware write protection and some use software write protection. Which one is best is out of the scope of this report, but the important thing is that write protection is used so that data on the source disk is not altered.

The process of imaging a hard disk could be done on the confiscated computer or on the examiner's system. If possible it is better to do it on the examiners system, because the examiner will have better control of what is done to the disk.

2.2.7. File recovery, data extraction and data reduction

It is important to notice that the extraction of data is not limited to recovery of deleted files or recovery of data from slack space. There is a lot of information on the computer that is readily

available, like temporary files, e-mails etc. That will be further covered in later chapters, but this chapter will focus on different recovery techniques.

The recovery processes differs depending on the file system used and the recovery process of NTFS file systems are further described in chapter 3.2.4. Following are some important techniques used to recover and extract data from a disk image.

2.2.7.1 Harvesting

Harvesting is done to gather metadata relating to directories, files, and fragments salvaged during the recovery process [Cas04]. Recovered metadata could be used to recover deleted files, NTFS stores for example clusters allocated by a file in the Master file table and even if the file has been deleted, the information may still exist in those clusters. This will be further covered in chapter 1.

In case the suspect has used a file scrubbing application file recovery may not be successful, but it is possible that metadata information can be found and witness of the file's existence. Such information may be as useful as the file itself.

Metadata can also be found within files [Cas04]; Microsoft Office files have for example time stamps and some other metadata embedded in them. Embedded metadata should therefore also be harvested.

2.2.7.2 Recovery of slack space and deleted files in unallocated space

Deleted files can give indications of criminal activity, and therefore it is important to locate and recover deleted files. When a file is deleted and the reference pointing to the file is not successfully recovered another approach could be used. Provided the cluster(s) has not yet been overwritten by new data the files could be recovered using *file carving* (physical extraction).

Common file types like graphic image files (jpg, bmp) and text files (txt, doc) all use different known headers. File carving takes advantage of this fact and scans through the disk for these known patterns. When a specific file header is found the data is carved out from the following clusters [Cas04]. In some forensic applications file carving is only checking for the known headers at cluster boundaries where files normally start. This is important to be aware of because some files could be embedded in other files and only searching at cluster boundaries would not list such files [Cas04].

If a cluster containing a deleted file is overwritten there could still be information of evidentiary value in slack space because this unused part will contain data from the previous file. That information could be valuable even if the entire file is not recovered.

2.2.7.3 Keyword searches

Keyword searches include searches for text or hexadecimal values on the disk. Files including specific words may be of interest and could be found through keyword searching. Searching for keywords are often performed on the binary data of the disk but there is a risk of not finding all occurrences of the keyword. A physical search is usually done in continuous clusters. If a file containing the keyword is fragmented and the keyword is split on two non-continuous clusters the occurrence of the keyword would not be listed. Combining physical and logical keyword searches might give a better result [Alt04].

2.2.7.4 Extraction of encrypted and compressed data

Files that are encrypted or compressed are not readable and could therefore go unnoticed when trying to recover deleted files, searching for specific keywords etc. Therefore the disk should be scanned for encrypted and compressed files. When identified the files should be decompressed and decrypted if possible. Decrypting files could be a difficult task depending on the algorithm used, but there is a lot of software that could aid in that process.

2.2.7.5 Data reduction

There is a lot of data that are of no interest for the investigation and therefore these files (here referred to as irrelevant files) could be removed. Reducing the number of files to be reviewed will save time and ease the work for the forensic examiner.

Hash values could be calculated to identify and locate known irrelevant files on the disk. When the irrelevant files are identified the examiner can exclude them. Duplicate files could also be reduced with the use of hash values, but some additional factors should be considered before removing them though. Duplicate files are often found when several backups are stored on the same disk. However files could have been relocated and renamed between two backups and such information may be valuable. Removing duplicate files solely based on hash values might destroy valuable information. Therefore file names and file paths should be matched as well [Cas01].

In some cases only certain file types are of interest and all other are considered irrelevant for the investigation. Files could therefore be filtered based on file type and all irrelevant files can be excluded [Cas01]. The filtering process should not only look at the file extension, because it does not guarantee that a file is of a certain type. Instead the files should be filtered out based on other file characteristics such as the file internal header.

2.2.8. Analysis of extracted data and crime reconstruction

When performing a forensic investigation of a computer the investigator needs to find evidence that could be used to reconstruct events and actions that took place on the computer. Digital evidence can in addition to identify the object and its source be used to sequence events, determine locations and paths, establish time of the actions etc. Such information is important for the investigator to be able to draw unambiguous conclusions. The basic questions that the investigator needs answer when trying to reconstruct the crime is: who, what, when, how, where and why.

Collected evidence that are used to reconstruct the crime fall into three categories: relational, functional and temporal [Cas00].

Relational evidence describes how objects interact and where they are in relation to the crime or to each other. Relational reconstruction could give information about geographical locations of suspects, victims, computers and the interactions that have been taken place etc. Relational evidence also includes locations of files and folders on a computer, location of hidden and missing data. Missing data refers to data that is stored in other storage medium outside the computer, or to previous files or programmes that have been deleted intentional or unintentional.

Functional reconstruction is performed to understand how a particular application or system works. This information could be used to determine what has happened based on the

knowledge of how the application responds to a specific event. It is important that the configuration of the system/application at the time of the crime also is determined.

Temporal reconstruction is conducted to sort events and actions in the order that they have been taken place. This is often done with the help of time stamps indicating last modification, last access, and creation time. Some applications also save log files that include time stamps for actions taken and it is important not only to look for evidence on the local computer (as explained in chapter 2.1.2).

In [NIJ04] the analysis of extracted data has been divided into some other categories; Timeframe analysis; Data hiding analysis; Application and file analysis and Ownership and possession. These categorizes are closely related to the ones mentioned above, but the functional reconstruction is somewhat missing. The timeframe analysis correlates to temporal reconstruction. Combining the Data hiding analysis and the Application and file analysis correlates to the relational reconstruction; Ownership and possession is a combination of both relational and temporal reconstruction. There is no analysis of the functional aspects of systems and applications used.

Why is this important? As explained in chapter 2.2.1 the forensic value is defined as the possibility to draw conclusions from collected evidence and the forensic quality refers to how believable the evidence is. The investigator needs to be certain that the evidence collected is of both high forensic quality and has a high forensic value. This implies that the investigator must have knowledge of how the evidence was created. For example if a temporal reconstruction results in a list of actions and their time references; the investigator has to interpret the time information accurately because different systems and applications handle time information differently. Some systems/applications make adjustments to time zones and daylight savings and some do not. Whether adjustments have been made or not obviously have a severe effect on a temporal reconstruction of the crime.

3. Applying Computer Forensic on Windows NTFS systems

As Casey explains in [Cas01], forensic analysis of computer systems presumes a strong understanding of the underlying file systems. There are several tools that could aid in a forensic process, but as with most software there are limitations of what the tool can do. It is important to understand those limitations and how they could affect the results presented by the tool, read more in chapter 4.1. The forensic examiner must also know how to interpret the results in order to get an understanding of what has really happened. An understanding of the file systems might also help the investigator utilize the forensic tool more effectively.

Chapter 3.1 will cover the basics of NTFS and chapter 3.2 will describe how to examine NTFS computers for forensic evidence.

3.1. Introduction to NTFS

There are two major file systems used on Windows computers today, FAT 32 and NTFS. NTFS is the preferred file system for computers running Windows 2000 or Windows XP operating systems. NTFS is therefore a widely used file system on Windows systems today. Because of that, this report will focus on NTFS and if nothing else is said NTFS is the file system discussed.

In NTFS everything is treated as a file; bootstrap program, configuration data, application data, program executables and the operating system itself are all comprised of files [Sol00]. The following sections will cover the basics of NTFS, but is limited to topics that are relevant for computer forensic investigations.

3.1.1. NTFS disk structure

3.1.1.1 Volumes and file systems

A volume is a logical partition of the disk. There may be several volumes on one physical disk and each volume is represented by a drive letter and a colon (e.g. C: and D:) in Windows systems. Each volume is formatted with its own file system like FAT and NTFS on Windows systems or EXT2 and EXT3 in UNIX/Linux systems. Newer versions of Windows operating systems (Windows 2000/XP) use NTFS even if it is possible to use FAT formatted disks. NTFS offers higher security and more flexibility than the previous FAT file system.

3.1.1.2 Cluster

As explained in chapter 2.2.2 a computer hard disk is divided into sectors and the file systems bundles one or more sectors together to form a cluster. A simple rule is that the larger the disk the larger the cluster size (number of sectors in each cluster). Cluster size may be changed when formatting the disk. Table 1 shows the default cluster sizes for different disk sizes on NTFS formatted disks:

Volume Size	Default Cluster Size
512 MB or less	512 bytes
513 MB-1024 MB	1 KB
1025 MB-2048 MB	2 KB
Greater than 2048 MB	4 KB

Table 1 Default cluster sizes on NTFS formatted disks

On a NTFS formatted disk all clusters have a *Logical Cluster Number (LCN)*. LCNs are the sequential order of the clusters from the beginning of the volume to the end. LCN 0 (zero) refers to the first cluster in the volume (the boot sector). NTFS converts the LCN to a physical disk address (byte offset of the volume where the cluster resides) by multiplying the LCN with the cluster size [WP03].

Clusters belonging to the same file are also given a *Virtual Cluster Number (VCN)*. VCNs are the internal order of the clusters in a file and do not need to be physically contiguous on the disk.

3.1.1.3 Master File Table

Every file on a NTFS volume is represented by at least one record in the Master File Table (MFT). The MFT is an array of records and is continuously changed as files and folders are copied, renamed, deleted etc. The first 16 records in the MFT are reserved for NTFS specific metadata files [Sol00]. These metadata files are described in Table 2.

Name\MFT	Record	Description
\$MFT	0	NTFS's Master File Table. Contains one base file record for each file and folder on an NTFS volume.
\$MFTMIRR	1	A partial copy of the MFT. Serves as a backup to the MFT in case of a single-sector failure.
\$LOGFILE	2	Transactional logging file
\$VOLUME	3	Contains volume serial number, creation time, and dirty flag
\$ATTRDEF	4	Attribute definitions
.	5	Root directory of the disk
\$BITMAP	6	Contains bitmap of all used/unused clusters on the volume
\$BOOT	7	Boot record of the drive
\$BADCLUS	8	Lists bad clusters on the drive
\$SECURE	9	Contains unique security descriptors for all files within a volume.
\$UPCASE	10	Maps lowercase characters to uppercase
\$EXTEND	11	Used for optional extensions such as quotas, reparse point data, and object identifiers.
	12–15	Reserved for future use.

Table 2 NTFS reserved file records in the Master File Table

Some of these NTFS metadata files are important in order to understand the discussion in the following chapters and therefore some additional comments are given below.

The first record in the MFT is the file record for the master file table itself, followed by a record for the partial copy of the MFT (MFT mirror, \$MftMirr) located in the middle of the disk. MFT mirror acts as a backup for important metadata files in case the MFT is corrupted [Sol00].

The bitmap file, \$Bitmap, is used to identify which clusters are in use (allocated by a file) and which are not [Sol00]. Every cluster on a volume is represented by a bit in the bitmap file, and if the cluster is in use the corresponding bit has the value one. The bitmap attribute is also found in folder records, where it is used to keep track of which clusters in the “index allocations buffers” are in use and which are not, as described in 3.1.1.6.

When a cluster contains a bad sector, the operating system marks the entire cluster as bad and stops using that cluster to save new data. \$Badclus list clusters that NTFS has identified as bad [Sol00].

3.1.1.4 MFT records

MFT records include a series of attributes (and their values) and have a size limit of 1 KB; the cluster size of the volume does not affect the size of the MFT records. Attributes are divided into two logical components: a header and the data. The header stores the attribute's type and identifies the location (byte offset from the header) and length of the attribute's value, see Figure 1 [Sol00].

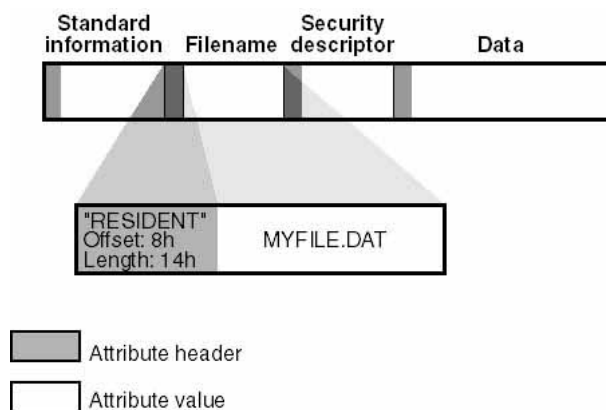


Figure 1 MFT record attributes, figure from [Sol00]

There are several attributes defined for NTFS, as seen in Table 3. Because of the size limit all attributes may not fit in a single file record, and will be stored in one or more clusters on a different location of the disk. Attributes residing in the MFT file record are called *resident attributes*, and attributes whose values are stored elsewhere are called *non-resident attributes* [Sol00]. There are some attributes that always are resident like the standard information, and the file name attribute. The data attribute, which includes the file data is however often a non-resident attribute.

Attributes that grows too large in size and do not fit in the file record are stored in one or more *runs*. A run is defined as an allocated space of one or more continuous clusters. NTFS keeps track of the runs by using LCNs (to locate the starting cluster) and the number of continuous cluster at that position of the disk. NTFS also make use of the files VCNs to map the internal order of the runs [Sol00].

If the number of attributes grows too large to fit in the file record, NTFS creates a second MFT record for the additional attributes. An Attribute List attribute is used to describe the location of the additional attributes. An attribute list is also created if the number of runs for a file is too large, for example when a file becomes highly fragmented [Sol00].

Attribute Type	Description
\$VOLUME_INFORMATION	Volume versions and label information.
\$VOLUME_NAME	
\$FILE_NAME	File or directory name. A file can have multiple filename attributes
\$STANDARD_INFORMATION	File attributes such as read-only, archive, time stamps, including when the file was created or last modified.
\$SECURITY_DESCRIPTOR	Previous versions of NTFS stored private security descriptor information with each file and directory (present for backward compatibility).
\$DATA	File data
\$INDEX_ROOT, \$INDEX_ALLOCATION, \$BITMAP	Implement filename allocation and bitmap indexes for large directories (directories only).
\$OBJECT_ID	Used by the distributed link tracking service.
\$REPARSE_POINT	Stores a file's reparse point data.
\$ATTRIBUTE_LIST	Lists the location of all attribute records that do not fit in the MFT record.
\$EA_INFORMATION, \$EA	OS/2-compatibility extended attributes
\$LOGGED_UTILITY_STREAM	EFS stores data in this attribute that's used to manage a file's encryption.

Table 3 NTFS Master File Table attributes

3.1.1.5 File records

Every file record consists of at least the following attributes/value pairs; standard information, filename and the unnamed data attribute. The unnamed data attribute contains the file's "raw" data or the run-information [Sol00].

If all the attribute/value pairs fit in the MFT file record, no other associated data resides on the disk. This means that a small file only exists in that MFT record, but a large file has data stored in clusters (data runs) elsewhere on the disk. Every file may be associated with multiple runs and the run-information is listed in the data attribute of the MFT file record, see Figure 2.

The run-information of a highly fragmented file may be too extensive to fit in the data attribute of the file record. In such a case an attribute list is used to point out the data attribute headers of additional MFT record(s) containing the remaining run-information, as shown in Figure 2.

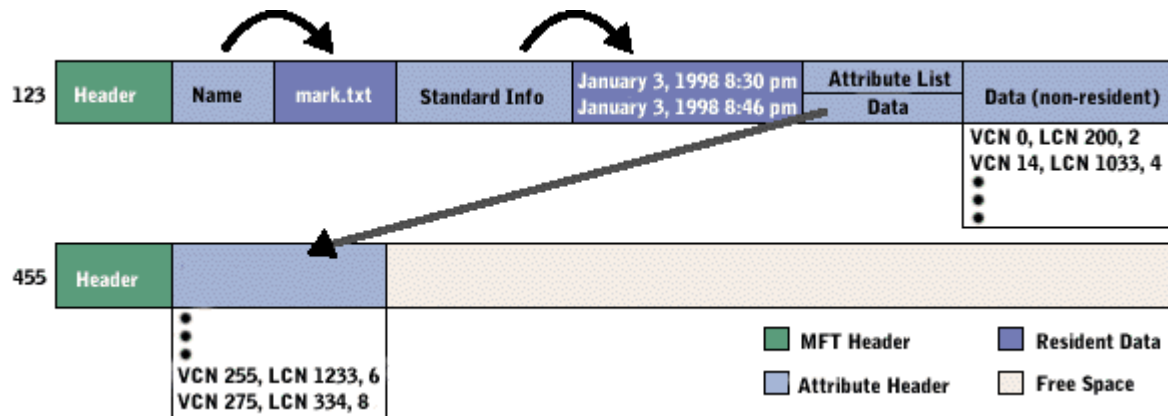


Figure 2 NTFS Master File Table file record, figure from [WP03]

3.1.1.6 Folder records

Folder records include an index root attribute which points out the entries (index entries) for each file and subfolder, shown in Figure 3. Index entries consist of the filename and a copy of the standard information. As the number of files in the folder grows the space may not be enough to fit all index entries, and the additional index entries are stored in index allocation buffers. The index allocation attribute contains the run-information to point out where to find the index buffers. The bitmap attribute is also used to indicate which VCNs in the index allocation buffers are in use [Sol00].

Index entries are created in the file's parent folder record or index allocation buffers (if the folder contains a lot of files and subfolders). When a file is deleted the corresponding index entry is marked for deletion.

Directory's MFT Record

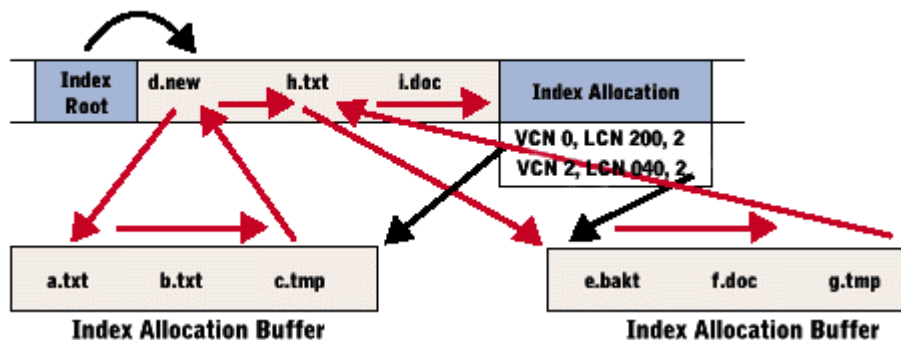


Figure 3 NTFS Master File Table folder record attributes, figure from [WP03]

3.1.2. File names

File names in NTFS are stored using Unicode-based characters [Sol00]. File names in NTFS can be as long as 255 characters and use embedded spaces, multiple periods and special characters that are not allowed in MS-DOS file names. NTFS also stores a short file name within the same MFT record as an additional file name attribute. The short file name corresponds to MS-DOS 8.3 file name where the first eight characters represent the file name chosen by the user, and the last three characters represent the file extension. NTFS automatically generates the short file name by deleting all spaces, all special characters and additional periods. If there are several files with similar file names, NTFS uses the first six characters and appends a '~' and a number. The MS-DOS name can be used just like the long

file name but the short file name makes it possible to open the file from systems that do not support long file names.

The use of long file names gives the user an opportunity to include a “description” of the file content in the file name. Thus file names can be used as evidence of, for example, that a user has knowledge about the content of a particular file.

3.1.3. Multiple data streams

Unlike FAT, NTFS support multiple data streams. As explained earlier a file consists of, at least, a standard information attribute, a filename attribute and the unnamed data attribute. The unnamed data attribute is the *primary data stream*, or often referred to as the *default data stream* [Sol00]. In addition to the default data stream, each file may be associated with several named data attributes, called alternate data streams (ADS). ADS may include resources such as icons, file signatures, passwords, executable program code etc. Windows Explorer uses ADS to store the summary information listed in the summary tab of the dialog box that appear when right-clicking a file [Sol00].

3.1.4. File creation and deletion

3.1.4.1 File creation

When a file is created on a NTFS disk the bitmap file in the MFT must be updated. The clusters that the file now occupies that previously was marked as “unallocated” is now changed to “allocated” in order to reflect the current status of the disk. A file record in the MFT is created where non-resident data, if any, is pointed out by the data-runs in the data attribute. An index entry is also created to point out where to find the file in the logical structure. The index entry is created in the folder record in which the file is located (the file’s parent folder) [Cas01].

3.1.4.2 File deletion

Just like when a file is created, the Bitmap file has to be changed when a file is deleted to reflect the status of the disk. The bits corresponding to the clusters previously allocated by the file is changed to zero (marked as unallocated). The file record is marked for deletion, but will remain on the disk until it is overwritten. The file’s index entry is also marked for deletion, but index entries are quickly overwritten [Cas01].

3.1.5. Time and Date on NTFS systems

As explained earlier temporal reconstruction is an important step in a forensic investigation of a computer system. Times and dates can reveal a lot of what has happened when and in what order, supposed the time stamps are interpreted correctly. In order to understand the time stamps and how they are affected by the local settings in the computer, a brief explanation of how NTFS stores time information is given below.

There are some widely known time references; Greenwich Mean Time (GMT), Universal Time (UT) and Cordinated Universal Time (UTC) of which UTC is the most commonly used in the IT community [Boy04]. Time stamps in the MFT on a NTFS system are stored in UTC, but time is often displayed in local time. In the system’s registry information there are registry keys that are used by the operating system to make adjustments for local time differences and daylight savings. For example, Windows Explorer displays the local time for files and not the

UTC. Another example is the computer clock that is visible in the lower right corner which is automatically adjusted when changing from summer to winter time and vice versa.

In NTFS the UTC time is stored as a 64 bit number, which is equivalent to the number of 100 ns intervals since 00:00:00 on 1 January 1601 (i.e. 64 bit Windows FILETIME time format) [Boy04]. In other file systems time is calculated using other time format, but that is out of the scope for this report.

3.2. Forensic analysis of NTFS systems

The following chapters discuss forensic analysis of NTFS systems, and where evidence could be found. There will be some case scenarios presented, where evidence found on NTFS computers have been used in court.

3.2.1. NTFS and forensic investigations

During an examination of a computer system the computer forensic investigator tries to find evidence that can answer the following questions: who, what, when, how, where and why? But is it possible to answer these questions, and what degree of forensic value and what forensic quality can be obtained?

NTFS, like most other file systems, was not designed with computer forensics in mind but there is a lot of information on the computer that could be used in an investigation.

It is possible to find evidence of computer usage because a lot of the actions taken by the user leave traces on the computer. Creating, deleting, renaming, copying, modifying and accessing files will all cause changes to metadata files in the MFT. Executing programs will leave the same kind of traces, since a program is treated as a file like everything else. Printing documents will also leave traces, since the document is cached before it is printed. Examining files may therefore give an understanding of what programs were executed, what files were accessed and modified and so on. Of course the forensic quality will be contested in court, and it may be hard to prove forensic value of the evidence.

3.2.2. Where is evidence found?

There is often a tendency to argue how to recover deleted files and how to interpret file fragments when discussing evidence acquisition. But there is a lot of information that does not only exist in unallocated space or slack space and is therefore easily recovered. Such files are for example files located in the Recycle Bin, Shortcut files and so on. These files are rich sources of evidence and should be examined thoroughly. The following chapters will explain where to look for evidence on a NTFS computer, but remember that it is not an exhaustive list.

Extraction and analysing “normal” files (user-created files like e-mails, documents, spreadsheet files etc.) that have not been deleted is a relative straight forward process and will therefore not be covered in the following chapters.

3.2.3. The importance of metadata files in computer forensics

Metadata contains a lot of information about files and are therefore a useful source of evidence in a computer forensic investigation. Examining metadata files may give evidence of user activities, the computer’s current and previous configuration and so on.

Often when a suspect becomes aware that he/she is under investigation, he/she might try to eliminate all traces by deleting files that could be used as evidence. Locating and recovering metadata files might be enough to track user activities even if recovering of the actual data files is unsuccessful.

3.2.4. File recovery

MFT file records belonging to deleted files may be possible to harvest, because file records are not permanently deleted and will remain on disk until they are overwritten by new file records [Cas01]. The chance to successfully recover deleted (marked for deletion) file records decreases with time, since NTFS overwrites deleted file records before allocating additional space for the MFT. The file record contains the standard information (MAC times amongst other things) and the filename. Such information could be very useful in an investigation. If the file record is recovered the data-runs for the file's non-resident data will also be known and could easily be recovered. Without the file record a physical search of the disk could still locate and recover the deleted file, supposed it is not fragmented. Fragmented files are very hard to recover completely through a physical search but even if only parts of the file are recovered it may include important evidence. File carving (physical extraction) is explained in chapter 2.2.7.2.

The index entry belonging to the deleted file may not be possible to recover since it is quickly overwritten. The index entry could be important in the relational reconstruction of the crime. When an index entry is marked for deletion all the following entries are moved up and therefore overwriting the marked entry. If the entry is the last entry in the folder record or in the index allocation buffer, it will not be overwritten and can therefore be recovered [Cas01].

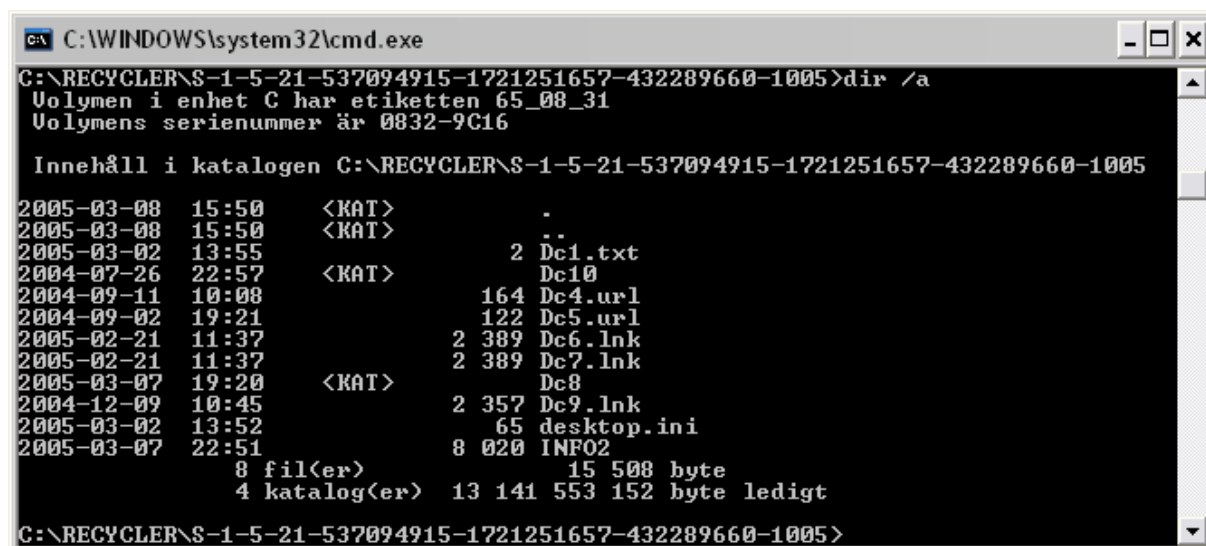
As explained above the possibility of recovering metadata files decreases with time since it is likely that they have been overwritten [Cas01]. Metadata belonging to recently deleted files however may successfully be recovered (as when a criminal delete files in panic when the police are knocking on the door).

3.2.5. Recycle Bin

Usually when a file is deleted on a Windows computer, the file is moved to the Recycle Bin. That means that it is possible for a user to retrieve a file that has been deleted by mistake, supposed the Recycle Bin has not been emptied [Cas01]. The user may however turn the Recycle Bin feature off or holding down the SHIFT key while pressing the DEL key, and in those cases the file will not be sent to the Recycle Bin. The file will still exist on the computer though and can be recovered, see the previous chapter. The Recycle Bin is found in the hidden system folder RECYCLER on NTFS computers. On FAT32 systems (Windows 95/98/ME) the Recycle Bin folder is named RECYCLED.

On file deletion a copy of the file is moved to the Recycle Bin and the file is renamed to *DC<index>.<extension>*. The first file moved to the Recycle Bin gets the index number '1' and the second number 2 and so on, see Figure 4. After the Recycle Bin has been emptied and the system rebooted the index numbering starts over.

NTFS stores information needed to recover the file in a hidden system file named INFO2, also seen in Figure 4. The INFO2 file includes the file's original name and path, file size, date and time of deletion (date and time of creation of the INFO2 record) and its index in the Recycle Bin. Each file record appended to the INFO2 file is 800 bytes in size and the INFO2 grows larger as more files are deleted and moved to the Recycle Bin [Cas01].



```

C:\WINDOWS\system32\cmd.exe
C:\RECYCLER\S-1-5-21-537094915-1721251657-432289660-1005>dir /a
Volymen i enhet C har etiketten 65_08_31
Volymens serienummer är 0832-9C16

Innehåll i katalogen C:\RECYCLER\S-1-5-21-537094915-1721251657-432289660-1005

2005-03-08 15:50 <KAT>      .
2005-03-08 15:50 <KAT>      ..
2005-03-02 13:55          2 Dc1.txt
2004-07-26 22:57 <KAT>      Dc10
2004-09-11 10:08        164 Dc4.url
2004-09-02 19:21        122 Dc5.url
2005-02-21 11:37         2 389 Dc6.lnk
2005-02-21 11:37         2 389 Dc7.lnk
2005-03-07 19:20 <KAT>      Dc8
2004-12-09 10:45         2 357 Dc9.lnk
2005-03-02 13:52         65 desktop.ini
2005-03-07 22:51         8 020 INFO2
                8 fil(er)          15 508 byte
                4 katalog(er)    13 141 553 152 byte ledigt

C:\RECYCLER\S-1-5-21-537094915-1721251657-432289660-1005>

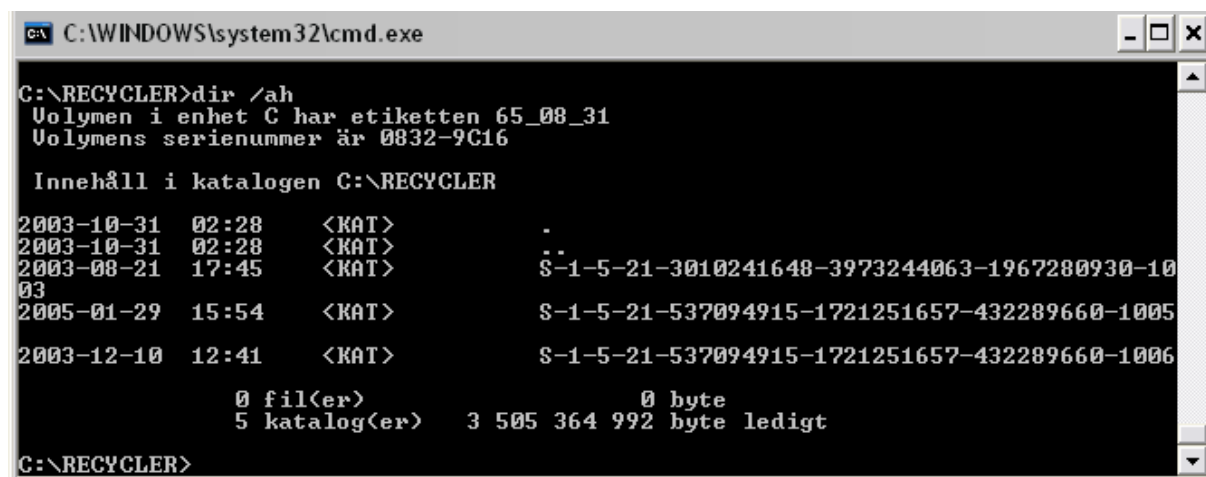
```

Figure 4 Recycle Bin on a NTFS system

There is an INFO2 file created for each user and the file is found in a user specific folder created the first time he or she deleted a file. The INFO2 file is found in the following location on Windows 2000 and XP computers:

C:\RECYCLER\<USER SID>\INFO2

The user specific folder (named by the User SID) makes it possible to map deleted files to specific users. Figure 5 shows the RECYCLER folder on a computer where 3 users have their own Recycle bin. All 3 users has got an own folder that contains an INFO2 file and a copy of the files deleted by the user.



```

C:\WINDOWS\system32\cmd.exe
C:\RECYCLER>dir /ah
Volymen i enhet C har etiketten 65_08_31
Volymens serienummer är 0832-9C16

Innehåll i katalogen C:\RECYCLER

2003-10-31 02:28 <KAT>      .
2003-10-31 02:28 <KAT>      ..
2003-08-21 17:45 <KAT>      S-1-5-21-3010241648-3973244063-1967280930-1003
2005-01-29 15:54 <KAT>      S-1-5-21-537094915-1721251657-432289660-1005
2003-12-10 12:41 <KAT>      S-1-5-21-537094915-1721251657-432289660-1006
                0 fil(er)          0 byte
                5 katalog(er)    3 505 364 992 byte ledigt

C:\RECYCLER>

```

Figure 5 NTFS RECYCLER folder

Files found in the Recycle Bin have been deleted by the user and not by the system, because files deleted by the operating system are not moved to the Recycle Bin. When the Recycle Bin is emptied the INFO2 records are deleted, but there might be possible to recover them. The copies of the files are also deleted from the Recycle Bin folder, but as with the INFO2 records, they might be recovered, see more in chapter 3.2.4 about file recovery.

3.2.6. Internet Activity (Index.dat files)

There are several web browsers used on the Internet, but the following chapter will discuss Internet Explorer that comes with Windows operating systems. Internet Explorer is widely used on Windows computers and is one of the most used web browsers on the Internet.

Internet is a wide client – server network and when a URL is typed in the browser the client asks the server for the page. To allow for quicker access to sites already visited; Internet Explorer caches the content of the visited web page including the time of visit, the address, images, cookies etc. When revisiting a site Internet Explorer checks the Web site server for changes to the page [WP02]. If there are any changes to the page, the new version is retrieved. If not the cached page is used. Web pages can therefore be opened from the local hard disk instead of downloading the page again.

Websites often place small text files on a user's computer to save information about web sessions, these files are called cookies. Cookies are useful in many different situations; cookies can be used to automatically sign a user into a specific site, it could hold information about a user's purchases in a web shop and so on.

Information about the cached files, cookies and where to find them on the hard disk is stored in hidden system files named index.dat. The location of index.dat files depends on the version of Windows and if user profiles are used on the computer. On Windows 2000/XP computers the Index.dat files are normally found on the following locations (note that the drive letter could vary):

1. *C:\Documents and Settings\<Username>\Local Settings\Temporary Internet Files\Content.IE5\index.dat*
2. *C:\Documents and Settings\<Username>\Local Settings\History\History.IE5\index.dat*
3. *C:\Documents and Settings\<Username>\Local Settings\History\History.IE5\MSHistXXXXXXXXXX\index.dat*
4. *C:\Documents and Settings\<Username>\UserData\index.dat*
5. *C:\Documents and Settings\<Username>\Cookies\index.dat*

The third location is actually one of several folders named slightly different. The X's corresponds to the included file's last access dates and each folder contains an index.dat file.

Windows users are often given the advice to now and then empty the Temporary Internet Files folder to free space on the computer. What happens is that all content, i.e. html-pages, images etc. is deleted (the files may still be recovered from unallocated space), but an index.dat file will remain. An Index.dat file is a system file and is not easily deleted manually since it is always used by the operating system [WP02].

A lot of internet pages state that the index.dat file contains all the URLs that a user ever visited and can be examined with specialised tools [Inet03], [Inet04]. This might be true for index.dat files created by UrlCache versions earlier than 5.2 (in [Inet03] the UrlCache version is 4.7). This has not been verified in this report, since earlier versions have not been available during the testing. In Internet Explorer 5 and 6 UrlCache version 5.2 is used. In that version the index.dat file is overwritten by a common pattern and therefore the information cannot be retrieved. The size of the index.dat file is not changed though, but all information is overwritten and Internet Explorer starts to fill the file with new URLs.

There are a lot of tools available that are supposed to erase all traces of Internet activity, but that promise is not always true. Because some of the tools just delete the index.dat file without overwriting it, the information could still be retrieved if the index.dat file is successfully recovered from unallocated space. A lot of the tools are effective though and the use of such tools has been increasingly popular (number of downloads of such application have increased).

The Index.dat files found in any of the first four locations described above are stored in binary format, which makes them impossible to read without specialised tools. The cookies are stored in ASCII format and can be read without a tool, even though specialised tools makes it easier to read the files [WP02]. There are both open source and commercial tools available that could be used to review the content of the index.dat files.

The index.dat files found in the first three locations includes web site content such as html-files, images, scripts etc. The fourth index.dat file holds UserData records about the files used to store information locally on the computer, much like cookies. Since they are rarely used the index.dat file in the UserData folder will not be further discussed. The fifth and last location relates to cookie files stored on the computer. The content of the index.dat files found in the first three and the fifth location will be explained below.

3.2.6.1 Index.dat (Temporary Internet Files folder)

Index.dat file includes a lot of records and there are four different types: HASH, URL, LEAK and REDR. The last three types refer to the index.dat activity records, which is the actual Internet activity that has been taken place. The first type (HASH) is used as a lookup table to find the activity records within the index.dat file. As described in a research study of the Index.dat file [WP02], the activity records have the following format:

TYPE, LENGTH, DATA

The length of the activity records are one or more 128-byte blocks and the DATA field contains the information depending on the type of the activity record.

The LEAK records are similar to the URL record, described below, and the REDR record is created when the browser is redirected to another site. REDR records contain the URL to which the browser was redirected.

Information that is found in an URL record is the last modification date and time (at offset 0x08), last access date and time (at offset 0x10), local cache folder (where the cached file is found), URL to the web site visited, the filename of the cached file, the full HTTP header of the response from the Web server and the user account used when visiting the Web site, see Figure 6. The date and time information is stored in 64-bit FILETIME format and is shaded in the figure. The local cache folder is found at offset 0x38 (which value in this case is 0x03, or 3 in decimal) and is a reference to n:th folder listed in the beginning of the index.dat file, see Figure 8. [WP02]

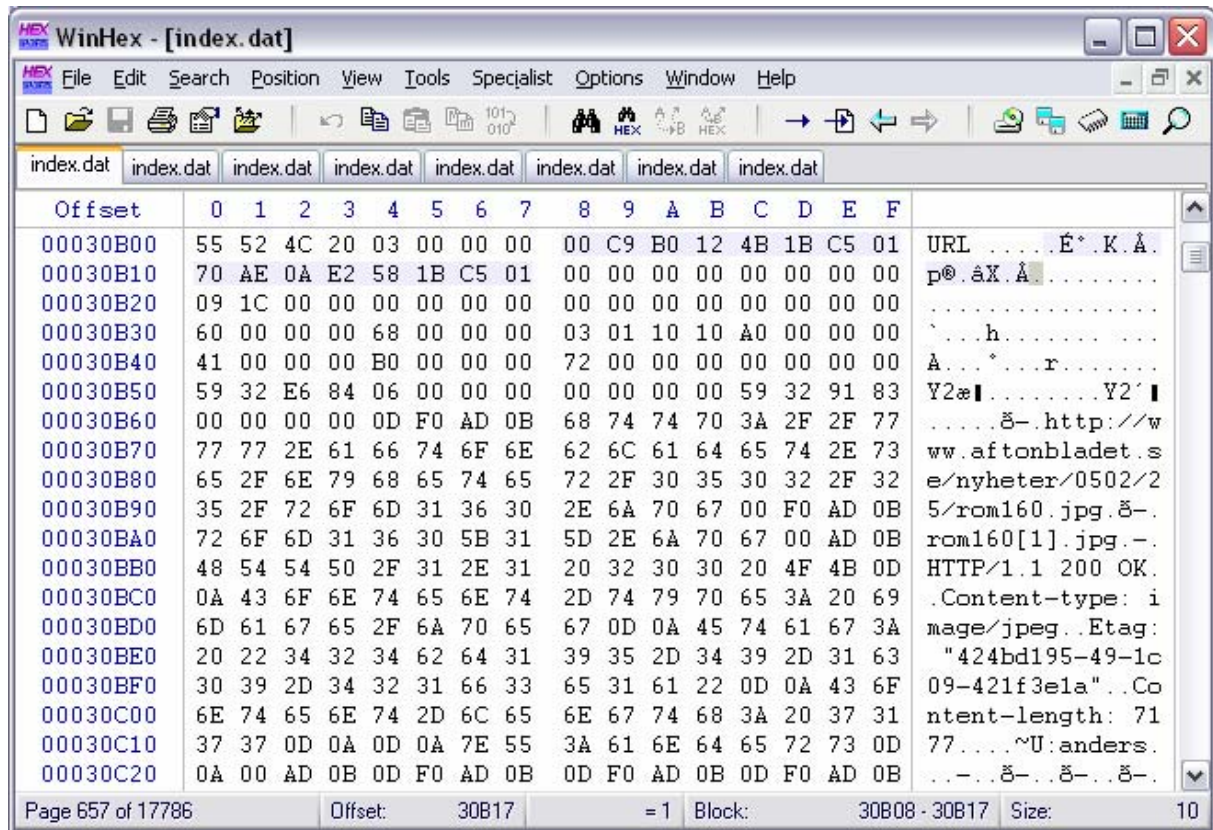


Figure 6 Index.dat file displayed using X-Ways evaluation version of WinHex

Last modification time refers to the time that the information was last modified on the server from which the information was downloaded. Last access time refers to the time when Internet Explorer was used to access the web page.

Even if the information in an index.dat file is somewhat readable using a hex editor (as in Figure 6) an index.dat viewer is the preferred choice during examination of Internet activity. Figure 7 shows how the information is displayed in one such index.dat viewer.

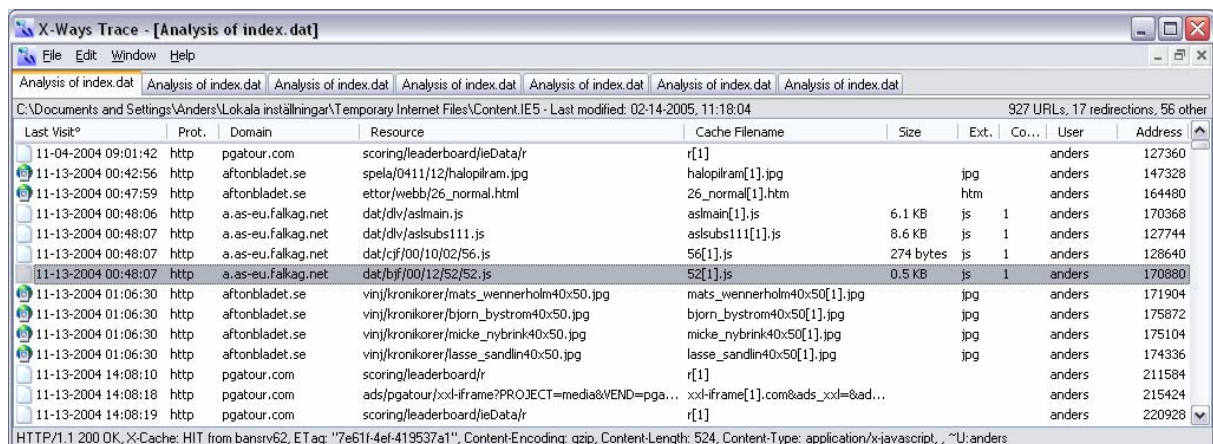


Figure 7 Index.dat file displayed using an evaluation version of X-Ways Trace

Figure 8 shows the beginning of the index.dat file and as can be seen the UrlCache version is 5.2 and all the subfolders where the cached files are to be found are listed (in this example there are 12 subfolders). The number of subfolders may vary depending on the size of the index.dat file.

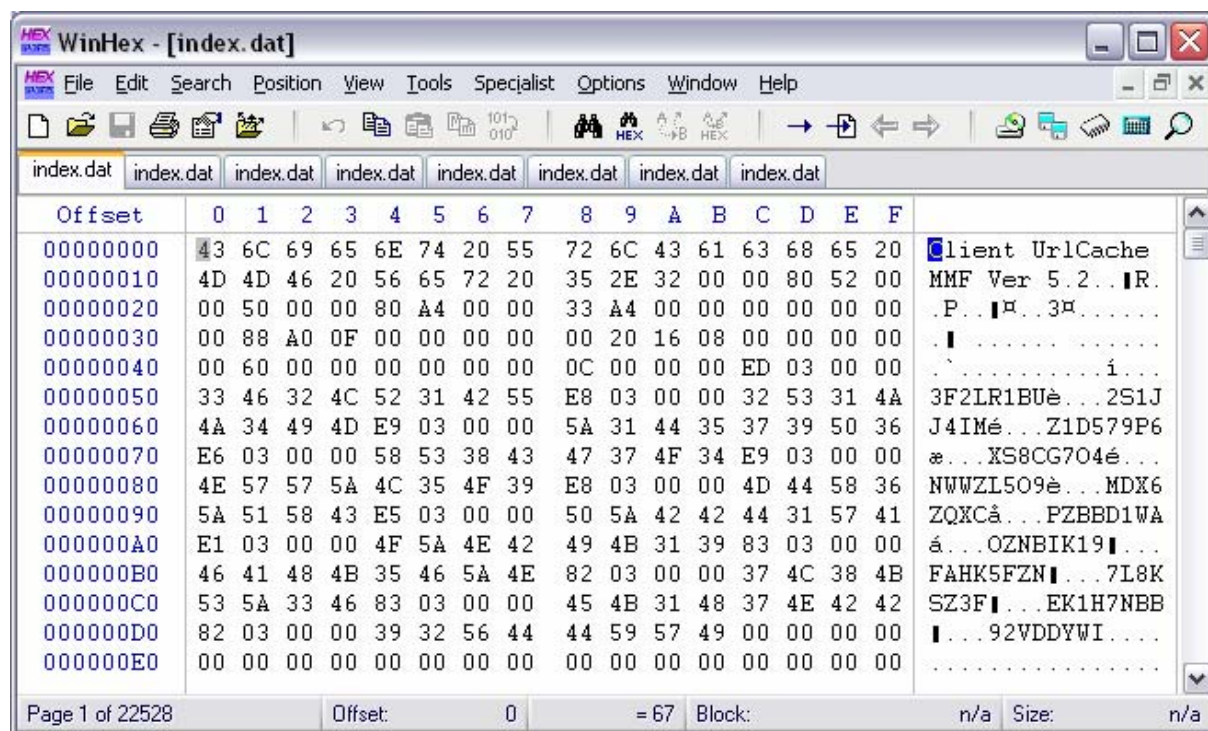


Figure 8 Cache folder entries in NTFS index.dat file

3.2.6.2 Index.dat (History folders)

Index.dat files in the history folders are used by Internet Explorer's auto complete function. An overview of the Internet history can be displayed in Internet Explorer (press CTRL + H). In the history.IE5 folder (in this discussion called the "upper history folder") there is one index.dat file and several subfolders. These subfolders are named using the date/dates of the Internet activity it contains, see Figure 9.

As can be seen in Figure 9 there is one new folder created each day containing an index.dat file. After the week is complete the daily index.dat files will be written to a weekly folder that contains one index.dat file with all information from the previous daily index.dat files. Depending on how many days IE has been configured to keep history the number of folders in the upper history folder will vary.

The URL records of the index.dat file in the history folders differs from the ones found in the Temporary Internet Folder's index.dat file. A lot of the information is excluded, such as server response and information about the cached files. This is because the history folders don't have anything to do with caching of web content. Notice that the time stamps in the index.dat files are not correct in the weekly history folders. The timestamp is changed when the daily index.dat files are written to the weekly folder. The last access time now corresponds to when the information was written to the weekly history folder and not when the actual Internet activity took place.

```

C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Anders\Lokala inställningar\Tidigare\History.IE5>dir
a
Volymen i enhet C har etiketten 65_08_31
Volymens serienummer är 0832-9C16

Innehåll i katalogen C:\Documents and Settings\Anders\Lokala inställningar\Tidigare\History.IE5

2005-03-10  10:16    <KAT>          -
2005-03-10  10:16    <KAT>          ..
2003-09-15  13:17                113 desktop.ini
2005-03-10  10:33             2 375 680 index.dat
2004-04-25  03:19    <KAT>          MSHist012004042520040426
2005-02-21  00:02    <KAT>          MSHist012005021420050221
2005-02-28  10:25    <KAT>          MSHist012005022120050228
2005-03-07  09:39    <KAT>          MSHist012005022820050307
2005-03-07  09:39    <KAT>          MSHist012005030720050308
2005-03-08  09:12    <KAT>          MSHist012005030820050309
2005-03-09  08:20    <KAT>          MSHist012005030920050310
2005-03-10  10:16    <KAT>          MSHist012005031020050311
                2 fil(er)                2 375 793 byte
                10 katalog(er)       13 151 973 376 byte ledigt

C:\Documents and Settings\Anders\Lokala inställningar\Tidigare\History.IE5>

```

Figure 9 History folders

3.2.6.3 Index.dat (Cookies)

The index.dat file includes the URL from where the cookie was received, the date and time, and the name of the cached cookie file. The cookie files are stored within the same folder.

3.2.6.4 Case scenario, Internet cache files: False report

The following case scenario describes how Internet cache files were used to create evidence in a real case. The case scenario is from [Cas01].

“In another recent case, detectives investigated a woman’s complaint that she was the victim of stalking by a former boyfriend. The woman claimed that the former boyfriend was sending threatening e-mail to her current boyfriend. During the investigation, she made another report alleging that she had been the victim of a home invasion during which she was assaulted, and she again identified the suspect as the same ex-boyfriend. When the detectives examined the woman’s computer, they found that the temporary Internet cache files contained references to an America Online account. Further examination of the Internet cache files and the records of America Online showed that the woman had set up an account with a screen name similar to that of the former boyfriend, and had sent the ‘threatening’ e-mail messages herself.”

3.2.7. Shortcut files (.lnk)

Shortcut files (or link files) points to a target file or application. Shortcuts are used to quickly access or execute a file or a program, without having to locate the target on the system. Shortcuts are also used to access folders or devices such as printers and scanners, and they can therefore tell a lot of the computer’s current and previous configuration, file accesses, devices etc [Cas01]. Shortcuts are often found on the Windows Desktop or in the Windows Start Menu, but there are several other locations that hold shortcut files, as will be further developed.

A file's file record in the MFT includes the MAC value for the file (included in the standard information attribute). In addition to its own MAC value, the data attribute of a shortcut file's MFT record includes the MAC, name and full path of the target file. In case the target file still exist and is located on the same computer the target file will have its own MFT record where the same information will be found. If the file has been deleted or it resides on a removable media there is no MFT record for the target file, and thus the shortcut will give valuable information, such as the target file's current or previous location and name [Cas01]. The time stamp is also important information that can be used in the temporal reconstruction of the crime.

Figure 10 shows the time stamp (MAC) found in a shortcut file (MAC time belonging to the target file). Date and time information is stored as 64-bit values and the target file's creation time is stored at offset 28 (0x1C), last access at offset 36 (0x24) and last modification at offset 44 (0x2C).

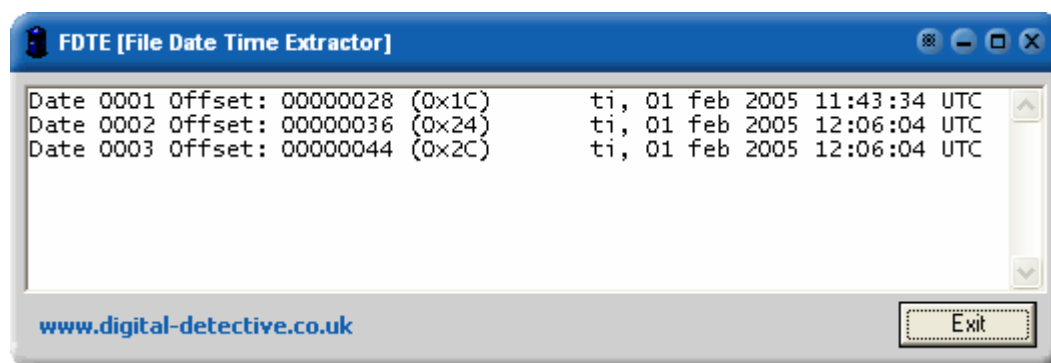


Figure 10 MAC information of a shortcut file

Every time a file is accessed a shortcut is added to the folder RECENT which is a hidden system folder that exists for every user account on the computer. A shortcut found in the RECENT folder therefore gives an indication that the user must have had knowledge about the file's existence, since the file has been accessed [Cas01]. The time stamp in the shortcut file can be used in a temporal reconstruction of events. Shortcuts in the RECENT folder remain even if the target file is deleted. There may also be shortcuts relating to files located on additional hard disks or removable media. An examination of the qualified full path and name of the target file can indicate that files residing on storage medium, that currently is not connected, have been accessed.

During an installation of an application the user is often given a choice whether to create a shortcut on the Desktop and in the Start Menu or not, but sometimes a shortcut is added without the user's knowledge. The shortcut can give an indication that a user is aware of specific files or application but the forensic quality of such conclusions may be contested. It is also possible to verify if the shortcut has been created during the installation or after. In case the shortcut was created during installation of the program the creation time of the shortcut should match the date and time of the installation. If the shortcut has been created at a later date or time the suspicion of that the user had knowledge about the target file would be strengthened [Cas01].

Another location where shortcuts can be found is in the Send To folder. The Send To folder contains shortcuts that are created when a user sends a file to a destination like an e-mail programme or to the “My Documents” folder [Cas01].

An examination of available shortcuts on the computer can, as explained, reveal a lot about the current or previous configuration and be of great help during relational reconstruction of a suspect’s computer. Time stamps included in shortcut files will also help the investigator with the temporal reconstruction of the events and actions on the computer.

3.2.7.1 Case scenario: Link Files Link Evidence

The following case scenario describes how link files were used to create evidence in a real case. The case scenario is from [Inet01].

“When a group of employees left one company to join a competitor, the former employer believed the employees had misappropriated its customer list. The former employees denied taking the list. The company hired CFI to examine the computers used by the former employees to see if there was evidence to the contrary. When the entire customer list was found on the former employees’ old computers, along with evidence that the customer list had been copied onto removable media, CFI examined the computers the employees were using at their new place of employment. The employees’ current computers contained a link file that indicated the customer list had been copied from removable media. CFI found the entire client list in the unallocated space on one of the hard drives. This evidence allowed the former employer to obtain an injunction prohibiting the former employees from contacting any of the names on the list on behalf of their new employer.”

3.2.8. Thumbnail files (thumbs.db)

Windows creates thumbnails for graphic image files (JPG, GIF, PNG, and BMP) which are used when listing files as miniatures in Windows Explorer. Thumbs.db may contain thumbnails for graphic images that have been deleted [Cas01]. Other information that can be found in thumbs.db files is the original filename and last modified date. On Windows 2000, the full path of the original image file can also be retrieved. On Windows XP the full path can not be found, only the file name of the original image file can be collected.

Several forensic tools today have support for analysing thumbs.db files, and the information found in those files can give evidence of deleted graphic files and indicate that the user had knowledge about the files. Two examples of tools that have support for viewing thumbs.db files are Forensic ToolKit (FTK), and Encase. Figure 11 displays the content of a thumbs.db file, where only two of the graphic image files still exist on the disk. As can be seen in the figure, the thumbnails included in thumbs.db allows the investigator examine the graphic files even though they have been deleted. A lot of additional information is also available.

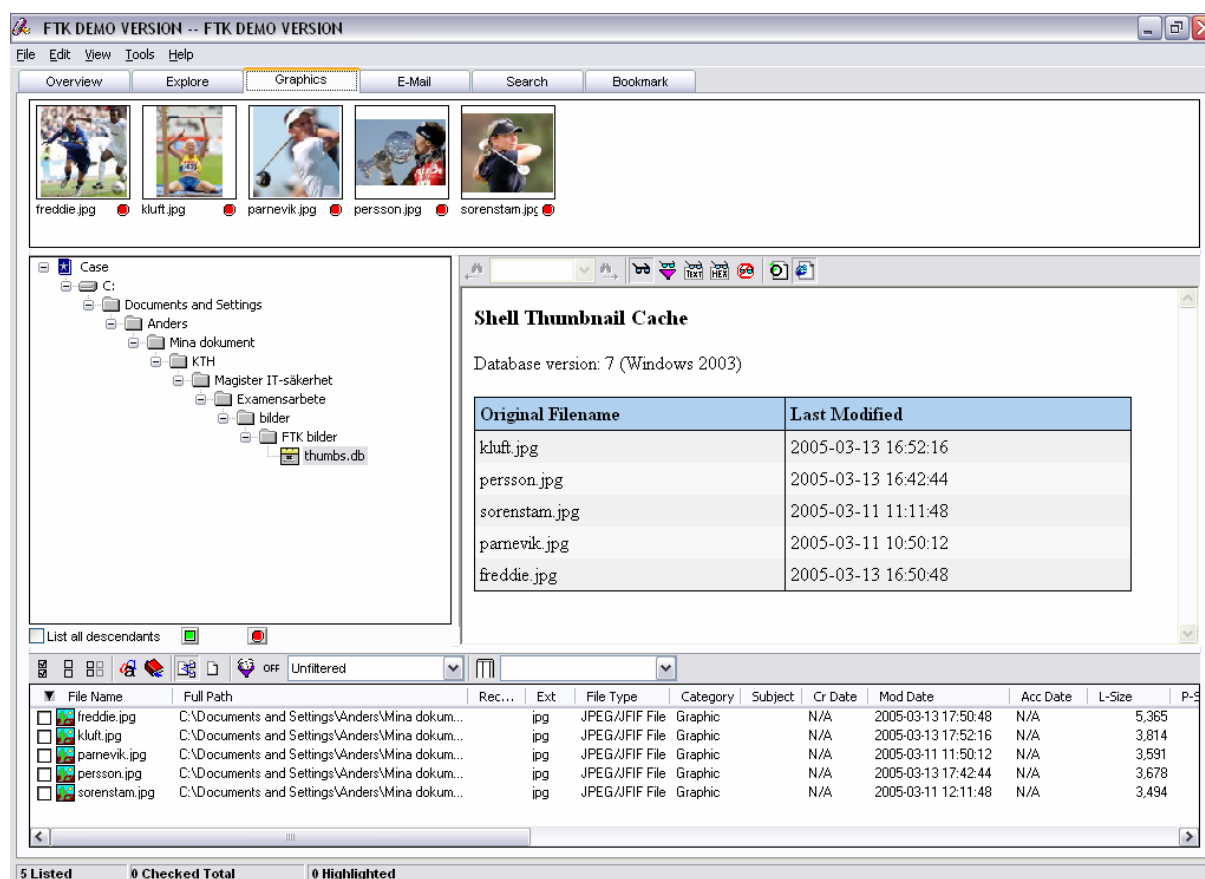


Figure 11 Viewing thumbs.db in a Demo version of Forensic Toolkit

The thumbs.db file is a hidden system file which is by default not visible in Windows Explorer. Therefore many users are not aware of their existence and thumbs.db files are often found intact, i.e. they have not been overwritten or deleted with the intention to cover up tracks.

3.2.8.1 Case scenario: Thumbnails

The following case scenario describes how a thumbs.db file was used to provide evidence in a case. The case scenario is from [Cas01].

“In a recent federal criminal investigation, the examiner located a folder containing more than 400 evidentiary images. When the examiner questioned the nature of the thumbs.db file, further analysis showed its function and contents. The file was found to contain more than 900 images, many representing files of evidentiary value that had been deleted from the folder.”

3.2.9. Registry entries

The registry on Windows computers are a rich source of evidence, and it contains information about settings for installed hardware and software. It also contains the user specific settings and preferences on the computer, thus changes made on the computer, for example in the control panel or to installed software, is reflected in the registry entries [Inet06].

The Registry has a hierarchal tree structure and there are five main branches (subtrees), see Table 4. The subtrees contain keys, sub-keys and entries. Keys and sub-keys do not contain

any data; instead they make up the hierarchal structure of the Registry. Entries consist of an entry name, a data type and a value. The value contains the actual data associated with the key or the sub-key. Values stored in the Registry are several types of values of which five are more common; REG_BINARY, REG_DWORD, REG_EXPAND_SZ, REG_MULTI_SZ, REG_SZ. Each key (and sub-key) may have none, one or several sub-keys and values associated with it [Sol00].

Subtree	Description
HKEY_CURRENT_USER (HKCU)	Contains the environment variables, personal program groups, desktop settings, network connections, printers, and application preferences for the current user.
HKEY_USERS (HKU)	Defines the default user configuration on the local computer and the user configuration for the current user.
HKEY_LOCAL_MACHINE (HKLM)	Contains information about the local computer system, including hardware and operating system data, such as bus type, system memory, device drivers etc. It also contains information about installed software, current configuration data, including Plug and Play information, network security information and other system information.
HKEY_CLASSES_ROOT (HKCR)	Contains two types of data: <ul style="list-style-type: none"> • Data that associates file types (file extensions) with applications. • Configuration data for COM objects, Visual Basic programs, or other automation.
HKEY_CURRENT_CONFIG (HKCC)	Stores configuration data for the current hardware profile.

Table 4 Registry Subtrees

The HKCU, HKCR, and HKCC subtrees do not contain any data but provides easier access to the data. The three subtrees contain pointers to the data that is found in HKLM and HKU.

The Registry Hive is the set of keys, sub-keys and values that makes up the Registry. The data is stored in their respective supporting files that are loaded when Windows starts [Sol00]. Table 5 lists the Registry hives and their supporting files. The supporting files for all Registry hives, but HKEY_CURRENT_USER, are found in the following location:

C:\Windows\System32\Config

The supporting files for HKEY_CURRENT_USER are found in:

C:\Document and Settings\<user name>\

Registry hive	Supporting files
HKEY_CURRENT_CONFIG	System, System.alt, System.log, System.sav
HKEY_CURRENT_USER	Ntuser.dat, Ntuser.dat.log
HKEY_LOCAL_MACHINE\SAM	Sam, Sam.log, Sam.sav
HKEY_LOCAL_MACHINE\Security	Security, Security.log, Security.sav
HKEY_LOCAL_MACHINE\Software	Software, Software.log, Software.sav
HKEY_LOCAL_MACHINE\System	System, System.alt, System.log, System.sav
HKEY_USERS\DEFAULT	Default, Default.log, Default.sav

Table 5 Windows Registry hive

Many applications do not clean up its own registry entries when the application is uninstalled, and thus a lot of entries are still left after the uninstall process. This results in that the registry

will grow and contain information about many applications that previously was installed on the computer.

Locating and recovering the supporting files of the Registry hives can give a lot of information about currently or previously installed hardware and software. It will also give evidence of user profile configuration and so on. For example it might be possible to tell if a suspect have used a file scrubbing application to cover up tracks of illegal activity, by examining the Registry [Cas01].

3.2.9.1 Case scenario: Registry entries

The following case scenario describes a case where the examination of the Registry was used to collect evidence. The case scenario is from [Cas01].

“In a recent investigation by The Los Angeles County Sheriff’s Computer Crime Unit, a detective investigated an employee suspected of misappropriating confidential computer information stored by his company. When the detective examined one of the workplace computers, he found remnants of a key-trapping program in the registry. During an interview, the suspect admitted to having installed, used, and deleted the key-trapping program for the purpose of obtaining user names and passwords of co-workers.”

3.2.10. Printer spooler files

Printing jobs are done in the background and make use of temporary files created by the spooling process. The content of the printing job is written to a spool (.spl) file and information such as user name, document name, and data type (RAW or EMF) is written to a shadow (.shd) file. The data type found in the shadow file determines if the spool file is a RAW or Enhanced Metafile (EMF) file. EMFs are used by the default Windows NT print spooler, and EMF files are encoded to provide printer independence. If the spool file is in RAW format the spooled data is formatted for a particular printer, thus RAW spool files are device-dependent [Cas01].

The .spl and .shd files have the same file name, usually a number such as 00002.SPL and 00002.SHD. By default, both files are written to the following location:

C:\WINDOWS\SYSTEM32\SPOOL\PRINTERS

When the printing job is done both files are deleted automatically, but as with all files that have been stored on a hard disk it might be possible to recover the files from unallocated space.

Recovered spooler files indicate that the user had knowledge about the files existence since the files have been printed out. SPL and SHD files contains information about the print job, such as the name of the file printed, the owner of the file, the printer used, and the data to be printed [Cas01]. The printed files may have been deleted and overwritten or might not ever have been saved to the local hard disk. By recovering spool and shadow files that are created by Windows during the printing process, printed files could be viewed even if the actual data files are not successfully recovered.

An easy way to view a printed file is to print out the recovered .spl file using the following DOS-command (in case the data type is RAW the file must be printed out using the printer type that is found in the .shd file):

C:\...|>copy XXXXX.SPL lpt1

The X's corresponds to the file name of the recovered spool file. This command also implies that the printer is connected to the parallel port on the computer. If the data type of the spool file is EMF any printer should work. Some forensic tools have support for decoding EMF files so that they can be viewed; Forensic Toolkit is one such tool.

There are some features that a user could use to protect the printed files from being disclosed. One is to use Encrypted File System (EFS), but if not configured properly the encryption will have no effect. Using EFS on a computer will allow for encryption on selected folders and its contents. If the C:\WINDOWS\SYSTEM32\SPOOL\PRINTERS folder is not encrypted all files will lose their encryption when printing. This means that even if a file is encrypted on disk the content will be possible to recover if the file has been printed out and the .spl file is successfully recovered. Using encryption on the printer folder will allow for the file to be encrypted during the printing process. It is also possible to print files without letting Windows create the temporary spooler files. By configuring the printer to "print directly to the printer", no .spl file is created.

3.2.10.1 Case scenario: Print spooler files

The following case scenario describes a case where print spooler files were used as evidence in a court trial. The case scenario is from [Cas01].

"Print spooler evidence was the only evidence in a counterfeiting case in Orange County, California. Department of Consumer Affairs examiners arrested a suspect for selling counterfeit state license certificates and seized his computer. Although the examiners had seized some of the counterfeit certificates from victims, they were unable to locate evidence on the computer. When the examiners requested a second review from the California Department of Insurance, Fraud Division, the Computer Forensic Team identified several deleted enhanced metafiles that exactly matched the paper copies that had been seized during the investigation. The only evidence present on the drive was the enhanced metafiles. The defendant was convicted at trial."

3.2.11. Additional locations of evidence

3.2.11.1 Alternate data streams

An interesting aspect concerning ADS is that they are transparent to Windows file management tools. This implies that ADS is not visible when listing files and folders using Windows Explorer (or the command line executable DIR) and prior to Windows XP ADS didn't appear in the process listing in Windows Task Manager. ADS could therefore be used to hide information from being discovered, or used to run code in an "invisible mode".

3.2.11.2 Host Protected Area

Host Protected Area (HPA) is not a NTFS specific feature but it is an important topic and therefore a brief explanation will be given below.

Hard disks can have an area that is not normally accessible by users. The HPA is a storage area outside the file system and was designed so that PC distributors could store for example

diagnostic utilities or other data on the disk. By using specialised software this area could be used to hide data and many forensic image tools are still not able to image the HPA.

One way to discover if the HPA is used is to examine the boot partition for changes made to allow for accessing the HPA. An earlier well-known tool named AREA51 did just that; it modified the boot partition to include pointers to the HPA. HPAs can also be detected by executing low-level ATA commands to get the maximum disk address and the maximum user accessible address. If there is a difference between the two a HPA exists and should also be imaged and analysed.

3.2.11.3 Hiding information in bad clusters

NTFS marks clusters as bad when sectors on the disk is damaged. Clusters marked as bad will be ignored by the operating system and the cluster is no longer addressable. This could be used for hiding information, but there is not much literature on this subject.

When a sector goes bad on a hard disk NTFS marks the entire cluster as bad. That cluster will not be used by the system anymore. However if only one sector is bad, the rest could be used for hiding information. In order to store data in the good sectors of the cluster marked as bad, the cluster has to be unmarked manually. When the information has been written to the good sectors the cluster is marked as bad again. To retrieve the information the cluster is unmarked, the data is read, and the cluster is marked as bad again.

It is of course possible to mark good clusters as bad in order to hide more data; several clusters could be used to store a larger portion of data.

One of the reasons why this is not a well discussed topic might be the difficulties of marking and unmarking bad clusters. There is little information on the subject and I've been unsuccessful in finding any tool doing this in an automated fashion. Well, how are clusters (un)marked in NTFS? The only information I have found about this is in the FAQ section on a website of a company developing computer forensic tools [Inet05]. A question about how clusters are unmarked in NTFS was asked in the FAQ and the CEO of the company gave the following answer to the question:

“You could open the NTFS drive in WinHex and use the directory browser to list the clusters allocated to the system file \$BadClus. These are the bad clusters. Individual sectors are not accounted for. Usually \$BadClus has a size of zero, of course (=no faulty clusters). To unmark bad clusters, you would have to edit the FILE record describing \$BadClus, i.e. at least the data runs and the file size, and you would have to mark the clusters as free (unallocated) in the system file \$Bitmap.”

Winhex is a hexadecimal editor developed by their Company. The answer make sense because the \$BadClus is the NTFS specific metadata file that includes all bad clusters on the disk and the \$Bitmap file is another NTFS metadata file and, as explained in 3.1.1.3, it contains a bitmap view of allocated clusters on the disk.

3.2.12. Future file systems and computer forensics

As Florian Buchholz and Eugene Spafford explain in [Buc04] there are some shortcomings of the information that is searched for evidence on current systems. In the paper they discuss

what information is desired for different types of forensic investigations, how feasible it is to obtain it and how the information is to be stored.

Buchholz and Spafford categorize the desired information as:

1. Information that is available to the system and recorded on non-volatile media
2. Information that is available to the system but is not recorded.
3. Information that is not currently available to the system but could be made available.
4. Information that is impossible to be obtained by a computing system.

The two first categories above are the ones that are available on most computers today, but the other two are interesting in a forensic point of view. The authors of the paper takes up an interesting viewpoint of computer forensic since with some changes to the file systems more information would be available and it would be easier to reconstruct what has happened when and who did it etc. The system could be designed in such way that the fundamental questions mentioned earlier (who, what, when, how, where and why) could be answered with a high degree of forensic value and quality.

4. Computer forensics tools

4.1. Defining computer forensic examination and analysis tools

Computer forensics includes acquisition of data, extraction and analysis data, preservation and presentation of evidence. A lot of tools have been developed to support these different steps in the forensic process. Unfortunately most of these tools only provide access to evidence but do not provide methods for verifying the reliability of the results [Car03].

In [Car03] Carrier defines the goal of identification and analysis phases of digital forensics as:

“To identify digital evidence using scientifically derived and proven methods that can be used to facilitate or further the reconstruction of events in an investigation.”

There are a lot of tools available on the market that are supposed to do wonderful things, but without using a proven method the reliability of the evidence should be contested. A lot of factors can affect the results presented by a tool, and the more complex the tool is (or the system under investigation) the more likely it is that an error is introduced.

Computer forensic tools are used to overcome two of the basic problems with digital data; the *complexity problem* and the *quantity problem*.

The complexity problem refers to that data in its raw format are too complex for a human to understand and interpret in an efficient way. Tools are therefore used to help the investigator interpret and translate the information so it can be understood.

The quantity problem is that there are often huge amount of data that must be analysed and going through every bit of data is highly inefficient. Computer forensic tools use different data reduction methods to sort out the information that needs closer attention.

4.1.1. Layers of abstraction

In [Car03] Carrier describes the purpose of a computer forensic tool as:

“to accurately present all data at a layer of abstraction and format that can be effectively used by an investigator to identify evidence. The needed layer of abstraction is dependent on the skill level of the investigator and investigation requirements.”

A layer of abstraction translates the input data to an understandable format using a rule set that defines the way to interpret the input data. That could for example be translate a raw data file consisting of bits (1 and 0) into an UNICODE formatted text, using the UNICODE character set as the rule set in the translation. Output from one abstraction layer, like the UNICODE formatted text previously mentioned, could be fed as input to the next layer of abstraction. Another rule set could be used to for example filter out specific words from the text.

It is important to understand that errors could be introduced in every layer of abstraction and the more layers included in a tool the greater the risk for an error. Two errors that refer to abstraction layer are:

- **Tool implementation errors** – errors introduced by programming flaws or tool design errors.
- **Abstraction errors** – errors introduced when an abstraction layer is translated using a rule set that is not sufficiently well defined. This could lead to errors in the translation because the tool has to make assumptions of how to interpret the data.

No matter how careful the designers and the programmers are when developing a tool, there is always a risk that there are errors introduced making the output data inaccurate. If possible, a margin of error should therefore be calculated. The margin of error is a great help during the analysis of the results and thus avoid making misinterpretations. To minimize the margin of error the accuracy of the input data must be verified and the rule set must be as complete and accurate as possible. Figure 12 illustrates the abstraction layer translation.

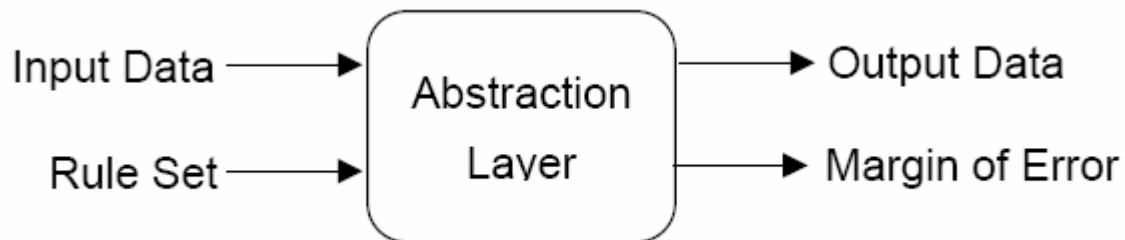


Figure 12 Abstraction layer, figure is from [Car03]

Layers of abstraction are often nested, i.e. there are several layers of abstraction that makes up a higher-level layer. In a simple example from [Car03] the storage of a HTML file has 4 higher-level layers; Physical media, Media management, File system, Application. These higher-level layers use the output from the previous level as input. Within each level there are several layers of abstraction and it is the output from the boundary layer that is used as input for the next level. The boundary layer is the last layer within the higher-level layer. Figure 13 shows the abstraction levels and layers of an HTML file.

Physical Media			Media Management	File System			Application
Head	Cyl	Etc.					
Sectors			Partition Table				
			Partition	Boot Sector	FAT	Data Area	
			...				
			File			ASCII	
						HTML	

Figure 13 Abstraction levels and layers of an HTML file, figure is from [Car03]

4.1.2. Categorizing computer forensics tools

There are two categories of computer forensics tools; *translation tools* and *presentation tools*. Translation tools take input data and use a rule set to translate the data into an output, which still may be hard to interpret. The output from the translation tool is therefore used as input to a presentation tool which uses another rule set to present the information in an easy understandable format.

Many computer forensic tools suites implement both categories of tools within the same package. Several layers of abstraction are therefore handled by the same tool and the complexity of the tool is obviously increased. The increased complexity of the tool will increase the risk of errors being introduced, since there are several layers of abstraction in multiple levels.

Another way to categorize computer forensic tools are at the level they operate; Bootable environments tools, Data acquisition tools, Media management analysis tools, File system analysis tools, Application analysis tools and Network analysis tools.

- **Bootable Environments Tools:** Software that you can use to boot a suspect system into a trusted state.
- **Data Acquisition Tools:** Tools used to collect data from a suspect's system.
- **Media Management Analysis Tools:** Tools used to examine the data structures that organize media, such as partition tables and disk labels.
- **File System Analysis Tools:** Tools used to examine file systems and disk images to recover and view the content of files and folders.
- **Application Analysis Tools:** Tools used to analyze the file content for example viewing log files, images etc.

There are other categorizes as well like Physical media analysis tools, Network analysis tools, Memory analysis tools, but these are out of the scope for this report.

4.2. Commercial Computer forensics tools

There are several commercial tools available on the market. Following are a brief description of some commercial tools that have been used in this report. Encase have not been used, but since it is a common and well-known tool Encase is also mentioned in the discussion below.

It is important to be aware that even though the commercial tools have been thoroughly tested during the development phase there may still be abstraction errors or tool implementation errors that have not yet been discovered. This is true for all tools, and “fixes” are often described on the company homepage where new releases of the tool are described. Because technology is constantly developed there are also features that not yet have been implemented by all tools.

The Decision on which tool to use should depend on the purpose of the investigation. A single tool is never the best for every investigation; one tool may be well-suited for a specific problem while another is better for a different problem. It is also good practice to use a tool in conjunction with another tool. This minimizes the risk that tool implementation errors and/or abstraction errors go unnoticed.

4.2.1. Encase (Guidance Software)

Encase [ENCA] is the most widely used tool by professional computer forensic investigators. It is a complete computer forensic tool suite that could be used for imaging hard disks and analysing image files or live systems, i.e. Encase can be used for data acquisition and can be used as a media management, file system and application analysis tool. It is a complex translation and presentation tool that involves several layers of abstraction.

The Encase licence fee is high and the tool is therefore mostly used by companies specialised on computer forensics. It is one of the “accepted” tools in court, and Encase have been used in numerous of investigations.

4.2.2. Forensic Toolkit (AccessData Corporation)

Forensic Toolkit [FTK] is another computer forensic tool suite, but with a licence fee that is much lower than for Encase. The tool involves, just like Encase, many layers of abstraction and is a combined translation and presentation tool. Forensic Toolkit is also widely used, and has been thoroughly tested.

Forensic Toolkit (and Encase) supports several different file systems and allow the investigator to recover files both by physical and logical techniques.

4.2.3. Winhex (X-Ways Software Technology)

Winhex [WINH] is a hexadecimal editor that offers many examination and analysis capabilities (i.e. it is a combined translation and presentation tool), Many layers of abstraction are used, but it is not as complete as Encase or Forensic Toolkit.

File recovery is done by using file carving (physical recovery), and as described earlier in the report that is somewhat limited. File carving with Winhex is primarily done on cluster boundaries, since that is where file headers normally is found. Files not aligned to cluster boundaries can be found though, because Winhex includes features to search for sector-aligned file headers and to make an even more thorough search at byte-level.

4.2.4. X-Ways Trace (X-Ways Software Technology)

X-Ways trace [TRAC] is an application analysis tool, and is used to track and examine Internet activity and deleted files on Windows computers. It is a combined translation and presentation tool that deciphers index.dat and INFO2 files found on Windows computers. It uses less abstractions layers than the tools mentioned above and the results can be exported and presented in for example MS Excel.

4.3. Open source Computer forensics tools

The debate of whether to use open source tools, that may be tested by a larger audience since the code is readily available, or use commercial (black box tools) is out of the scope of this report. One thing is for sure, there is more and more computer forensic tool available for free and some of them have successfully been used in court.

Errors introduced by a open source tool may be discovered earlier than for commercial tools, since the source code is available to the public and can be examined by a large number of people. This may not be the case though, and as with commercial tools there may be errors (tool implementation errors and abstraction errors) that have not been discovered.

Following is a short description some open source computer forensic tools.

4.3.1. The Sleuth Kit (TSK)

The Sleuth Kit [TSK] is probably the most well-known and the most widely used open source computer forensic tool suite.

The Sleuth Kit (TSK) is a collection of UNIX-based command line tool. There is a graphical interface to the tools in TSK that could be used to get an easily manageable environment. The graphical interface is called Autopsy and is HTML-based. TSK includes several tools that each performs their specific tasks. The tools included in TSK include few layers of abstraction. TSK allows viewing a disk both physically and logically.

4.3.2. File Date Time Extractor

There are several open source computer forensic tools available for download at Digital Detective's homepage, but only the File Date Time Extractor [FDTE] has been used in this report.

File Date Time Extractor is a software application that searches files for embedded 64 bit time stamps. There are several time stamps inside files like Word documents that could be of use in an investigation. This tool may give false positives since patterns that may look like a time stamp can occur naturally in a file. The tool includes few layers of abstractions and is a combined translation and presentation tool.

5. Dangers of misuse of Forensic Analysis Tools

Previous chapters have covered topics of how to find evidence of illegal activity on a computer. Many of the same techniques can of course be used with the intention to steal sensitive information or for other malicious intent. Therefore this chapter is reserved to discuss the potential misuse of forensic analysis tools, which could interfere with information security and intrude on company or private information.

5.1. Potential misuse of forensic analysis tools

In office environments information is often stored on protected servers that are located in locked rooms. This might seem pretty secure but in reality computer users often store information locally on the computer to be able to work offline. This is especially true when it comes to laptop computers which are more prone to theft or loss than a normal desktop computer.

This report has covered the basics in computer forensics and explained what can be done with computer forensic tools. There are a lot of powerful tools for free or that can be acquired for a small amount of money. The “investigator” (intruder) could use a bootable forensic CD which could allow him to make an image of the hard disk of the computer and thereby get hold of all information stored on the computer. Forensic tools also uses write protection which makes it almost impossible to detect if such a tool has been used, since no traces will be left on the target computer. All that it takes is that the intruder has physical access to the computer, and in office environments it is common that everybody that has access to the office also has access to all the computers.

Using a computer forensic bootable CD will bypass both the login procedure and NTFS files system permissions, since both these functions are dependent of the file and operating system. Following are some areas in which computer forensic tools could be used with a malicious intent.

5.1.1. Information theft

An attacker could use computer forensic tools in order to steal confidential information that if disclosed can have a severe effect on the company. There are people and companies willing to pay for trading secrets, patents and other sensitive information.

There is of course other information that may be stored on the computer and that an attacker would like to steal, like credit card information, personal information etc.

5.1.2. Supervision

Computer forensic tools could also be used in order to keep an eye on someone, for example an employee. The tools could be used to read e-mails, track Internet activity and so on. There are several applications that do this on a network basis, but they will not disclose all the actions that a computer user has taken. For example, laptop computers are often connected to other networks and other mail servers could be used to send and receive e-mails. These

actions could be examined using computer forensic tools. That obviously raise some ethical considerations, but that is out of the scope for this report.

5.2. Avoiding misuse of Forensic Tools

In the BIOS setup it is possible to set the booting sequence to the preferred order in which the computer should look for devices to boot from. The BIOS configuration could be changed to only boot from the hard disk and not from any other media. In this way an attacker can not boot another operating system without changing the BIOS configuration. It is therefore important to protect the BIOS from being changed by an unauthorized person. In newer computers it is possible to activate two different passwords for the BIOS. One is used for booting the computer and the other is used when changing the configuration settings. The former might be a bit intrusive on the usability since there will be an additional password (supposed a user account password is used) to type in every time the computer is turned on. The latter might be a better choice since it must only be used for the BIOS to accept changes to the settings.

It is important to notice that the use of BIOS passwords is not totally secure. The hard disk could be removed and put in another computer where the attacker has access to the BIOS configuration and therefore could boot another operating system to make a disk image. Another approach an attacker may take is to physically reset the passwords. On the motherboard of the computer there is often a BIOS reset switch which clears the BIOS and the passwords. After resetting the passwords the attacker may change the booting sequence to boot from another device. There is still another way, and that is to remove the battery for the CMOS chip (where the BIOS is located). Removing the battery will reset the BIOS configuration and gives the attacker the same opportunities as mentioned above.

As explained above configuring the BIOS to prompt for a password, to accept changes to the settings, is not totally secure. But it makes it a lot harder for an attacker to make a disk image, so passwords should be used especially on laptop computers. Removing the hard disk or get physical access to the motherboard on a laptop computer is also a lot harder than on a desktop computer.

BIOS passwords should not be the only protective measure, but instead one of them. It is good practice to enforce defence in depth to secure the information on the computer. In Windows 2000/XP a new feature was introduced, EFS (Encrypting File System). EFS gives the opportunity to encrypt information stored on the computer. EFS can only be used on NTFS formatted disks, EFS is not supported on FAT systems.

Another security measure that is included on most hard disks today is the ATA-password. There are several different names on this password feature that is included in the ATA-specification, for example HDD Password and Security lock are two examples of frequently used names. This feature is especially common on laptop computers and is a good complement to BIOS passwords and encryption.

Enabling the ATA-password will reject read and write operations to the hard disk if not the correct password have been typed in [Inet02]. The password is stored on a reserved area on the hard disk and moving the hard disk to another computer will not bypass the ATA-password. It is possible to unlock the hard disk using specific hardware, but that hardware is expensive.

Security is always a question of time and money. The security features mentioned above are all implemented in the computers and do not cost the user anything. They provide a fairly good security against the potential misuse of computer forensic tools, but if the attacker has the time and the money the security measures mentioned can all be circumvented. If security is of high priority there are other security measures that can be bought to provide a higher degree of security.

Computer forensic tools are invaluable to forensic investigators during a forensic examination of a computer system. The increasing number of free or almost free computer forensic tools is now becoming a threat to companies and private persons. Today there is a lot of sensitive information stored on computers which could have a devastating effect if disclosed. The potential misuse of forensic tools must be taken seriously and it is important to take protective measures to secure sensitive information stored locally on the computer. Enabling the ATA-password, enabling the use of EFS, configuring BIOS only to boot from the hard disk and using a password for changing the BIOS settings are some features that could and should be used. It makes it a lot harder for an attacker to disclose information, through the misuse of computer forensic tools.

6. Research findings

Following is a discussion about the results, limitations, reliability and validity with the research, conclusions and some comments on future work. Limitations, reliability and validity were briefly covered in the method discussion in the beginning of the report. The discussion below is a follow up and describes the outcome after the writing of the thesis has been completed.

6.1. Results

Since this research was a theoretical research study the result is the report itself. There were four central questions formulated in the introductory chapter and those questions have been answered in chapter 2-4.

Chapter 2 is the theoretical framework for the research study and gives an introduction to the forensic process and how computer systems are analysed. The focus of the report was on how to apply forensic science on NTFS computers, but in order to put the subject in a bigger context the forensic process had to be covered briefly.

In chapter 3 the theoretical framework was used and applied to Windows NTFS computers, and in the beginning of the chapter important aspects concerning NTFS were covered. The theory has been explained and visually exemplified using different computer forensic tools. By using visual presentations of the finding the reader is introduced to both the theory and how tools can be used in an investigation. Tools used in the report have also been described in chapter 4.

With the ambition to give the reader another viewpoint of computer forensics, chapter 5 was added covering the potential misuse of computer forensic tools. This topic was not included in the original plan for the thesis, but the subject was considered important and was therefore included in the report.

6.2. Limitations

The decision to not use experimental research and only cover NTFS has allowed for a more thorough literature review. But since the literature on the subject is limited there were some problem areas that could have been clarified and others that could have been verified by using experimental research. Unfortunately there wasn't enough time for conducting experimental research.

The legal aspects on computer forensics have not been covered in this report, but it is a very important area. There are a lot of rules and regulation that has to be followed during a computer forensic investigation. Investigations involving several countries may be complicated in a legal point of view, because rules and regulations may differ between countries. Omitting the legal aspects from this thesis has allowed the focus to be on the technical aspects of computer forensics. It has also been possible to use different tools to visually describe the theory without the need for a discussion whether the tool uses accepted methods.

Application specific files have not been covered, because examination of such files is dependent of the purpose of the investigation and the specific application that created it. The files covered by this report are files that are not created by a specific application rather by Windows itself and that can yield information that is valuable in a computer forensic investigation. There is one exception; the index.dat files described in chapter 3.2.6 are created by Internet explorer, but the importance of that file is too great not to be discussed. Another file that also could have been discussed is the .pst files created by Outlook. These files include e-mails sent and received by MS Outlook, but since there are several other mail programs widely used a discussion about the .pst files was not included in this report.

6.3. Reliability and validity of the research

This report is written following an extensive literature review. One of the major problems with such literature review is to determine if the information gathered is reliable or not. As mentioned in chapter 1.9 literature and secondary sources of information are associated with Comparability and Reliability problems. In the ambition to present reliable facts and counter the reliability problem the focus of the literature review has been on articles published in peer-reviewed and accepted journals and on printed books on the subject. Researchers and authors of white papers, and other secondary sources have been checked upon.

Computer forensics is a relative new science and the amount of literature on the subject is somewhat limited. Literature regarding computer forensics applied to NTFS has been especially hard to come by. Most literature found was focused on FAT file system but since there are many similarities between FAT and NTFS the information could sometimes be applied to NTFS as well. Obviously such information is subject for a comparability problem. The information was therefore analysed thoroughly and verified on computers utilizing NTFS file system.

6.4. Discussion

Following are some conclusions drawn from the research.

Chapter 3.2 described how forensic science is applied to NTFS computers. Various sources of evidence were covered such as index.dat files, cookie files, print spooler files, thumbs.db files, registry files, and INFO2 file etc. These and other sources of evidence outside the computer should be examined such as DNA, fingerprints, and notes on paper etc.

Searching several sources of evidence increase the possibility for the investigator to draw unambiguous conclusions about what has happened. There are three major categories of evidence; inculpatory evidence, exculpatory evidence and evidence of tampering. That means that evidence like time stamps, internet activity and installed software all support or contradict a given theory or show that the system was tampered with. All evidence collected (inculpatory, exculpatory and evidence of tampering) is to be used in the relational, temporal and functional reconstruction of the crime. Analysing and combining evidence collected from the various sources described, helps the investigator to reconstruct the crime and to draw unambiguous conclusions.

Time stamps plays an important role in the reconstruction of a crime, but time and dates may not always be displayed in a correct manner. Some applications make adjustments for local

time and daylight savings and some does not. It is obviously an important aspect to be aware of since this could have a severe effect on the case. Time and date information should therefore be thoroughly examined to be sure that the correct information is obtained.

One of the major concerns in an investigation is to link the evidence to a physical person. It could be hard to prove that the suspect is connected with the evidence found on the computer. This is the biggest challenge for the investigator, since anyone could have used the computer especially if password protected accounts have not been used. Physical evidence may tie a suspect to the computer, for example it may be possible to prove that a specific person has used the computer if fingerprints are found on the computer keyboard. Knowing that the suspect has used the computer makes it easier to draw further conclusions. This is one of the reasons why securing evidence outside the computer is important.

File recovery pervades much of the forensic work during a computer forensic investigation, whether it is to recover the actual deleted files (word documents, graphics image files etc.), temporary files (printer spool files) or files containing metadata (index.dat, thumbs.db, INFO2).

File recovery is important in an examination of a computer, since a lot of the information may be deleted. Deleting files on NTFS computers does not mean that the file is permanently deleted. The file is often recoverable from unallocated space or slack space. Even if the data has been overwritten and the file is not successfully recovered there may still be a lot of traces left on the computer that can give evidence of the files existence. The chance of successfully recovering files decreases with time, because the clusters in unallocated space may be overwritten. NTFS also overwrites MFT records relatively quickly, making file recovery harder.

Software tools are only part of the solution, and should not be explicitly trusted. The tools should be tested to verify that the tool behaves in a certain manner. It is always important to understand how a tool handles different tasks and this is even more important if the tool is new and has not been thoroughly scrutinized by independent bodies. There are some widely used and accepted tools that could be used in most investigations. Other tools that are less used and that are developed for a specific task may also be used but the investigator should be prepared to answer questions in court related to the tool and its inner workings. The decision on which tool to use should depend on the purpose of the investigation. The investigator's knowledge and skill with the various tools could also be of importance.

No matter how skilled an investigator is with a tool it is also important that the investigator is familiar with the file system run on the computer. Understanding how the file system works will help the investigator to interpret the data found, or not found.

There is also another side of computer forensics; tools developed for computer forensic examinations of computers may be used with a malicious intent. This is a serious threat and there are several powerful computer forensic tools which could be used by an attacker. Such tools may be used to steal sensitive information or to keep an eye on someone. Using a forensic tool makes it is possible to bypass security measures like password protected accounts, but there are several ways to protect the computer and the information stored on it. Some examples of security measures that should be used is the password protection of BIOS, enable the ATA-password and use EFS.

6.5. Future work

This report has covered Windows NTFS file system, but computers may use other file systems. Windows computers may for example use FAT and Unix and Linux computers use EXT2 and EXT3 file system. There are still other file systems used by other operating systems. A computer may also have several operating systems and different file systems on one physical hard disk. In order to investigate computers with multiple operating systems and file systems the investigator therefore needs to have knowledge of these additional file systems. Future studies could be done on file systems not covered by this report and should at least cover the most widely used file systems on computers today.

Computer forensic tools can help finding hidden data in unallocated space or in hidden partitions, but as far as I know, there is no tool that looks for data in clusters marked as bad by NTFS. It is possible to manually mark and unmark bad clusters and therefore logically it should be possible to hide information in such data areas. Future studies should be made on this topic.

The legal aspects on computer forensics are an interesting area that should be furthered investigated. Is there a way to categorise and approve computer forensic tools for certain investigations and situations? How are cross-country investigations handled, and how are differences between the countries rules and regulations managed?

7. References

7.1. Books and per-reviewed articles

- [Alt04] Altheide Cory, Forensic analysis of Windows hosts using UNIX-based tools
Digital Investigation Volume 1, Issue 3, Pages 197-212 (September 2004)
- [Boy04] Boyd Chris and Forster Pete, Time and date issues in forensic computing – a case study.
Digital Investigation Volume 1, Issue 1, Pages 18-23 (February 2004)
- [Buc04] Buchholz Florian and Spafford Eugene, On the role of file system metadata in digital forensics.
Digital Investigation Volume 1, Issue 4, Pages 18-23 (December 2004)
- [Car03] Carrier Brian, Defining Digital Forensic Examination and Analysis Tools Using Abstraction Layers
International Journal of Digital Evidence Winter 2003, Volume 1, Issue 4
- [Cas00] Casey Eoghan, Digital evidence and computer crime
Academic Press; Bk&CD Rom edition (March 15, 2000)
- [Cas01] Casey Eoghan, Handbook of Computer Crime Investigation
Academic Press; 1st edition (October 15, 2001)
- [Cas04] Casey Eoghan, Tool review – Winhex
Digital Investigation Volume 1, Issue 2, Pages 114-128 (2004)
- [NIJ01] National Institute of Justice. Electronic Crime Scene Investigation: A Guide for First Responders. Washington, D.C.: U.S. Department of Justice, National Institute of Justice, 2001. NCJ 187736.
<http://www.ojp.usdoj.gov/nij>
- [NIJ04] National Institute of Justice. Forensic Examination of Digital Evidence: A Guide for Law Enforcement. Washington, D.C.: U.S. Department of Justice, National Institute of Justice, 2004. NCJ 199408.
<http://www.ojp.usdoj.gov/nij>
- [Sol00] Solomon David A. and Russinovich Mark E, Inside Microsoft® Windows® 2000, Third Edition
Microsoft Press, Redmond, Washington (2000)

7.2. White papers

- [WP01] Carrier Brian, Open Source Digital Forensics Tools: The Legal Argument (September 2003)
http://www.cerias.purdue.edu/homes/carrier/forensics/docs/opensrc_legal.pdf
(2004-11-25)

- [WP02] Jones Keith J., Forensic Analysis of Internet Explorer Activity Files
http://www.foundstone.com/pdf/wp_index_dat.pdf (2005-02-07)
- [WP03] Russinovich Mark, Inside NTFS
<http://www.windowsitpro.com/Articles/Index.cfm?IssueID=27&ArticleID=3455> (2005-01-10)

7.3. Internet sources

- [Inet01] Examples of Computer Forensics in Action
http://www.forensics.com/html/whats_new_case_studies.html (2005-02-15)
- [Inet02] Atea Security
<http://www.atremo.se/informationscentrum/security-info/sakerhetsartiklar/276.html> (2005-03-24)
- [Inet03] Mil Incorporated
http://www.milincorporated.com/a3_index.dat.html (2005-03-11)
- [Inet04] AceSoft
http://www.acesoft.net/delete_index.dat_files.htm (2005-03-11)
- [Inet05] X-Ways Support Forum
<http://www.winhex.net/> (2005-03-07)
- [Inet06] Windows Registry Tutorial by WinGuides.com
<http://www.winguides.com/article.php?id=1&guide=registry> (2005-03-15)

7.4. Computer forensic tools

- [ENCA] Encase, Guidance Software Incorporation
<http://www.guidancesoftware.com/> (2005-02-14)
- [FDTE] Digital Detectives
<http://www.digital-detective.co.uk> (2004-12-19)
- [FTK] Forensic Toolkit, AccessData Corporation
<http://www.accessdata.com/> (2005-02-17)
- [TRAC] X-Ways Trace, X-Ways Software Technology AG
<http://www.x-ways.com/> (2004-12-18)
- [TSK] The Sleuth Kit, Carrier Brian
<http://www.sleuthkit.org/> (2005-03-30)
- [WINH] Winhex, X-Ways Software Technology AG
<http://www.x-ways.com/> (2004-12-18)