Computer Anti-forensics Methods and Their Impact on Computer Forensic Investigation

Przemyslaw Pajek and Elias Pimenidis

School of Computing IT and Engineering,
University of East London, United Kingdom
`pajkow@gmail.com, e.pimenidis@uel.ac.uk`

Abstract. Electronic crime is very difficult to investigate and prosecute, mainly due to the fact that investigators have to build their cases based on artefacts left on computer systems. Nowadays, computer criminals are aware of computer forensics methods and techniques and try to use countermeasure techniques to efficiently impede the investigation processes. In many cases investigation with such countermeasure techniques in place appears to be too expensive, or too time consuming to carry out. Often a case can end up being abandoned and investigators are left with a sense of personal defeat. The methodologies used against the computer forensics processes are collectively called Anti-Forensics. This paper explores the anti forensics problem in various stages of computer forensic investigation from both a theoretical and practical point of view.

Keywords: Computer Forensics Investigation, Computer Forensics Tools, Computer Anti-forensics Methods.

1 Introduction

Locard's principle states that when a crime is committed, there is a cross-transfer of evidence between the scene and perpetrator [1]. In the digital world, evidence resides mainly on computer hard drives in the shape of files, logs, or any other artefacts depicting pertinent activity. Projecting Locard's principle into the cyber world an understanding of the correlation between such types of evidence, the times when particular events took place and the users who committed those actions can be reached. The main task of computer forensic investigators is to reveal and connect these three facts into one coherent statement revealing the whole nature of the particular action. On the contrary, the main aim of computer anti-forensics is to hide or alter electronic evidence so that it cannot be used in legal proceedings or it is too costly and time consuming to retrieve and examine. Computer anti-forensics methodologies vary and can be applied so they can contaminate any stage of the computer investigation process. Whilst most of the techniques are used directly against computer forensic tools, some of these methodologies can be used for quite legitimate reasons. Encryption for example can be used to protect company assets; digital watermarking can be used to prevent copyright infringement in

digital imaging. Conversely, these techniques, if applied against computer forensics, could potentially hide crucial data from investigators.

Many authors have discussed computer anti-forensic techniques and have hailed them as very efficient ones. However, very little practical work has been done in this area in terms of testing those techniques and practically evaluating their efficiency and effectiveness. The main aim of this research is to identify the most known computer anti-forensic techniques and test practically them against computer forensic software. The key question to be addressed is whether computer anti-forensics can hinder the investigation process and prevent real artefacts being discovered and being admissible in the legal process? The work presented here is based on the experimental part of the first author's MSc dissertation.

2   Computer Forensics Methodologies versus Anti-forensics

To efficiently test those techniques, it is necessary to identify the stages of computer forensic processes where investigators follow clear and well defined procedures [2]. At every stage different computer forensic methodologies have been used and various anti-forensic techniques have been applied against these methodologies.

2.1 Stage One – Elimination of Source

Preservation of data process relies on the securing of all data found on an inspected drive irrespective of its pertinence to an investigation. At this stage the main task is to only acquire an identical image of the analysed media.

One of the main methods which could be applied in this task is to prevent pertinent data being preserved. One of the easiest and most efficient methods would be to block the access to the media; however when investigators have permission to investigate it, such a move is not possible. Therefore the next sensible option is the elimination of a source [3]. Like most of counter-forensic techniques these methodologies can be applied only before image acquisition.

The easiest method of elimination of a source is simply a disabling tool responsible for creation of source. This could be, for example, done by various modifications of computer settings and registry. Operating system will stop logging users' activity and so. If a user performed a particular action, this would be automatically hindered. An example of this can be the editing of an operating system group policy, in this way the system will not log a visited website in the browser's history [4].

The next method of elimination of a source is log and disk wiping. This method relies on the deletion process, where special tools need to be applied in order to "safely" delete traces of data from any places on the hard drive. This should include all entries on HDD incl. all MFT entries and its attributes, orphan files and so forth. Previous research in this area has revealed that not all programs claimed to be anti-forensic can efficiently delete all traces of data [5]. Similar experiment was used to check whether a new set of tools available in 2008 were more efficient than those applied in 2005.

2.2   Stage Two – Hiding the Data

In the second stage of the computer forensic process, usually investigators identify and extract the information which can be pertinent to the investigation. At this stage counter forensic tools have the greatest use. In contrary to the previous stage, they do not delete relevant data but hide it in a way that it is very difficult to find and examine. Research about these methods has distinguished some ways of how data can be hidden on digital media.

Unusual directories and manipulation of file headers - in this method, information is hidden in unusual places. For example, the file system and deep nested directories. This gives a greater likelihood that hidden files will be overseen during the "harvesting" process. Another method which would inhibit investigators is the manipulation of file extensions and file headers. Whereas simple manipulation of extension does not make any difference for forensic software, manipulation of file headers and footers may potentially misled forensic software.

Hiding data in slack space – On media like hard drives, the data is saved in clusters divided by sectors. Currently the most popular files system in use is NTFS. Main information about files such as file name, size, time stamps and the number of clusters is stored on the hard drive [6]. However, actual data is stored in other place. In many cases saved data does not use all sectors in a dedicated cluster, some of them remain unused. Counter forensic tools can use those places to store data. Tools like Slacker can scrap and spread data into those places. The only way to retrieve it is to use slacker again by special reversible algorithm.

Stenography – This is a technique of hiding data within other data where the presence is not revealed. In many cases it is used for legitimate reasons by inserting digital watermarks in the image, so the owners can easily protect themselves from copy right infringement. Simultaneously, instead of watermarks, any other data can be inserted. Investigators using forensic software may easily bypass data hidden in that way, as many forensic packages are not especially dedicated to reveal steganography. Every innocent family picture may be in fact a well camouflaged collection of very important data [7].

Encryption – in many cases it is used to protect information from unauthorized access but it can be applied against the computer forensic investigation process. Using encryption, the presence of data is not concealed, but it is extremely difficult to examine encrypted media using forensic software. The process may become too time consuming and too cost full to implement.

2.3   Stage Three – Direct Attacks against Computer Forensic Software

One of the main ways to carry an attack against computer forensics software is to exploit and use its vulnerabilities against it. Computer forensics software, like any other computer program, was created by software vendors. If the credibility of forensics software is put to question during the legal process, any evidence found, may be dismissed due to the unreliability of the software. Currently there are two main ways as to how the reliability of forensic software may be compromised.

Time Stamp Modification – Every file on removable media has four values called M.A.C.E. Those values are responsible for recording Modification, Access, Creation time stamps of that file. Computer forensics packages reading those values, give indications to examiners about time and date issues of any updates and changes to the contents of a file. However those values can be manipulated and real time and date stamps may not be displayed correctly in computer forensics software. Knowing this, a legal defence process may put to question the reliability of computer forensic software.

Hash Collision – Hash function is an algorithm used to create a unique fixed value string from any amount of data. This process is irreversible. In computer forensics it guarantees that digital evidence has not been changed during the investigation process. However in 2005 a student from China created a hash collision from two different inputs of data. The student managed to create the same hash outputs from the two different sets of inputs [8]. Having this ability, any user could efficiently undermine the credibility of digital evidence. During this research, this method was only considered from the theoretical perspective in potential in threatening the credibility of computer forensics software [9].

3   Experimenting with Anti-forensics Techniques

For the experimental part of this work all the techniques discussed here were tested on new hard drives. The criteria used to evaluate the various techniques applied in this experiment were drawn from professional reports on previous research in this area. Based on these, the following test structure was followed during the experimental phase of the work which was split into three stages.

Stage one – for wiping / safe deletion tools

- Test whether traces of previously deleted data can be revealed
- Test whether actual data can be recovered

Stage two – hiding data techniques

- Amount of potential data that can be safely hidden
- Technique applied
- Test whether presence of hidden data can be revealed
- Test whether actual data can be read

Stage three – undermining credibility of the software

- Test whether time stomp modifications can be revealed
- Test whether original time stomp values can be recovered

In determining the counter-forensics tools and methodologies used for experiment the only two criteria applied were those of availability and popularity in the most popular search engines. In most cases counter-forensics methodologies are available through internet search engines. Therefore the assumption that the more popular a tool is in search engines; the likelihood of this tool being applied increases is a plausible one.

Table 1. Tools utilised in the first experiment

Tool Name	Manufacturer	Version	Release Date	License	Downloadable URL
Eraser	Heidi Computers Ltd	5.86.1	2007	Freeware	http://www.heidi.ie/ node/6
Flex TK Express	Flexense	2.8.42	2008	Freeware	http://www.flexense. com/downloads.html
Free Wipe Wizard	Wizard Recovery.com	1.5	No older than 2006	Freeware	http://www. wizardrecovery.com/ free_wipe/free_wipe_ wizard.php
R-Wipe and Clean	R-Tools Technology Inc.	8.1b_14 62	2008	Free 15 days trial	http://www. r-wipe.com/ Disk_Cleaning_ Download.shtml

The hard drives were divided into the same 4 sectors which were the same for all tools. Various sets of documents were saved on them and subsequently erased by those tools. Next every partition was forensically imaged and put into analysis.

For the second experiment, i.e. hiding the data, only one tool was applied for each of the techniques mentioned above:

Table 2. Tools and Techniques for experiment 2

Technique	Tool Name	Manu-facturer	Ver.	Release Date	License	Downloadable URL
Manipula-tion of files signatures	Manual technique / tool used for editing files: Hex Workshop		4.23	2007	Trial	http://download.cnet. com/Hex-Workshop/ 3000-2352_ 4-10298339.html
Hiding data in Slack space	Slacker	Metas-ploit	-	2005	Free-ware	http://www.metasploit. com/research/projects/ antiforensics
	Run time disk Explorer/ Manual		3.41	-	Trial	http://www.runtime.org/ data-recovery-downloads. htm
Stenogra-phy	Invisi-ble Secrets	Neobyte Solu-tions	4.6.2	2007	15 Day free trial	http://www. neobytesolutions.com/ invisiblesecrets/
Encryp-tion	True Crypt	True-crypt	6.1 a	2003-2008	Free-ware	http://www.truecrypt. org/downloads

In this experiment various files of different types were manipulated, data in various files were also hidden by steganography. One partition was completely encrypted. However the Slacker tool did not work when it came to hiding data on the external partition. Sets of errors when reading the master file table on the external partition and lack of relevant documentation about the "Slacker" software made it impossible to use the Slacker tool for the experiment. Therefore further experiments were carried with manual hiding of data in slack space.

For experiment three hiding – manipulation of time stamps, 4 different packages were used in 4 different partitions, various sets of files were put into these partitions, and subsequently different time stomps were manipulated. Four separate forensic images were created and put into analysis.

Table 3. Tools used for experiment 3

Tool name	Manufacturer	Version	License	Downloadable URL
Attribute Changer	Romain Petages	6.10.a	Freeware	http://www. petges.lu/
Attribute Manager	Milksoft	2.6	10 days Trial	http://www. miklsoft.com/ downloads.html
Time Stomp	Metasploit		Freeware	http://www. metasploit.com/ research/projects/ antiforensics/
File Properties Changer	Segobit Software	1.32		http://www. segobit.com/ fpc.htm

Similarly to the previous experiment, the tool Time Stomp was excluded from the experiment as it failed to change most of the MACE values.

All images were forensically analyzed using the following computer forensics software tools:

- Forensic Toolkit-FTK Version 1.71 build 07.06.22 (Demo version)
- FTK 1.81.0 – fully licensed.

3.1 Experimental Results

Experiment one – Using Wiping tools:
Forensic analysis proved that most of the tools applied for wiping data did not efficiently delete all traces of it. From the four tools used, only the fourth one managed to

efficiently erase all traces of data and meet the target of effectively hiding data from forensics tools. The rest of the anti-forensics tools employed in this experiment left many traces that made it possible to prove that the data was previously saved on an analysed hard drive. Table 4 below summarises the outcome of this experiment and shows where traces of data were found following the wiping operation, for each tool utilised.

Table 4. Results of Stage 1 of the experimental work

Tool / Deleted area	Presence of deleted data revealed/ if yes where? MFT Entry	Actual Data/If yes from where?	Forensic tool used
Eraser v. 5.86.1-Freeware	Yes, NTFS\$LogFile, NTFS\$I30	No	FTK 1.71 - demo
External Examiner	Yes, NTFS\$LogFile, NTFS\$I30	No	FTK 1.81.0 – fully licensed
Flex TK Express V. 2.8.42 – Freeware	Yes, Orphan files, NTFS\$LogFile,, $MFT, $I30	No	FTK 1.71 - demo
External Examiner	Yes, Orphan files, NTFS\$LogFile,, $MFT, $I30	No	FTK 1.81.0 – fully licensed
Free Wipe Wizard 1.5 (Freeware)	Yes, NTFS\$LogFile, $MFT, $I30 + List of deleted files	Yes, 60% of data, by data carving	FTK 1.71 - demo
External Examiner	Yes, NTFS\$LogFile, $MFT, $I30 , change log file+ List of deleted files	Yes, 60 % Data , method not defined	FTK 1.81.0 – fully licensed
Tool 4 R-Wipe and Clean v 8.1b_1462	No	No	FTK 1.71 –demo
External Examiner	No	No	FTK 1.81.0 – fully licensed

Experiment two – Hiding the data:
Using the trial version of the FTK1.71 computer forensics package, 50% of the ma-
nipulated signatures were efficiently recognized by forensics software. Manually
hidden data in the file slack space were fully revealed and completely readable in fo-
rensics software. In terms of steganography computer forensics software did not re-
veal the data, but an entropy test [10] raised suspicion. Encryption has also managed
to hide the data so that forensics software did not recover actual data but is at also
raised suspicion. Detailed results can be found in the table 5 below.

Table 5. Experimental results stage two

Techniques applied /Areas of hard drive	Presence of hidden data revealed	Actual Data recovered	Recovery technique applied	Forensic tool used
Manipulation of file Extensions and Signatures	Yes – 50% of tested files	Yes - 50% of tested files	General Forensic Software	FTK 1.71 – demo
External Examiner	Yes- 75 % of tested files	Yes - 75% of tested files	General Forensic Software plus hex editing and checking for extension.	FTK 1.81.0 – fully licensed
Stenography in single image files	No, but raised suspicion	No	General Forensic Software + Entropy tests	FTK 1.71 – demo
External Examiner	No	No	FTK 1.81.0 – fully licensed	FTK 1.81.0
Encryption of Volume	No, but raised suspicion	No	General Forensic Software + Entropy tests	FTK 1.71 – demo
External Examiner	No	No	General Forensic Software	FTK 1.81.0
Manual Hiding in File Slack Space	Yes	Yes	General Forensic Software	FTK 1.71 – demo
External Examiner	Yes	Yes	General Forensic Software	FTK 1.81.0

It is possible that further tests utilizing programs such as Outguess or Stegdetect would have revealed hidden data by stenography or encryption, but forensics software on its own was unable to do it.

Experiment three- Time Stamp Modification:
All three packages measured against computer forensic software succeeded the in value created, however in values for Accessed package the demo version succeeded at

Table 6. Experimental Results Stage 3

Tool used	Time Stamp - value Created	Time Stamp - value Modified	Time Stamp - value Last Accessed	Time stamp - value Entry Modified	Forensic tool used
	Manipulation revealed in % Original. Time stamp recovered in %	Manipulation revealed in % Original. Time stamp recovered	Manipulation revealed in % Original. Time stamp recovered in %	Not Analysed	
Tool 1 - Freeware	No - 0%	No - 0%	Yes–25%		FTK 1.71 – demo
External Examiner	Yes- 100%	Yes – 100 %	Yes – 100 %		FTK 1.81.0
Tool 2 – 10 Days Trial	No - 0%	Yes- 33%	Yes -25 %		FTK 1.71 – demo
External Examiner	Yes- 100%	Yes- 100%	Yes- 100%		FTK 1.81.0
Tool 3- Time Stomp - Freeware	Tool Failed to work	Tool Failed to work	Tool Failed to work		FTK 1.71 – demo
External Examiner					
Tool 4 – Trial	No – 0%	No – 0%	Yes - 50%		FTK 1.71 – demo
External Examiner	Yes- 100%	Yes- 100%	Yes- 100%		FTK 1.81.0

100 % in value created, while for Values Modified and Last Accessed the success rates averaged 33%. In a full version of forensic software, where an experienced external examiner analysed the image, results were more disadvantageous for packages. Detection rate was 100% in all three packages.

4   Discussion – Conclusions

The conducted experiments proved that not all counter-forensics techniques are efficient when compared against forensics software. In many cases tools failed to hide or delete important data. It is highly likely that the detection rate achieved by forensics software depends on the sophistication of the counter-forensics technique applied. This is clearly visible in the encryption and steganography examples where processes and algorithms are much more complex compared to other used in these experiments. The main aim of this research was to test whether current known counter-forensics technology can efficiently interfere with computer forensics processes. The results displayed in the tables in section 3 above clearly indicate that only two of the tested techniques have quite high success rates. The results obtained for other techniques vary amongst them but the average rate is not as high as in those for encryption in steganography. In analysing the results it is very important to take into account the experience of the analyst and the software package applied in the experiment. A more experienced investigator with a fully licensed package could have had a better detection rate than a trainee with trial version only. It is also important to remember that in every forensic case the objectives are different, as are the techniques applied to prevent achieving them.

To achieve more accurate results and to be able to develop a fuller picture as to the current status of counter forensic methodologies, research should be continued in this field. It is recommended that more specific research should be carried about each individual technique against a range of different forensic tools. Also, the various anti-forensics techniques should be evaluated against packages specifically designed for detection of those techniques in order to develop a much clearer opinion as to whether it is possible to beat counter forensics. It is important to note that research on anti-forensics are also a great source of information for those who want to harness systems against those techniques that aim to hide malicious activity and to other interested groups such as software vendors. Knowing, whether counter forensics posses a real threat to computer forensics can be vital in many cases in the future and to the development and shaping of electronic transactions and possibly to the future of the whole digital world.

References

1. Harris, R.: Arriving at an anti-forensics consensus: Examining how to define and control the anti-forensics problem. Journal of Digital Investigation 3(suppl. 1), 44–49 (2006)
2. Warren, G., et al.: Computer Forensics, Incident Response Essentials, p. 4. Addison-Wesley, London (2002)
3. Sartin, B.: Anti-forensics, distorting the evidence. Journal of Computer Fraud and Security (5), 4–6 (2006)

4. Kubrispick: How to Disable Delete Browser History in Internet Explorer,
    `http://www.zimbio.com/Windows+XP/articles/1578/`
    `How+Disable+Delete+Browser+History+Internet`
    (accessed March 26, 2009)
5. Geiger, M.: Evaluating Commercial Counter-Forensic Tools. Carnegie Mellon University,
    Pittsburgh (2005)
6. Carrier, B.: File System Forensic Analysis, p. 283. Addison Wesley, London (2005)
7. Frith, D.: Stenography approaches, options and implications. International Journal of Net-
    work Security (4), 4–6 (2007)
8. Mironov, I.: Hash Functions, Theory Attack and Applications (2005),
    `http://research.microsoft.com/pubs/64588/hash_survey.pdf`
    (accessed May 15, 2009)
9. DRWS, Hash Challenge (2008),
    `http://www.dfrws.org/hashchallenge/index.shtml`
    (Accessed December 19, 2008)
10. AccessData Corporation, FTK manual v 1.71 –Entropy test User Guide (2007),
    `http://www.accessdata.com/media/en_us/print/manuals/`
    `FTK1UsersGuide.pdf`