

On the Genesis of Computer Forensics

Derek Bem

A thesis submitted in fulfilment
of the requirements for the degree of
Doctor of Philosophy

School of Computing and Mathematics
University of Western Sydney
July 2009

© Derek Bem 2009

Let me not seem to have lived in vain.

Tycho Brahe, 1546 – 1601

Dedication

This work is dedicated to my twin sons: Oscar and Conrad, the dual source of light illuminating my path forward.

Acknowledgement

I gratefully acknowledge my indebtedness to those who inspired and encouraged me all along the way. You know who you are, and I thank you.

This is to certify that the dissertation is my own original work except where otherwise indicated. No part of this dissertation has been submitted as part of any other degree.

Derek Bem

30 July 2009

Table of Contents

List of Figures.....	9
Abstract.....	10
1 Introduction.....	12
2 Thesis Outline.....	16
3 Computer Forensics as a Discipline of Science: Development and Trends	19
3.1 Historical Background	20
3.2 Current Research Directions.....	24
3.2.1 Technical Example: Memory Forensics.....	24
3.2.2 Organisational Example: Information System Audit.....	29
3.3 Summary of Contributions.....	31
4 Novel Approaches in Conventional Computer Forensics: File Systems.....	33
4.1 Investigating NTFS File System Images.....	35
4.2 Investigating NTFS File System Backups for Hidden Data.....	37
4.3 Live Investigations of NTFS File System Encrypted Data.....	40
4.4 Summary of Contributions.....	42
5 Virtual Environments: New Technologies For Forensic Investigation.....	44
5.1 Forensic Image in a Virtual Environment.....	45
5.2 Using Virtualisation to Improve the Analysis Process	48
5.3 Easing Reliance on Closed Source Software	52

5.4	Summary of Contributions.....	54
6	Systemising the Computer Forensics Body of Knowledge: Tertiary Education	56
6.1	Emergence of a Discipline of Science.....	57
6.2	Computer Forensics Education	60
6.3	Computer Forensics in Foundation Studies.....	63
6.4	Summary of Contributions.....	66
7	Summary of Contributions and Future Work	68
	References	74
	List of Appendices	85
	Appendix A	90
	Appendix B	108
	Appendix C	118
	Appendix D.....	124
	Appendix E	135
	Appendix F.....	145
	Appendix G.....	162
	Appendix H.....	175
	Appendix I	188
	Appendix J.....	202
	Appendix K.....	209
	Appendix L	223
	Appendix M	229

Appendix N.....	235
Appendix O.....	245

List of Figures

Figure 1-1 Emergence of Computing Machinery and Computer Forensics (time not to scale)	13
Figure 2-1 Thesis Structure.....	16
Figure 4-1 Computer Forensics Methodology and Processes.....	34
Figure 6-1 Computer Forensics – Synergies With Other Disciplines	57

Abstract

This thesis presents a coherent set of research contributions to the new discipline of computer forensics. It analyses emergence of computer forensics and defines challenges facing this discipline, carries forward research advances in conventional methodology, introduces novel approach to using virtual environments in forensics, and systemises the computer forensics body of knowledge leading to the establishment of tertiary curriculum.

The emergence of computer forensics as a separate discipline of science was triggered by evolution and growth of computer crime. Computer technology reached a stage when a conventional, mechanistic approach to collecting and analysing data is insufficient: the existing methodology must be formalised, and embrace technologies and methods that will enable the inclusion of transient data and live systems analysis. Further work is crucial to incorporate advances in related disciplines like computer security and information systems audit, as well as developments in operating systems to make computer forensics issues inherent in their design. For example: it is proposed that some of the features offered by persistent systems could be built into conventional operating systems to make illicit activities easier to identify and analyse.

The analysis of permanent data storage is fundamental to computer forensics practice. There is very little finalised, and a lot still to be discovered in the conventional computer forensics methodology. This thesis contributes to formalisation and improved integrity of forensic handling of data storage by:

- formalising methods for data collection and analysis in NTFS (Microsoft file system) environment,
- presenting safe methodology for handling data backups in order to avoid information loss where Alternate Data Streams (ADS) are present,

- formalising methods of hiding and extracting hidden and encrypted data.

A significant contribution of this thesis is in the field of application of virtualisation, or simulation of the computer in the virtual environment created by the underlying hardware and software, to computer forensics practice. Computer systems are not easily analysed for forensic purpose, and it is demonstrated that virtualisation applied in computer forensics allows for more efficient and accurate identification and analysis of the evidence. A new method is proposed where two environments used in parallel can bring faster and verifiable results not dependent on proprietary, close source tools and may lead to gradual shift from commercial Windows software to open source software (OSS).

The final contribution of this thesis is systemising the body of knowledge in computer forensics, which is a necessary condition for it to become an established discipline of science. This systemisation led to design and development of tertiary curriculum in computer forensics illustrated here with a case study of computer forensics major for Bachelor of Computer Science at University of Western Sydney.

All genesis starts as an idea. A natural part of scientific research process is replacing previous assumptions, concepts, and practices with new ones which better approximate the truth. This thesis advances computer forensics body of knowledge in the areas which are crucial to further development of this discipline.

1 Introduction

“Forensis” vs. “forensics”:

The Latin word “forensis” [133] means: relating to the forum or a legal business conducted in public. It entered the English vocabulary in the 17th century as the term “forensics”.

The term “computer forensics” is singled out from a more general forensis category and represents a subset of that category [119]. The modern meaning of forensic is limited to the areas of legal and criminal investigations, where it refers to using a broad spectrum of sciences to answer questions of interest to the legal system [57].

Computer forensics is a new discipline of science, and not surprisingly its coming into being is hindered by the lack of unified theories, terminology, and professional standards. The emergence of computer forensics is closely coupled to slow development of calculating devices which rapidly accelerated in the second half of the 20th century when the first electronic computers were developed – see Figure 1.1.

The first electronic computers were built in the mid 1940s [74], and rapid development of this technology was soon followed by various computer related crimes: the first prosecuted case was recorded in Texas, USA in 1966 [51]. Traditionally computer crime has been defined as *“any criminal act committed via computer”* [27], but soon a more detailed classification developed to distinguish between criminal acts where a computer was used as a tool, and acts where a computer or computer systems themselves were abused or misused [152].

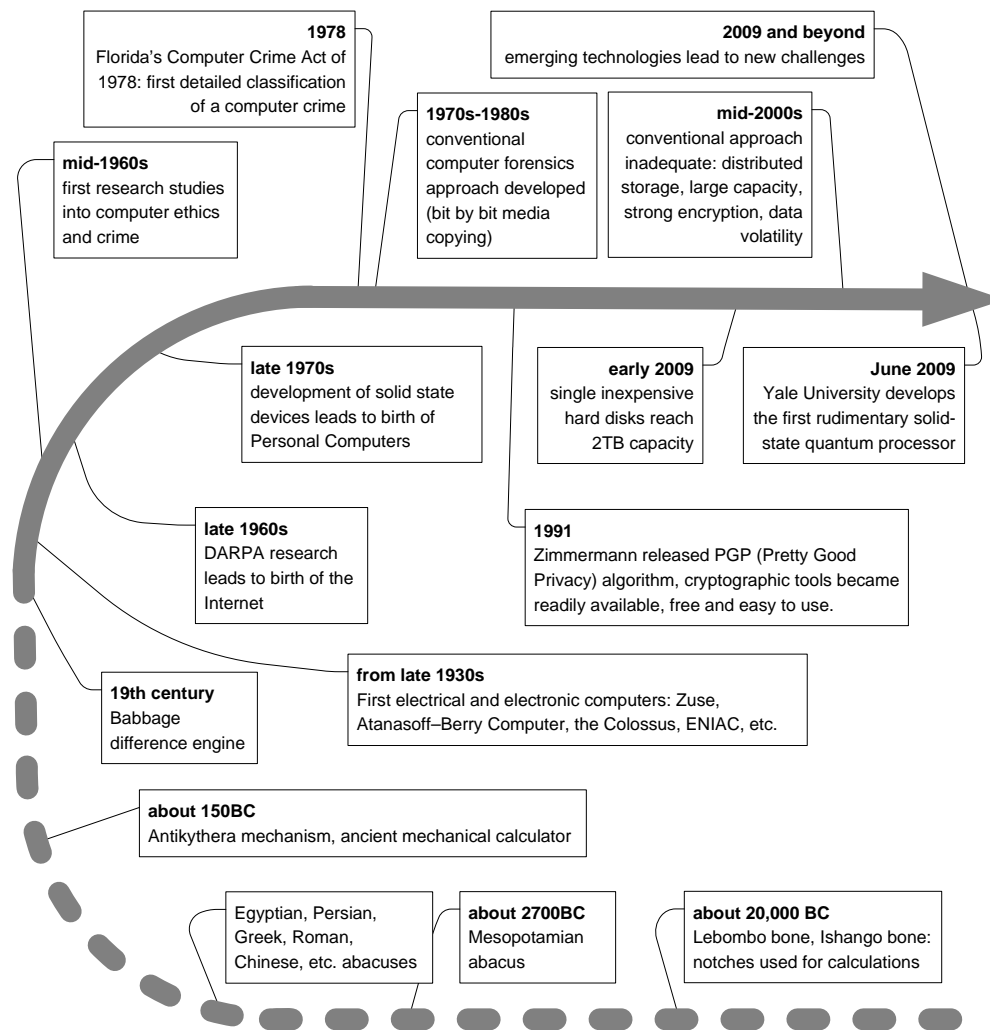


Figure 1-1 Emergence of Computing Machinery and Computer Forensics (time not to scale)

For early investigators it became obvious that if findings were to be useful as court evidence they had to comply with the same rules as conventional investigations [115]. It also became apparent that computer crime had features justifying a separate field of knowledge or discipline, commonly known as ‘computer forensics’, which focuses on the gathering of evidence from computers and computer networks. Computer forensics is a multidisciplinary field of forensic science encompassing computer science, computer

engineering, and law, and it aims to solve, document, and enable prosecution of computer crime (see Appendix A p.45¹).

At the time this thesis was written (mid-2009) the discipline of computer forensics was still in the process of emerging as a distinct body of knowledge. While the distinctive position of computer forensics might be generally accepted, its formal recognition as a field of forensic science has not yet eventuated. Even the terminology is not uniform: while the vast majority of professionals concerned with computer crime agree on the term ‘computer forensics’ it is not unusual to see different terms, for example ‘digital forensics’, the term which reflects a wider range of devices which may be used to store or transmit digital data (voice recorders, mobile phones, etc.) [78]. Occasionally the terminology used is misleading, for example ‘forensic computing’ [95], which is application of computer technology in forensics (for example to match fingerprint or DNA patterns).

There are two main challenges facing the discipline of computer forensics in 2009 and beyond: better formalisation of the existing conventional methods, and better understanding and use of emerging technologies. Technological progress has brought not only changes in the nature of crime, but also changes in the ways in which crimes are committed. Child sexual abuse [88] and terrorism [90] are not new, but they are examples of very serious crimes that may now be supported by computer technology [55]. The technology advances very fast, and computer crime advances with it. Computer forensics and the law need to proceed even faster to be a step ahead of computer crime. The best way to address this challenge is by providing high quality

¹ **Important note on page numbering in appendices:** all appendices are papers which were previously published, and they are paginated differently to this thesis. Thus each appendix page has two numbers: one as in the original publication, and the second one corresponding to the consecutive page of this thesis. The references to page numbers in this thesis correspond to the respective original paper pagination. For example Appendix A is a paper published on pages 43-59 of the *Journal of Information Science & Technology* JIST 5(3); thus a reference to the third page of the paper is quoted in the thesis as: “Appendix A p.45”

education, and thus creating a growing group of people with the knowledge and skills to further develop and practice this discipline professionally.

Computer forensics is a new discipline of science, and formalising it is a process that takes dedicated effort from committed professionals willing to champion the new ideas. The body of research presented in this thesis contributes in a material way to the genesis and advancement of computer forensics.

2 Thesis Outline

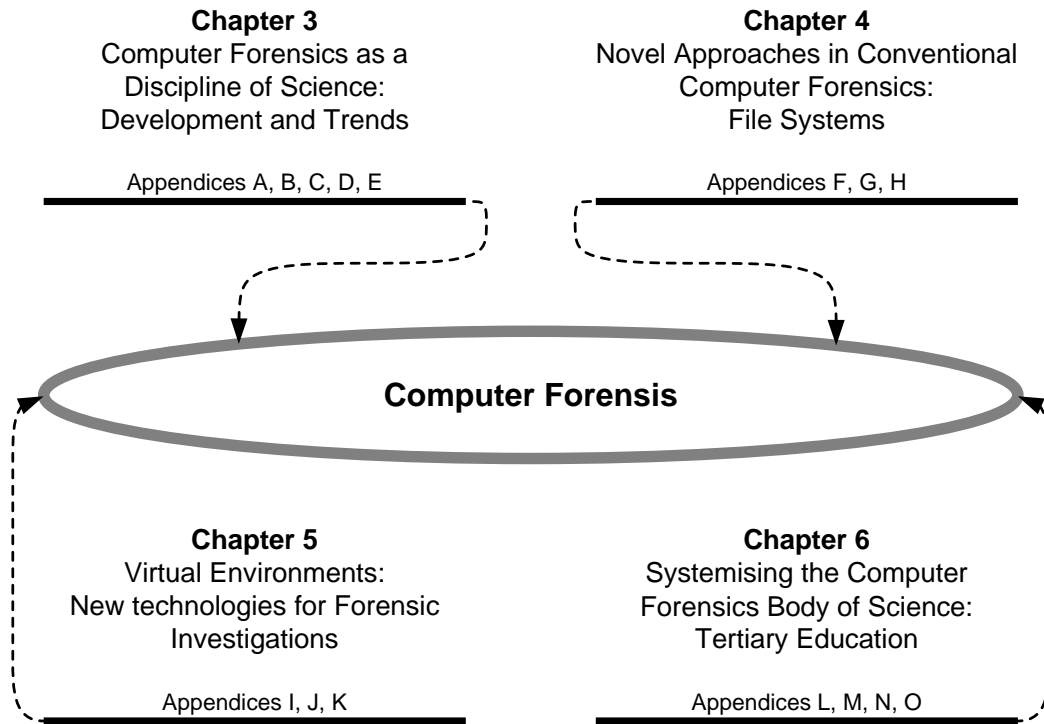


Figure 2-1 Thesis Structure

The body of this thesis is based on 15 publications which are included as Appendices A to O. These publications form a significant body of academic achievement within the discipline of computer forensics by contributing in a material way to its genesis and advancement. The thesis covers computer forensics emergence, development and trends, new discoveries and techniques in conventional methodology, application of virtualisation in forensics, and finally the design and development of computer forensics curricula at tertiary level. The thesis is organised into four main chapters:

Chapter 3 (Computer Forensics as a Discipline of Science: Development and Trends), based on the research work presented in Appendices A to E, identifies a number of new areas in computer forensics, and presents results of international collaboration in

research projects exploring three of these areas: forensic acquisition of computer memory, user data persistence in memory, and commonalities of computer forensics with information security (IS) audit.

Chapter 4 (Novel Approaches in Conventional Computer Forensics: File Systems), based on the research work presented in Appendices F to H, presents approaches and techniques enhancing the conventional computer forensics methodology. Formalised, forensically sound methods are described for data collection and analysis in NTFS (New Technology File System) and ADS (Alternate Data Streams), and extraction of EFS (Encrypted File System) files.

Chapter 5 (Virtual Environments: New Technologies for Forensic Investigation), based on the research work presented in Appendices I to K, connects two apparently unrelated areas: the virtualisation (understood here as simulation of the computer in the virtual environment created by the underlying hardware and software) and forensics. The chapter discusses the mechanism of booting and analysing a forensic image in a virtual environment. Further it proposes a method of using virtualisation to build a parallel multi-operating system environment where two systems, conventional and virtual, are used independently in the analysis phase of computer forensics investigations. The methodology is further developed and formalised, and the conclusion is reached that the proposed approach leads to a gradual shift from closed source software tools to open source software (OSS) tools.

Chapter 6 (Systemising the Computer Forensics Body of Knowledge: Tertiary Education), based on the research work presented in Appendices L to O, starts with the observation that a discipline of science must have a well defined, widely accepted and distinct body of knowledge which has to be sufficiently systemised to become a part of the tertiary curriculum. This in turn makes it possible to educate students, and create a

growing group of people who practice this discipline professionally and continue to strengthen its position. Unlike other fields of computer science, no guidelines or recommendations exist for computer forensics curricula, and the chapter presents how computer forensics curricula was designed, developed and implemented at tertiary level.

The degree of authorship of Derek Bem in the papers included in the appendices has been declared in the section **List of Appendices**.

3 Computer Forensics as a Discipline of Science: Development and Trends

(the corresponding research work has been presented in Appendices A to E)

We live in a society where virtually all records about individuals and organizations are stored and processed digitally on computer systems. Details of our business and private lives, what we do for entertainment, who are our friends, etc, are likely to be stored somewhere in electronic format [136]. This affects the foundations and all related mechanisms necessary for the 21st century global economy, government, business, banking, military, health, education, science, arts, and any other human pursuit [136]. Societies, businesses and individuals are becoming increasingly aware of this irreversible and prevailing phenomenon.

As every individual's and organization's dependency on computer technology increases, so does their vulnerability to information security breaches. Cyber attackers have almost unlimited opportunities to misappropriate or corrupt data repositories owned by individuals and organisations, and often little physical effort or time is required to commit an act which may carry serious consequences [156] . Computer security aims to prevent such incidents, but no security measure, electronic or physical, is totally reliable. Absolute security does not exist; we are always dealing with a balance of probabilities and risks. At the same time organisations are reluctant to disclose security failures in the fear that bad publicity may be even more damaging to their business than the original security breach [32].

The discipline of computer forensics is one of Australia's national research priorities, under the category Safeguarding Australia. The relevance of computer forensics is

further illustrated by a recent statement from US Federal Bureau of Investigations [53]:
“over fifty percent of the cases FBI now opens involve a computer”.

This chapter describes how the discipline of computer forensics emerged and evolved, and what are the current technical and organisational research directions.

3.1 Historical Background

Computers represent the fastest growing technology ever developed by humanity [41]. Unfortunately the rapid adoption of computers in all human activities was followed by various computer offences [60]. What makes computer crime unique is the apparent ease to commit an illegal act. One of the first research studies into computer ethics and crime was conducted in the mid-1960s by Donn Parker who noticed that: *“when people entered the computer center they left their ethics at the door”* [117]. Parker was also instrumental in assisting with one of the worlds first computer specific legal acts, Florida’s Computer Crime Act of 1978 [39], and he proposed the first detailed classification of a computer crime, suggesting four categories [116]:

1. *“a computer can be the object of a crime: when a computer is directly affected by a criminal act,*
2. *a computer can be the subject of a crime: when a computer is an environment in which the crime is committed,*
3. *a computer can be used as a tool for conducting or planning a crime,*
4. *the symbol of a computer itself can be used to intimidate or deceive*²*“.*

² As for example in: ‘you can trust this information, I obtained it from a friend who downloaded it from a secret FBI computer’.

The original classification has evolved over time, and today three main, non-trivial categories of computer crime are commonly accepted: criminal acts in which a computer is the object of the offence, the subject of the offence, or the tool for its commission [149]. A considerable amount of research literature into computer ethics [34] shows a disturbing property of the human psyche: people who would never consider committing a crime often do not have any scruples when a computer screen separates them from unethical or illegal activity [19]. Often illegal computer acts are committed by young people who see it as a socially acceptable, even a desirable activity which is elevating their peer status [147]. While teenage hackers are much easier to intercept than experienced, hardened criminals, prosecuting young offenders is difficult and may even bring the reverse effect of increasing their standing amongst friends [153].

In the late 1960s finding a specialist who knew how to handle computer crime was very difficult, but already groups of people known as “hackers” were using their technical knowledge and information gained by social engineering ³ to commit computer related criminal acts [93]. As incidents of high technology crime became more common, law enforcement bodies responded by establishing specialised laboratories and allocating additional personnel to fighting computer crime, the area which many see as *“the hottest emerging field in law enforcement”* [114].

In a rather unexpected development many early computer hackers gained the recognition of people who were their original targets, and in turn private enterprises and police forces used their services on many occasions to solve cases of computer crime, or to protect against future incidents [11]. One of the most noted examples is the world-famous controversial computer hacker Kevin Mitnick, who was arrested and convicted

³ The term “social engineering” when used in the field of computer security, computer forensics and hacking circles refers to manipulating people into divulging confidential information relating to computer system codes, system access, passwords, etc.

for computer hacking in 1995, and who is currently a well paid computer security consultant and author [103]. There are many similar cases where the services of computer specialists, or hobbyists with no formal qualifications and with an ethically questionable history of high technology related fraud were contracted by various organisations to help with handling security incidents [87].

In the early days computer forensics mainly involved collecting evidence from single systems; the volume of collected data was very low by today standards and the tools used were not specifically designed for the task [96]. While many good practices were established by people who handled computer crime evidence, no commonly agreed upon standards were developed. The U.S. Department of Justice published a series of guides relating to high technology crime, and those guides are considered to be the rules to follow when handling digital evidence [17]. Today most of the previously established rules are inadequate, and the major new challenge for the future of computer forensics is to modernise its methods and processes to maximise the yield of valuable evidence. The technology grows very fast, and formal documents often fail to address new technological developments. Less detailed and general descriptions of various procedures are not uncommon, for example one of the current U.S. Department of Justice guides [67] when referring to collecting data from computer storage advises to: *"Acquire the subject evidence to the examiner's storage device using the appropriate software and hardware tools"*. The term 'appropriate' is vague, and its interpretation has been left for an individual investigator to determine.

The conventional approach developed in the early days of computer forensics history was to make an exact image (bit by bit copy) of the storage media from a seized computer using appropriate tools, and analyse the acquired image later [149]. While this practice was easy to formalise in the 1970s or 1980s when the storage device capacity

was measured in single or tens of Megabytes [106], it became increasingly difficult to implement where the capacities of storage increased. By mid-2009 many technological breakthroughs [15], the latest of them perpendicular recording of data on magnetic media [106], caused a single hard disk capacity to reach 2TB with higher capacities on the horizon. Large capacity drives create many practical issues: copying data is slow, the error rate per single device is higher [22], and searching for the evidence is slower [42]. To illustrate the problem: a single 1TB disk can digitally store all world literature produced in one year [77]. Online Internet storage and storage virtualisation became common and easily accessible allowing for data to be kept on systems which are physically at other locations, and can be accessed as if they were local [45]. New, physically small devices (for example USB flash drives) became common and very inexpensive, and new data acquiring methodologies have to be developed to handle their unique properties (see Appendix J p.1).

In 2009 computer technology reached a stage when a conventional, mechanistic approach to collecting and analysing data was no longer sufficient. Computer forensics moved beyond the analysis of hard disk images, and older, conventional guides for computer crime responders [73] are no longer sufficient; they need to be updated to reflect technological progress. Because of complexity of acquiring and analysing computer related materials it is not unusual to see many people involved in the process at different stages. Large corporations typically prefer to handle high-technology incidents internally, and set up their own internal investigative units [83]. One of the main reasons for such costly solution is that unfortunately, as opposed to some other areas of forensics, there is no commonly accepted computer forensic certification, and thus no clear definition who is an expert [98].

3.2 Current Research Directions

In the last decade we observed a process of connections being formed between computer science, technology, crime detection and security (see Appendix A p.45). Computer forensics research directions must embrace approaches that will address emerging technological and organisational challenges. The technical area which needs to be better understood and formalised relates to collecting and analysing transient (or non-permanent) data. The organisational challenge is the need to develop a common, hybrid methodology combining computer forensics and computer audit tools.

3.2.1 Technical Example: Memory Forensics

The conventional approach to computer forensics requires making a copy of the storage media to be later analysed on a separate computer [79]. While this process allows the investigator to demonstrate in the court of law that the evidence was retained in the original, unchanged state, all “live” data is lost when the computer is powered down. Lost volatile data is likely to be a part of or even the only evidence of an incident.

The computer forensics methodology which was developed and formalised in the last decade mainly focused on the analysis of permanent data storage, largely omitting issues related to capturing volatile data and the content of physical memory. Today computer technology has reached a stage when a conventional, mechanistic approach to collecting and analysing data is no longer sufficient. Forensic investigation of physical memory can reveal unique facts about usage of a computer system not available elsewhere. Such data may contain clues regarding the recent use of a system that are not available anywhere else, and these clues may be crucial to successful prosecution [28].

Criminals have become more sophisticated, and the tools to securely erase or hide information are nowadays readily available and easy to use. Evidence of criminal activity kept in electronic form on non-volatile storage can be erased or hidden effectively and quickly [139]. The ability for investigators to obtain reliable information from live systems including contents of a computer's memory may be critical, for example it may allow the prosecution to produce evidence of the offending material having been loaded into the computer's memory at the same time when the accused was present on the premises, or it may allow investigators to capture evidence of terrorist communications [109].

Analysing a live system has certain disadvantages: observations can not be repeated (the investigator is largely a passive observer), as they are obtained within a certain time constraint, and they are always intrusive to some extent (see Appendix B p.131). Additionally it is difficult to eliminate the possibility that some crucial evidence remained hidden as a result of an undetected rootkit, which modified the system and blocked access to some utilities and processes without the knowledge of the investigator [148]. For a forensic investigation the analysis of a memory dump, or a memory snapshot, would be preferable. It would allow performing non-intrusive tests in a controlled laboratory environment, and to repeat the tests using different tools, thus leading to a very high level of confidence that the obtained results are correct.

Assuming the capability to obtain a snapshot of memory, the next step in memory forensics, interpretation of content, also presents a whole range of unresolved issues [53]. One such issue is that all current operating systems use the buddy system [81] to manage physical memory, and the age of deallocated pages has no bearing on the time of subsequent reallocation [76].

From a computer forensics point of view it is crucial to notice that an investigator can not directly or indirectly determine the age of copied memory pages. Thus any information extracted from memory can not be reliably dated: it may be relatively old, or it may be very recent. This presents a problem from a forensics point of view. As opposed to non-volatile data stored on magnetic media, pages in memory do not contain time stamps or any other direct information about their age. Research shows that the age distribution of these pages does not change significantly with the level of demand, and is surprisingly similar in modern Windows and Linux systems which preserve almost the same number of pages with user data (see Appendix C p.71).

Another issue relating to memory forensics is that no tool analysing memory dumps currently exists which would fully satisfy computer forensics requirements. Such a tool should be able to recreate the state of the system by interpreting the content of allocated memory, identify and interpret the content of unallocated memory, and finally identify currently and recently opened files by analysing memory resident fragments of files (so called file cache). While various native operating system tools exist, such as the Unix crash dump utility [140], as well as third-party software tools for Windows memory dump analysis [126], they are all of limited value in recreating the state of the system, because the memory image they analyse is not guaranteed to be consistent. Additionally none of these tools have any capability to analyse unallocated memory, which potentially contains information on past usage of the system, and none of them allow the identification and analyses of the file cache.

Capturing the content of the physical memory dump suitable for computer forensics analysis itself presents a serious challenge. Many commercial and open source utilities appear to provide the capability of capturing memory contents, for example the

common and ubiquitous dd [125], however the results are always far from complete and unsatisfactory because of two main problems (see Appendix B p.131):

1. Any software tool used for memory capture is itself loaded into the target system's memory when it executes, thus changing its contents. The memory dump happens in parallel with the execution of such a software tool, and the tool execution can not be isolated and excluded from the memory contents being captured.
2. The passage of time issue: the memory snapshot can not be done instantaneously, thus it can not capture complete and unaltered contents of local memory. The memory dump takes place in parallel with the execution of a operating system [37]. As the memory image is being captured, it is also being changed by other parallel processes, and pointers from already captured memory may point to data that has changed before it was saved. Similarly, data created later in memory that has already been saved may result in dangling references or unreachable objects. The serial and dynamic nature of memory traversal and capture excludes the possibility of capturing one coherent status. Instead we capture a series of incoherent, separate states.

There were various attempts to address the first issue. One of them was the hardware experimental device Tribble [36], a PCI ⁴ controller card which is based on the PCI bus capability to access memory directly using DMA (Direct Memory Access) [118] ⁵. However any method which requires special hardware to be installed prior to an incident in a computer is of very limited use to computer forensics.

⁴ Peripheral Component Interconnect (PCI) bus was introduced by Intel around 1993 for interconnecting chips, expansion boards, and processor/memory subsystems.

⁵ DMA allows to access computer system memory directly for reading and writing without using central processing unit (CPU).

Recent research from Princeton University [9] demonstrates that when computer RAM memory modules are rapidly frozen and removed from the original computer, data stored in RAM will persist for several minutes [65]. This discovery may lead to acquiring contents of memory, but currently this research is at a very early experimental stage, and its possible future application in computer forensics is uncertain.

The second issue, passage of time, has not been satisfactorily addressed by either software or hardware methods of memory acquisition. Taking a consistent image of memory can only be achieved if all system activity is stopped for the time required for the acquisition process to be completed. No current commercial operating system offers such functionality, however a “stop-the-world” approach is inherent to experimental orthogonally persistent operating systems, for example Napier88 [18] and other similar systems [20], [75]. Using techniques developed for such systems may lead to achieving a reliable technique to capture coherent status of memory contents.

The method to achieve this is to design and implement additional functionality in the kernel of the operating system which would, on demand, capture the complete virtual memory image comprising physical memory and the content of the paging and swap areas, and then store the images on an external device for offline analysis and interpretation (see Appendix B p.133). The software would have to be part of the kernel of the operating system in order to gain access to the lowest levels of control, for example setting the interrupt priority levels and ability to execute privileged instructions [126], and when activated it would execute the memory acquisition code. Once a complete and coherent memory image is captured it represents complete information about the system state at the time of the dump.

3.2.2 Organisational Example: Information System Audit

The Committee of National Security Systems (CNSS) defines an information system audit ⁶ as: *“Independent review and examination of records and activities to assess the adequacy of system controls, to ensure compliance with established policies and operational procedures, and to recommend necessary changes in controls, policies, or procedures”*. IS auditing and computer forensics interact with the same computing environment, yet they are usually considered to be two different disciplines with their own sets of tools and methodologies (see Appendix E p.110).

Today computer forensics is no longer limited to examining data collected from storage media but also includes live systems and volatile data, thus making the distinction between IS auditing and computer forensics less clear [108]. IS audit materials often mention forensics in a restricted sense, only in the context when a security specialist recreates what had happened to the system during a security incident [43]. While such process may indeed be a part of computer forensics investigations, findings would not be court admissible unless the investigator adheres to proper forensics methodology.

While both IS auditing and computer forensics have the same goal, IS auditors almost always work with live production systems which can not be stopped [157]. Looking at the five stages common to computer forensics and IS auditing, the following can be observed:

1. Purpose and planning: an IS audit is preceded by clear definition of requested activities, rights and obligations of the auditor. Typically a charter of engagement letter restricts the auditor’s activities in certain areas. A computer forensics purpose is finding evidence of criminal activity, and adherence to security

⁶ Usually referred to as “IS audit”.

guidelines is of little relevance. A court order which requests forensics typically does not limit which areas of the system the investigator is allowed to analyse.

2. Identification of potential sources of evidence: an IS auditor collects evidence from a much wider range of sources, which may include hard copies of various documents, written internal policies, etc. A computer forensics investigator is rarely asked to look at those additional sources. It is the court decision what to include as part of a total evidentiary material set, and if necessary the court may request other experts to look at the additional materials.
3. Acquisition and preservation: IS auditing uses methods which are rarely forensically sound. One of the techniques used is penetration testing [151]⁷, which would be totally unacceptable for a computer forensic investigator, as it is invasive and considerably changes the status of the computer system [21]. Another example is the use of write blockers during forensics acquisition [104]; these are very rarely used by IS auditors.
4. Analysis: data collected by an IS auditor may be analysed by various tools, not specifically designed or approved for IS auditing, and generally there are no tools prescribed for IS auditing. In contrast computer forensics tools need to be carefully selected for the task, or the investigator must be prepared to prove in the court of law that a tool used is forensically sound.
5. Presentation: the results of an IS audit are in the form of a report, which does not need to explain technological aspects of the investigations. A computer forensics presentation is required to explain relevant technological aspects of the

⁷ “Penetration testing is security testing in which evaluators attempt to circumvent the security features of a system based on their understanding of the system design and implementation. The purpose of penetration testing is to identify methods of gaining access to a system by using common tools and techniques used by attackers”.

discovery, it must be ready for the court of law presentation and scrutiny, it needs to comply with a series of rules or good practices guidelines, and finally it needs to include a comprehensive summary that is easy to understand by less technically oriented audience.

With computer crime on the rise IS audit often leads to formal investigations which in turn may be used in the court of law [24]. Any IS audit should be seen as potentially leading to forensic investigation. Thus a need to develop hybrid methodology (see Appendix D p. 7) combining computer forensics and audit tools becomes more urgent. If such hybrid tools and methods for one field were developed with full awareness of the requirements of the other field, both the IS audit and computer forensics would benefit. Some methods used in IS auditing need to be invasive, for example penetration testing. These conflicting requirements could be easily reconciled by equipping software with two modes of operation: normal and forensic, the forensic mode preserving the original data unmodified. Some software tools already offer a similar approach, for example X-Ways WinHex [56] offers a selectable forensic mode which preserves analysed evidence unchanged.

3.3 Summary of Contributions

This chapter makes a significant contribution to strengthening the position of computer forensics as a discipline of science by presenting a comprehensive overview of its history, current research directions, and by analysing its future challenges (see Appendix A). Two areas crucial to further development of the discipline of computer forensics were identified by the initial research. The first one is technical, the second one is organisational:

1. Memory forensics: forensic acquisition of memory (see Appendix B) and user data persistence in memory (see Appendix C),
2. Information security (IS) audit (see Appendices D and E).

The work presented in this chapter contributes significantly to memory forensics by developing a methodology to quantify how user data persists in physical computer memory (see Appendix C p.70). It was also proposed that some of the features offered by persistent systems could be built into conventional operating systems to make illicit activities easier to identify and analyse (see Appendix B p.133). A new technique has also been proposed for forensically sound acquisition of memory based on the persistence paradigm (see Appendix B p.134).

International interdisciplinary cooperation with European IS security audit specialists led to the realisation that the separate disciplines of IS auditing and computer forensics investigation interact with the same computing environment (see Appendix E p.113). There are sufficiently many similarities between both fields to justify developing common methodologies and tools. Our research resulted in proposing a hybrid methodology combining computer forensics and audit tools, which ensures that the evidence found in the course of an IS audit conforms to audit requirements and is forensically sound (see Appendix D p.7).

4 Novel Approaches in Conventional Computer Forensics: File Systems

(the corresponding research work has been presented in Appendices F to H)

Conventional computer forensics is mostly concerned with retrieval and analysis of data from magnetic and optical permanent storage devices like hard disk drives, CD ROMs, etc. The conventional approach is to make an image (bit by bit copy) of the storage media from a seized computer, and search it for relevant information. The methodology supporting this approach was developed and formalised in the early days of the computer forensics discipline, and duplication as well as preservation of digital evidence is very well documented in literature [29]. While duplication of magnetic storage media is a relatively straightforward task [149], analysing the copied storage is not a process which can be fully automated and it requires skills at a high level and an in-depth understanding of file system structures – see Figure 4.1. Acquiring this expertise is not easy: literature is often obsolete and covers only file organization principles and algorithms [142].

Each file system is based on a different design philosophy and uses different internal structures, so it is not possible to develop one methodology which could be applied to all file system structures. What follows is that a computer forensics investigator requires a detailed understanding of the internal operation of a file system being analysed. When a new file system is introduced, specific information about how it is built and organised is needed immediately. In 1988 Microsoft embarked on a large project to create a completely new family of operating systems [48] with a new file system architecture. The first commercial product was released in 1993 as Windows NT 3.1, and all subsequent releases including the most recent, Windows 7, are based on the original NT code base. As part of the project the aging FAT (File Allocation Table) file system architecture

designed in the late 1970s was replaced by NTFS (New Technology File System, or NT File System) which since then became the native file system format for all versions of Windows.

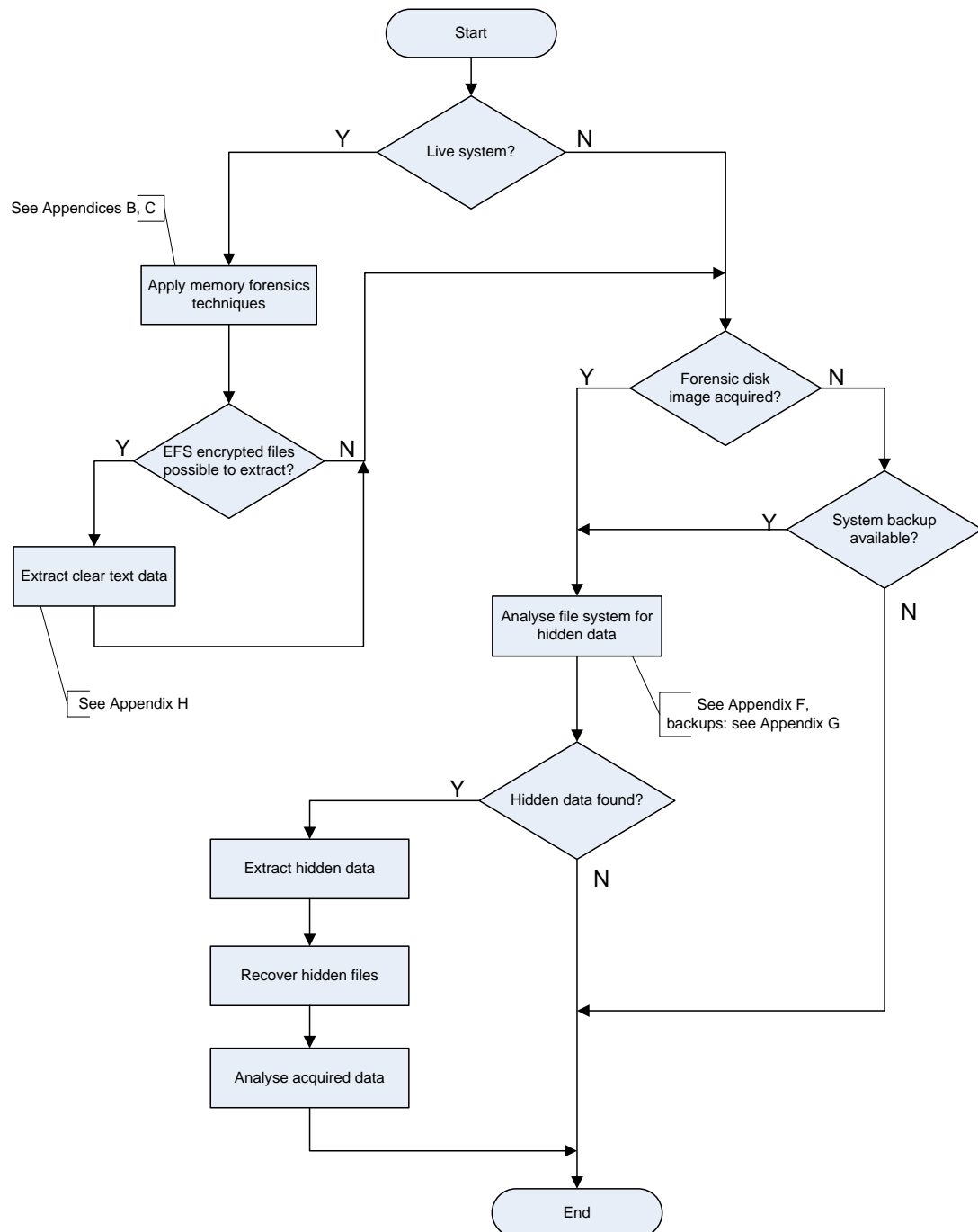


Figure 4-1 Computer Forensics Methodology and Processes

NTFS is advanced and complex, and out of all file systems currently in use it is by far the most widely used in both home and commercial environments. Yet its detailed structure and operation has not been released into the public domain by Microsoft. There is still a lot to be discovered in the conventional computer forensics methodologies as applied to investigations of NTFS file structures.

4.1 Investigating NTFS File System Images

In the early 1990s, soon after Windows NT was introduced, Microsoft published a series of books explaining the system's operation, but none of them fully documented the internals. For example the book "Inside the Windows NT File System" [47] released in 1994 by Microsoft Press offered a good conceptual description for users and administrators, but it lacked technical depth and details. This lack of depth in the documentation created problems in two areas: interoperability with the Linux family of systems and computer forensics investigations.

In a network environment computers running different operating systems must be able to read and write files between different machines, and Linux based systems require the capability to read and write Windows NTFS files [70]. In a non-network environment it is common to see a single computer in multi-boot configuration with Linux and Windows installed, and again cross platform file compatibility is highly desirable. Various Linux related projects focused on reverse engineering the NTFS structure. This resulted in publishing comprehensive NT system level documentation and the release of various tools [127]. In 2003 one of these projects, Captive [84], resulted in developing the first stable Linux read/write access to NTFS disk drives. Similar solutions soon followed, and today the most commonly used read/write NTFS driver for Linux is NTFS-3G [7].

The work of the Linux community concentrated on providing a Linux – NTFS interface, and it greatly contributed to a better understanding of the NT internals. However this work did not address the needs of computer forensics experts: it did not provide specifics about how to analyse images of the NTFS file systems, and it did not analyse various aspects of data hiding in the unique NTFS file system data structures. Understanding how information can be hidden in the file system and how to discover and extract it is one of the main tasks in computer forensics [12] – see Figure 4.1.

Typically when analysing computer system structures [89] one would expect to find some empty, unused fields reserved for future use or fields left unused for clarity of design ⁸. Both of those reasons for leaving the gaps are common, and create a basis for hiding information. Only some methods are effective in the long term, and thus of interest for computer forensics. Criteria for those methods are described by Provos and Honeyman [122]:

- No errors should be returned by standard system tools ⁹ (*this necessitates the development of special tools for the computer forensics field*).
- Probability of hidden data being overwritten during normal system operation should be very low.
- Hidden data can not be revealed using standard system GUI tools ¹⁰ (*again, a computer forensic analyst needs access to special tools*).

⁸ For example a byte (16 bits) may be allocated to map certain system conditions, but only 4 such conditions are allowed. For clarity the remaining 12 bits of the byte are left unallocated, as opposed to allocating them to map different, logically unconnected system conditions.

⁹ For example chkdsk utility.

¹⁰ For example Windows Explorer.

- A reasonable amount of data can be hidden ¹¹ *(the trivial case of a user hiding a very small amount of information, for example a single character, is trivial and of no practical interest to computer forensics).*

In NTFS every object is a file which includes metadata ¹² with relatively complex structures [110]. The most important structure of an NTFS volume is the Master File Table (MFT) implemented as an array of records (also called attributes) [126]. The structure of MFT creates a possibility for effective, non-trivial methods of hiding data in \$DATA, \$BadClus attributes and \$Boot file (see Appendix F p.212). Moreover NTFS is not sensitive to certain changes within the metadata structures, and does not warn if a file has additional attributes which are not necessary [35], thus increasing effectiveness of data hiding.

4.2 Investigating NTFS File System Backups for Hidden Data

The NTFS file system metadata structure \$DATA attribute provides the greatest scope for effective data hiding. This attribute supports a unique feature introduced by NTFS: Alternate Data Streams (ADS ¹³). The main motivation for introducing ADS in the early 1990s (see Appendix G p.449) was to provide better interoperability between Windows NT servers and Macintosh clients [134] which were using Hierarchical File System (HFS) to access Windows NT Resources [63]. There are also other uses of ADS: they can provide additional descriptions for files and folders which may include subject, author, keywords, descriptive comments, thumbnail previews of images, etc. Each

¹¹ It is always possible to effectively hide a very small amount of data (for example a single character) within any operating system structure.

¹² Metadata defines the structure of the file system itself and may provide additional information relating to the file, for example ownership, protection, timestamps or descriptive comments.

¹³ Microsoft also uses the acronym “ADS” in the same Windows environment in relation to Automated Deployment Services, with no connection to Alternate Data Streams.

Master File Table (MFT) record representing one file can have one or more \$DATA attributes; the additional \$DATA attributes are known as alternate data streams. Many basic, simple to use utilities exist which allow users to embed a file as an ADS of another file, thus effectively hiding it from common system utilities [33].

A computer forensics investigator needs to be aware of the mechanism of hiding information within ADS, and methods to search for such hidden data. While there are utilities which allow finding and extracting ADS, none of them is able to distinguish between the ADS used for legitimate purposes and the ADS used to hide information, thus they require the analysis of a huge volume of largely irrelevant data.

Forensic acquisition of storage devices involves making an exact, bit by bit copy of the original media using one of many, often free, imaging tools [125]. A forensic copy contains exactly the same data as the original [112]. File system backup software [50] is usually not considered to be a forensically sound tool, as typically it does not produce a true, bit by bit copy of the source. While some backup tools provide the option to create a full hard disk image to allow a so called “bare metal restore” to a new, empty disk, compliance of such software with forensics requirements can never be guaranteed, and would have to be verified [143]. There may be circumstances where system backups are used as additional sources of evidence and an investigator needs to understand the peculiarities of the specific software which was used to create the backup files being analysed.

Many free, open source and commercial backup packages exist; such a utility is also bundled with Microsoft Windows [14]. The intuitive assumption is that backup software should be able to copy and subsequently if needed to restore full information from a storage device. This assumption is however incorrect. If the option to create a full copy is selected, various free and commercial file backup tools behave differently when

handling structures unique to the NTFS file system. Specifically, files and folders with ADS (see Appendix G p.450) are not handled uniformly. Some backup tools are non-ADS aware, some are able to handle ADS only when both source and target media are NTFS formatted, and some can backup full ADS contents to a non-NTFS media and later restore it, but only if a target is NTFS formatted ¹⁴.

The following observations are of crucial importance from a computer forensics investigations point of view:

- When a backup is created and stored on non-NTFS formatted media, only some tools are able to backup the ADS part of the NTFS file structure.
- Backup software very rarely offers the option of extracting and separating the ADS part of the structure.
- Backup reporting logs are aimed at the system administrator, and typically are of limited forensics value.

No currently available backup software can be considered to be fully ADS compatible and no such software can produce comprehensive reports required by computer forensic investigators. As system backups can be produced with a wide variety of tools a computer investigator needs to be aware of their peculiarities. It is unavoidable that some information may be irretrievably lost, and it is unrealistic to expect that a computer forensics tool may be created to fully automate the process of handling forensic investigations of backups.

¹⁴ Appendix G proposes a formal classification of backup software into five groups: Class 0 – Class 4. The class allocated depends on how a software package handles ADS.

4.3 Live Investigations of NTFS File System Encrypted Data

For millennia obfuscating information and hiding it by various means was common, and the growth of computing only strengthened the demand for more sophisticated and easier to use methods [82]. Hiding data using the ADS property of the NTFS file system requires certain knowledge of the system's operation and special tools. With Windows 2000 Microsoft introduced an easier to use data hiding mechanism: Encrypting File System (EFS). The EFS is the NTFS specific technique to encrypt data which does not require an in-depth knowledge of computers, and it can be used even by an inexperienced user by simply selecting a box in the Properties menu of a file [54]. Encryption and decryption is handled by the system transparently, and a user handles the encrypted files in the same manner as clear text files [101].

The EFS runs as an integrated system service [26], using symmetric key encryption and public key technology [92]. The mechanism of protecting data by four layers of encryption (see Appendix H p.148) does not allow automated decrypting of data if it is copied to another Windows system. Subsequently, the EFS functionality is provided locally and only for NTFS local media.

The above mechanism has two crucial implications in the contents of computer forensics:

- When the owner of a system is logged on, copying an EFS file to a different file system, for example FAT32, saves it on the target as clear text.
- A forensic image of the NTFS disk contains EFS encrypted files which are practically impossible to decrypt, thus once the system has been powered off, information contained in the encrypted files is irretrievable [3].

The above observations led to the development of a new, forensically sound methodology (see Appendix H p.149 and Figure 4.1) to extract the EFS encrypted files from live systems. The methodology is based on further observation that the only window of opportunity to capture the EFS encrypted files in clear text is to copy those files to non-NTFS storage when access to a live system is still available. If files are copied from a NTFS file system to a FAT32 file system the encryption is not carried across, and the files are stored on the FAT32 media in clear text. To assure that potential evidence is not changed by the process, software tools used should not require installation on the system being analysed. All software utilities are prepared and tested in advance and stored on a transportable disk drive which is connected to the investigated computer. No tools from the investigated system are used as they may be compromised, for example by a rootkit [148], and may trigger an unexpected response or even damage the system and invalidate the collected material [91].

The investigator working in a live computing environment needs to be aware of the Locard's Exchange Principle, well known to all crime investigators [49]: *"Anyone or anything entering a crime scene takes something of the scene with them, or leaves something of themselves behind when they depart"* [128]. Following the Locard's Principle requires making minimal changes to a live system being investigated. The forensic soundness of the methodology was proved by verifying that the hash signatures of all EFS encrypted files were not changed as a result of copying them to a external USB disk. The one way hash function is commonly accepted as a unique fingerprint identifying a file; if the hash signature of a file made at different points of time is identical, the file was not altered [130].

The methodology of extracting EFS encrypted files as clear text demonstrates that no single technique can guarantee successful acquisition of all data from a system being analysed – all techniques complement each other. For example acquiring and analysing

the physical RAM memory [31] is not sufficient to extract the EFS encrypted files. While the memory image may contain encryption keys in clear text, they are a random combination of characters, and identifying them cannot be guaranteed, unless their location and length are known. An additional difficulty when analysing contents of memory is that RAM contents is dynamically changing, and only data in active use at the time of memory image acquisition or shortly before can be acquired (see Appendix C p.68).

Demonstrating that it is possible to extract the EFS encrypted files from live systems while preserving the original data presents a very strong argument for accepting live system techniques as indispensable to complete forensic analysis of a system.

4.4 Summary of Contributions

This chapter makes a significant contribution to research into formalisation of methods for data collection and analysis in NTFS. The findings presented here enhanced the current status of computer forensics as a formal discipline of science.

NTFS is complex and by far the most common file system with unique properties, considerable flexibility, and many ways to hide data. Many software utilities exist which allow the internal investigation of an NTFS file system, but unfortunately they are not particularly suitable as tools for computer forensic investigators. They require a high level of understanding of system operation, and their use is time consuming, in turn decreasing the effectiveness of the discovery process.

The major contribution of the research presented here is that it developed formalised methods that can be used to hide data in the NTFS file system, and the techniques that can be applied to detect and recover the hidden data (see Appendix F).

As an exemplar of this methodology, a number of system backup utilities were examined and the backup software capability to save and restore Alternate Data Streams (ADS) was analysed. Depending on ADS awareness it was proposed to classify such software into five classes, Class 0 to Class 4 (see Appendix G). For a computer forensic investigator it is crucial to be aware of which class of backup is being analysed, as different classes retain varying amounts of information about ADS during backup and restore, and often lose data. A safe methodology was designed for handling backups in order to avoid the information loss, which is relevant in forensic and general cases.

A full understanding of the mechanism of hiding and extracting hidden data resulted in creating a methodology of extraction for EFS (Encrypted File System) files. A comprehensive, formalised procedure for extracting the EFS encrypted files in a forensically sound way was created and described (see Appendix H). It was demonstrated that working with a live system allows the investigator to extract information which cannot be discovered using any other known forensic technique.

5 Virtual Environments: New Technologies For Forensic Investigation

(the corresponding research work has been presented in Appendices I to K)

The computer forensics investigative process comprises a number of steps which are very similar to investigative methodologies used in other rigorous, non-computer related crime investigations [113]. What makes computer forensics unique is that it requires in-depth knowledge of many topics belonging to what could be very broadly described as a field of computing; tools and methods may come from various, sometimes apparently unrelated areas, for example virtual systems ¹⁵.

The stages of the computer forensics investigative process varies in details between jurisdictions, however the main steps are relevant universally (see Appendix I p.2). The nature of computer crime and to a large extent the legal responses to it are similar around the world. Each local jurisdiction is guided by different detailed rules and mechanisms, however the conventional computer forensics process can be encapsulated in four key phases [85]: access, acquire, analyse and report. Each of these four phases can be further subdivided and formalised, for example [91] by observing that the methodology has to include a pre-incident preparation stage, detection of incidents, an initial response, formulation of response strategy, investigation of the incident, reporting, and resolution. A new, enhanced workflow based on virtual environments (see Appendix I p.4) has been proposed, and is described in the next sections.

As the complexity of computing applications grows, there is an increased misunderstanding of what computers really are and what they are capable of [141]. In the last phase of computer forensic investigations the investigator may be called to

¹⁵ Virtualisation is understood here as a simulation of the computer in the virtual environment created by the underlying hardware and software.

present the results to a non-technical audience. This is a difficult task, as highly skilled technicians often have poor skills of communicating their findings in a clear, easy to understand way. The problem was summarised by the U.S. Department of Justice [64]: *“Some judges, attorneys, and jurors may harbor doubts about the reliability and significance of digital evidence. To prevent misunderstandings at trial, concepts must be explained in simple terms with carefully selected analogies and visual aids”*.

Virtualisation [135] is one tool which can help to analyse the computer related evidence, and assist in presenting the findings in a court environment.

5.1 Forensic Image in a Virtual Environment

Typically in the process of evidence acquisition a full bit by bit image of the investigated system is made using Linux dd [35], or a functionally similar tool. If the imaged disk contains a full operating system, booting it to analyse the user activity and contents of the system appears to be a reasonable approach, and is often recommended [111]. Usually the original machine is not available, thus moving the acquired image to a forensic workstation for analysis and booting it as a guest under virtual machine is the next logical step. Indeed, if an investigator succeeded in booting the acquired system on his workstation he would be able to see exactly what the original owner saw, and by using their tools they would be able to analyse the system. Unfortunately while conceptually simple, there are many problems with this approach (see Appendix I p.3).

Most computer forensics investigators deal with Windows operating systems, simply because it is the most commercially successful computing platform [27]. Let us consider a case when the original machine being investigated runs a Windows operating system. As a anti-piracy measure Microsoft over the years implemented system level

mechanisms which prevent copying of the full operating system between different computers [99]. From the computer forensics point of view this means that the system could be easily booted only on hardware identical to the hardware it was originally acquired from, which is impractical. Attempts to boot on different hardware creates a series of issues, the severity of which depend on what differences Windows will detect. If they are relatively small, Windows may request new hardware drivers to be installed. If the differences are considerable, for example different CPU architectures, Windows will fail to boot and generate a kernel-level error.

Similar problems exist when attempting to boot the system in a virtual machine environment. Virtual machines emulate a very limited range of hardware; they were not created to duplicate a wide variety of possible hardware configurations. An additional problem arises because of Microsoft Product Activation policy [100] built into all versions of Windows. Microsoft uses different activation approaches for computers released by manufacturers (Original Equipment Manufacturer, or OEM), purchased in a retail store, or volume licensed by enterprise customers.

For a computer forensics investigator moving a full system to different hardware creates a difficult to solve problem. As a result of the product activation mechanism the image may simply fail to boot on different hardware.

During the product activation Windows checks ten areas of the computer system and creates an eight byte long hash number corresponding to the configuration. The check uses only some selected hardware components, such as a part of the CPU serial number, IDE and SCSI adapters, 'dockable' flag, BIOS ID, etc. The resulting eight byte hardware hash value and Windows serial number are used by Microsoft to activate Windows online or on the phone. Once the installation has been activated, on each login Windows checks that the hardware is similar to the hardware that it was activated on; if

the hardware is detected as “substantially different”, reactivation is required. The precise algorithm used to determine when the hardware is “substantially” different as opposed to “the same or similar” is not published by Microsoft for obvious reasons.

The above mechanism allows users to replace some computer components without being forced to reactivate the installation, for example video adapters can be changed to a different brand and type. If the motherboard is changed to a new type with a different BIOS and different integrated components this would unfortunately cause Windows to create sufficiently different hardware hash error requiring reactivation. This means that a computer forensics technician can not be certain what will happen when the image is transferred to different hardware and a boot is attempted – while it may sometimes work, often it will fail, and thus it can not be recommended as a formalised procedure.

If an image is booted on the same hardware as the hardware that it was acquired from Windows is still accessing many files during the boot process. For example during a normal, error free boot the Windows NT system accesses over 500 files, changing their contents and time stamps [85].

If an image is booted on different hardware Windows will request a series of new drivers to be installed thus making even more changes to the original image. While other operating systems (for example Linux) typically allow movement of the system image between machines without protections similar to Microsoft activation policy, it still requires a series of drivers when booted on a different (real or virtual) platform. In effect the forensic soundness of such an approach is questionable due to considerable changes to the original image, and may possibly render the results as being inadmissible in court. It definitely clashes with the fundamental rule of forensics requiring “minimal handling of the original” [95].

While booting of the forensic image in a virtual machine environment is a conceptually attractive option, in reality it can not be fully trusted as a reliable and easy to duplicate methodology. If the booting fails, or does not bring expected results a different approach is called for, as described in the next section.

5.2 Using Virtualisation to Improve the Analysis Process

In forensics there is always a danger of incorrect handling of the acquired materials which in turn can invalidate the whole process of forensics. The Australian Institute of Criminology guide [95] recommends two rules for the process of analysing computer evidence: “minimal handling of the original” and “do not exceed your knowledge”. This requires careful handling of the evidence, formalising and documenting all steps, and consulting a more experienced specialist if and when required – such an approach makes the process slow. At the same time there always existed a conflicting expectation to produce some tangible result within reasonable time [150], [149].

Using the best tool for a given task is of paramount importance in the computer forensics field, where the investigator has to handle a wider range of issues than in any other computer related field. In the process of investigating an incident it is not unusual to look into mechanisms of various operating systems, hardware ranging from small devices to large servers, distributed data storage, networks, etc. [146]. No software tool is perfect and universal, and some tools are more suitable for some tasks – for example it is well known that Linux based tools are generally better suited to handle network related investigations [53], while investigating the internal structure of NTFS file system is better handled by Windows based tools [38].

However it should be stressed that a simple methodology based on using Linux tools to analyse one type of data (for example network activity traces) and to use Windows based tools to analyse a different set of data (for example NTFS structures) is not necessarily correct in every specific case. Which tool (and which operating system platform) is best to handle a specific task depends on the specific tool, the specific task at hand, and the individual skill level of the investigator. Both Windows and Linux platforms and tools can and should complement each other at all levels [105].

Generally, using Linux tools to forensically analyse Linux and Windows images has many advantages and is very well documented [61]. At the same time computer forensics investigators are much more likely to come with Windows experience, and most commercial tools for forensic work are Windows based [111]. To illustrate with two examples: the most commonly used set of computer forensics tools is EnCase [30] from Guidance Software, and another tool, InfinaDyne's CD/DVD Inspector [46], is the only fully integrated tool to examine and analyse CD and DVD evidence – both are Windows based.

It would be counterproductive to abandon one computing platform in preference to the other, none is “the best” for computer forensics work: depending on a specific task at hand one of them may be more accurate, faster and easier to use.

Because of technical complexities and lack of detailed procedures courts often request that a finding should be verified using two different tools [137]. It strengthens the findings of an investigator if such confirmation of conclusions comes not only from two different tools, but from tools used in different operating system environments.

A methodology has been proposed (see Appendix I p.4) where two (or more) different computing environments are used to analyse the same evidence. The selection of

specific computing environments depends on the expertise of forensics personnel, the specific task at hand, and availability of tools. For one task it may be Sun Solaris and Windows 7, for another task it may be Linux Fedora and Windows XP. Because a virtual machine environment is used changing computing platform and tools is fast and easy, and it can be done using one computer only, provided it is sufficiently powerful and properly configured. Using one workstation has an additional benefit of increasing security (all work is physically kept on one machine) and simplifies backing up of the work.

An exemplar of this method has been presented as a specific scenario (see Appendix I p. 5) where the same disk image can be accessed from two different operating systems and different tools can be used depending on a task. Such a setup gives access to a wider range of tools, and better flexibility in deciding which tool to use for a specific task. Therefore it results in improved accuracy of the computer evidence analysis process, and it shortens the total time required for analysis. The following main steps are suggested (for the detailed workflow see Appendix I p.4):

- selected distribution of Linux is installed as host on a investigator computer, and preferred virtual machine software is installed and configured on the host,
- selected Windows editions (XP, Vista, 7, etc.) are installed as guest operating systems (one or more guests can be installed as needed),
- the acquired image is mounted on the Linux host.

Linux systems always offered an easy method of mounting and dismounting physical disks, partitions, folders and storage devices [132], and when a forensic image is mounted under a Linux host it is easy to divide it as required into logically separate parts, and make such parts available at will to the host Linux, the guest Windows or

both, also as read-only mounts [45]. This offers the investigator high flexibility. A single, large image can be divided into sections which are logically connected: for example a group of folders and files relating to accounting can be temporarily mounted in an 'accounting' folder, a group of folders and files containing games can be mounted as a 'games' folder, etc. When the original image is divided in such a way the investigator can concentrate on analysing a series of separate parts, temporarily excluding other parts of the image from the process, and thus search and extract data faster.

The process can be further improved if we assume two investigators (or two teams of investigators), less experienced and more experienced, who are allocated to the same task of analysing hard disk data (see Appendix I p.4 and Appendix J p.3). This is similar to the roles of CNF Technician (Computer and Network Forensic Technician) and CNF Professional (Computer and Network Forensic Professional) in the classification proposed by Yasinsac et al [159]. Any attempt to boot the image and to initially investigate it [120] is done by the less experienced investigator, who can search it for all details relevant to the investigation. While such approach is likely to invalidate the integrity of the acquired image, this is of no consequence to the investigation. The initial findings are passed to the senior investigator, who then uses the original, unmodified disk image and proper computer forensics tools and techniques to confirm the findings. Such an approach allows the team to use a less qualified person to do a preliminary, less rigorous analysis of the evidence which may bring results faster and more efficiently without invalidating the final findings.

Utilising a virtual machine environment has the unique advantage of not irreversibly closing avenues of investigations: every investigator working in the Linux host environment with Windows running as a guest would still have full access to the familiar Windows set of tools. Moreover no technician would be forced to use any environment

in preference to another; every investigator is given a free choice to select any tool on any computing platform, and to discover the advantages of new tools. The same disk image which is analysed from a Windows guest with Windows based tools is at the same time mounted on the Linux host and can be accessed with the native Linux tools. It is natural that an investigator in such a parallel Windows/Linux environment would start trying tools from another computing platform, and they will benefit from the new, powerful utilities and techniques.

5.3 Easing Reliance on Closed Source Software

Easing reliance on the expensive closed source software is of particular significance during the economic downturn of 2009 as shown in the March 2009 International Data Corporation (IDC) [4] survey. IDC conducted one of the most comprehensive research projects in the area of Linux adoption by large institutions (100 or more employees): they collected data from 330 organizations in different countries. The survey [58] showed two interesting trends:

- “the current economic crisis will also have a net positive impact on the use of virtualization software”
- “half of the survey participants stated that moving to virtualization is accelerating their adoption of Linux”

This thesis further supports the IDC findings: it summarises research work conducted in the last three years and confirms the soundness of adopting a Linux environment in the computer forensics field (Appendix K p.6). Best results are achieved when two environments, Linux and Windows, are not used exclusively, but in parallel.

Typically in the process of a computer forensic investigation the Windows environment and Windows based software tools are used [78]. This reliance on closed source software has certain advantages [30] and disadvantages [72], and traditionally created two groups of specialists: one working in Windows, another one in Unix/Linux environments.

Computer forensics is one of the areas which clearly demonstrate that to be more efficient a new type of specialist is required: one who finds it easy to switch between different environments at will. The methodology of using virtual environments described above has a significant advantage over the older, conventional approach [39]: it eases reliance on closed source (Windows based) tools. The setup does not require any drastic change in the existing methodologies: it introduces a parallel, Linux based environment and tools which an investigator may use to check or complement the findings obtained with Windows tools.

The specific advantages of using virtual environments in computer forensics application are:

- Cost savings: typically Windows software tools currently used in computer forensics investigations cost US\$5,000 or more with strict licensing limits – by contrast open source tools are free.
- Skill expansion: if skilled investigators are offered an easy to use dual operating system environment, they will start using the best tools from both platforms, and soon will expand their skills beyond Windows based tools. This is invaluable when the need arises to analyse Linux incidents.
- Verification of findings: many researchers stress this as a very important advantage of open source tools. It is clear that when using closed source tools

one can not completely prove what was done, to what extent the procedure used was forensically non-intrusive, or that the original files were not affected. While in most investigations such proof may not be required, in critical cases being unable to demonstrate a full understanding of the tool may lead to challenging the validity of the findings.

5.4 Summary of Contributions

An in-depth understanding of two apparently unrelated areas, virtual environments and forensics, created a unique perspective which led to using virtualisation for forensics more efficiently and accurately. Virtualisation was not created with computer forensics as the main application, and while it offers a very powerful set of tools they are aimed mainly at server applications [107]. The innovative research presented in Appendices I, J and K shows that virtualisation allows for more efficient identification and analysis of the evidence, and significantly enhances the clarity of presentation of findings in a courtroom situation.

A new approach was proposed (see Appendix I) where two environments, conventional and virtual, are used independently and in parallel. This approach can considerably shorten the time of the analysis phase of computer forensics investigation and it also allows for better utilisation of less qualified personnel. A specific instance of this methodology was presented as a formalised method of analysing USB flash drives in virtual environments (see Appendix J).

An approach was proposed leading to a gradual shift from closed source software tools to open source software (OSS) tools. The research (see Appendix K) demonstrates a more effective way of using Virtual Machines in Linux environments by complementing

the conventional techniques, and providing verifiable results not dependent on the tools used.

The method of using virtualisation to build parallel multi-operating system environments proposed in this chapter is ready for commercial application. Many robust, stable and free Linux distributions are available, for example openSUSE [8], which have virtualisation integrated at the system level.

6 Systemising the Computer Forensics Body of Knowledge: Tertiary Education

(the corresponding research work has been presented in Appendices L to O)

The Encyclopaedia Britannica defines science as “*any system of knowledge that is concerned with the physical world and its phenomena and that entails unbiased observations and systematic experimentation. In general, a science involves a pursuit of knowledge covering general truths or the operations of fundamental laws*” [1]. There are many studies dedicated to the history and philosophy of science [25] leading to a conceptual distinction between what the above definition calls “observations and experimentation” and a proper discipline of science. The process of a body of knowledge emerging as a science implies progress and gradual, consistent elimination of errors in methodology. What follows is replacement of the previous assumptions, concepts, and practices with new ones which better approximate the truth.

For a body of knowledge to become a science it not only needs to be unique in nature but it also needs to grow in size to achieve impact and legitimacy in the scientific community and society at large – see Figure 6.1. Forming a new discipline is a process that takes time, a great deal of effort, and participation of many dedicated like-minded people. This process does not always succeed: two exemplars of bodies of knowledge which emerged as well established areas of science are physics and computer science. An exemplar of a discipline which stagnated and failed to develop further is the study of formal organisations [23]. A closer examination of the past can provide valuable insight into what is needed for computer forensics to succeed and to emerge as a well established, distinct discipline of science.

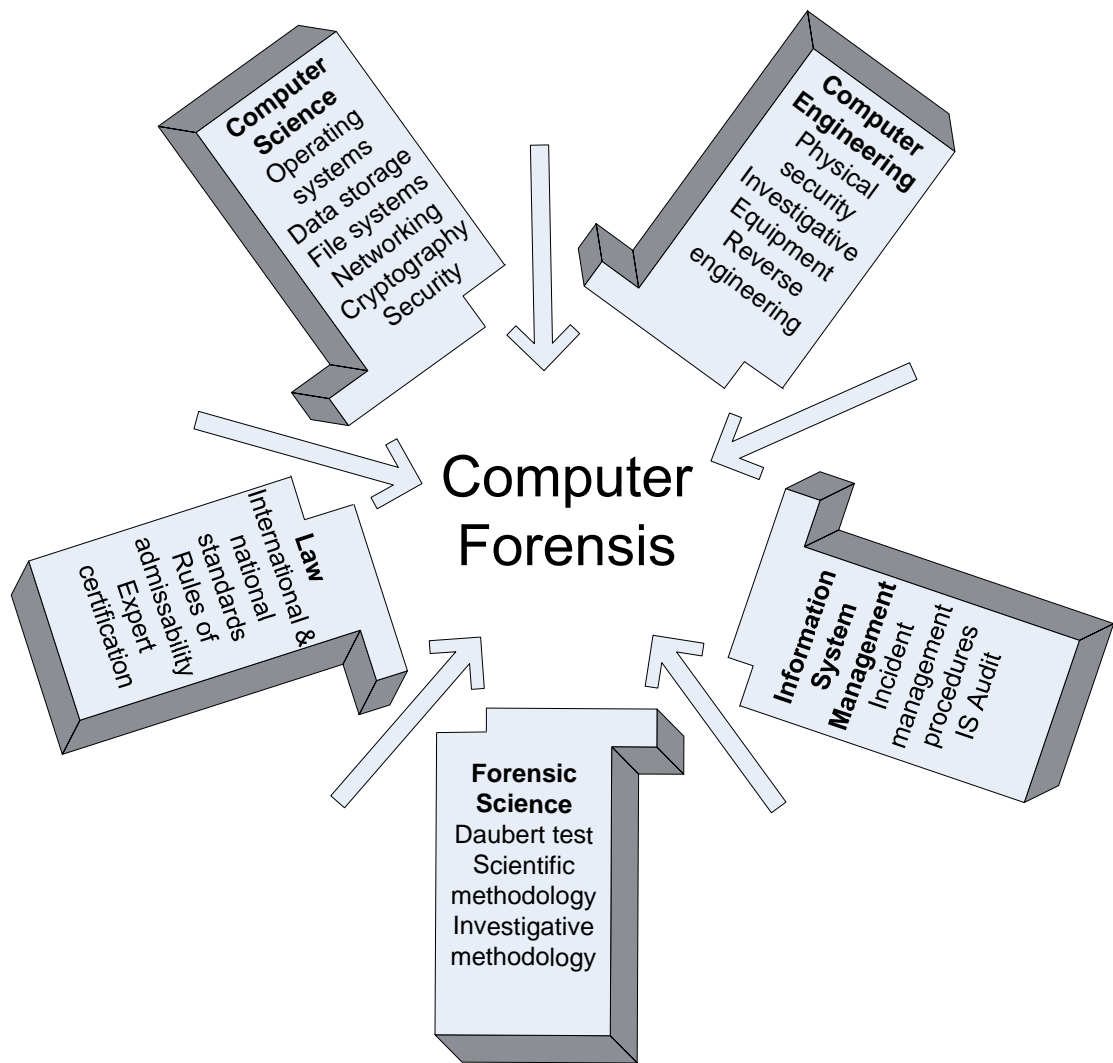


Figure 6-1 Computer Forensics – Synergies With Other Disciplines

6.1 Emergence of a Discipline of Science

Physics is the foundation of all natural sciences, as it is human nature to strive to better understand the universe. However for a long time there was no science named “physics”; this term appeared in the English language only in the 19th century during the second scientific revolution. Physics was not universally recognised as a discipline of science for a long time [69]; it took many years for the scientific community and society at large to embrace and accept it [25]. There are numerous examples of physicists who

were not called by that name by their contemporaries. For example Newton who lived in the 17th century was not called a “physicist” in his time. Indeed, what is today acknowledged as physics often has roots in discoveries which can be traced back thousands of years: for example the Chinese made simple magnets from lodestone and used them for navigation as early as 2700BC [71].

Another example of a well established although much younger discipline is computer science (see also Figure 1-1 in the Introduction chapter). The first electronic computers appeared around the late 1940s as specialised scientific instruments used mainly in military applications, and slowly developed into commercial machinery for business applications. Around the 1970s computers moved into homes and became known as “personal computers” or PCs, and finally around the mid 1980s networking and later the Internet became common [40]. In the early 1940s Thomas J Watson [154], then the president of IBM, was known to say: "there is a world market for about five computers" [123]. Should his prediction have proved to be correct it is unlikely that computing would become the distinct discipline of science it is today.

Mechanical calculating instruments (such as the abacus) existed since antiquity in different forms in all ancient civilisations [158]. It is uncertain when the body of knowledge relating to computing became computer science, but it appears to have happened around the late 1950s when electronic calculating machines became more common. However even in the 1950s it was not clear that computing machinery required its own, new discipline of science. Many considered it to be a part of electrical engineering or mathematics, and in some countries during the early period of computer development people referred to computers as “mathematical machines” [124].

Not every body of knowledge succeeds to become science: one such example is the study of formal organisations which started in the mid-1960s as studies of formal

structures and social mechanisms of large organisations. Substantial research [131] was conducted in the area of interrelation of organisational complexities, rules, levels of authority, etc. The work was carried by strong research groups located in the USA and Great Britain till the late 1970s, but by the 1980s the field was left abandoned and almost no new research had been published in professional journals. The work done in this field was not wasted: it is still being quoted in textbooks [144], and occasionally a new study is being published [144]. However depending on the aspect of an organisation being analysed and focused on, the study can be considered to be a part of psychology [68], sociology [66] or business management [129]. Formal organisations studies failed to establish itself as a separate discipline of science and eventually became marginalised because the field never achieved impact, and it “*lost connection to the larger issues which had generated the original questions*” [160]. This is not to suggest that the body of knowledge became irrelevant or forgotten: the field simply never reached sufficient momentum and support at large to grow and separate itself from other disciplines.

The above examples are separated by 200 years, yet they demonstrate surprisingly similar mechanisms which are at work. Physics and computing were disciplines unique and fundamental in nature, relevant to current issues, and of interest to sufficiently large groups of people to influence and impact society. They both succeeded and emerged as separate disciplines of science, while formal organisations studies failed to do so. However during each of their formative years the ultimate outcome was far from obvious.

Any body of knowledge needs to build a critical mass and wide acceptance. While technological progress, enthusiasm and a level of commitment from the participants are the necessary starting point to create the momentum, they may not be sufficient. If the research work and publications focus only on technical aspects, the transition from a

body of knowledge to a discipline of science may never occur, as demonstrated on the example of study of formal organisations.

Because of its cross disciplinary nature computer forensics needs to base its progress on a multiplicity of concepts from different fields— see Figure 6.1 . An expansion of the science base requires not only contributions with technical contents, but also an influx of research work and publications providing a conceptual platform for establishing a new paradigm for future researchers. The group of people who practice the discipline professionally needs to grow to strengthen its position, and students leaving higher education institutions need to see computer forensics as a viable and attractive career path. People starting their professional career are often uncertain as to which specific area of science or technology to enter. It makes their decisions easier if they were provided with the highest quality of education [102]; often it may even show them new directions they did not know existed. However for the body of knowledge to succeed in becoming a part of the tertiary curriculum it has to be formalised and systemised [80].

6.2 Computer Forensics Education

The study of the current computer forensics education in Australian tertiary institutions (see Appendix L p.1383) led to the systemisation of approaches to teaching in this field. The first computer forensics subject was offered by the Edith Cowan University (ECU) in 2003, and the ECU continues to provide a comprehensive range of computer forensics degrees at all levels [52]. Currently around six other universities offer various subjects in computer forensics, but there is a lack of a unified approach: unlike other fields of computer science, no guidelines or recommendations exist for computer forensics curricula.

The computer forensics specialisation for the Bachelor of Computer Science degree at the University of Western Sydney (UWS) was designed in 2005, first offered in 2006,

and is current as of 2009 [13]. Introduction of the computer forensics specialisation for the Bachelor of Computer Science degree at the School of Computing and Mathematics (SCM) provided further impetus to formally systemise the computer forensics body of knowledge.

In very broad terms computer forensics investigators belong to one of two groups [44]: one group are people trained to use specific tools, the other group are people who possess a good understanding of hardware, software and computing. Obviously any investigator should have some experience with commercial tools and good understanding of computers; these do not have to be mutually exclusive. All academics involved in developing the curriculum agreed that for a computer science student who gained a good understanding of various aspects of computing it is less important to be familiar with a specific commercial package, and more important to understand how a specific tool works and why it should be selected for a specific task. This methodology leans in a natural way towards open source tools, which are usually more difficult to use by someone with a shallow understanding of computing. However these tools offer excellent results and unprecedented power in the hands of a person who understands what they do, and how they do it.

The exemplar of the proposed methodology was the development of the curriculum for the computer forensics specialisation for the Bachelor of Computer Science degree at UWS is presented in Appendix N (see p. 16). The curriculum development was based to a large extent on research and professional experience in the related fields of the academics involved. A few textbooks were recommended as general reading and reference materials, but it was stressed that they will not be followed chapter by chapter during the delivery of the subject.

The subject delivery was based on current research work and recent materials collected from numerous sources. While some commercial tools were occasionally used during lectures and laboratories, the majority of tools used were free and open source (see Appendix M p.31). The student feedback during the subject delivery and from the anonymous end-of-session questionnaires confirmed the full success of this approach (see Appendix M p.32).

All experiences and the lessons learned during the design and delivery of the subject were clearly described for anyone interested to continue improving it (see Appendix M p.30). The effort has been already acknowledged internationally thus significantly contributing to building up the overall momentum of the discipline. Each year each student completing the Computer Forensics major increases the number of professionals who either work in the area of computer forensics, or draw on their knowledge of this field in their work.

Higher education students and professionals working in various computer related areas are often not aware of the discipline of computer forensics, and how it can benefit their work. To evoke their interest it is necessary to offer them access to tools which are readily available and thus encourage experimentation. Commercial tools are not the best choice for this purpose because of the cost involved: they address the need of specialist market, where the price of a tool is of little concern (for example the law enforcing agencies). A single software package can easily cost around US\$5,000 or even more [2]. Our research demonstrated (see Appendix N p.209) that in many cases access to expensive commercial software is not necessary, as the same results can be achieved with free or open source software [138]¹⁶.

¹⁶ For the purpose of this work it is only relevant to make a distinction between two broad categories of software licensing: no cost and commercial. No cost category is referred to as “free or open

To demonstrate the benefits of employing non-commercial software tools in computer forensics an innovative approach was proposed where two parallel environments in a virtual machine configuration allow switching between commercial software tools and free open source tools. This approach provides a valuable verification of results obtained when analysing the same problem in two different environments. Specific examples were provided to encourage experimentation amongst students as well as professionals. Many of them would not be able to duplicate the suggested hardware and software configuration and extend the experiments in the directions suggested if it required expensive software. The experimental configuration suggested offers a unique advantage of comparing what can be achieved in different Windows and Linux environments, and determining which tools are best suited for a given task. Such an approach creates a mechanism encouraging the use of both Windows and Linux environments for the same task, and subsequently expands the skills and expertise of personnel in a natural way. At the same time working in a dual operating system environment does not force an investigator more familiar with Windows to abandon the environment they feel more comfortable, allowing them to see what are the advantages of a Linux based tool. Finally, less common operating system environments can also be used, for example Sun Solaris which offers a somewhat different virtualisation environment with unique computer forensics capabilities [155].

6.3 Computer Forensics in Foundation Studies

For a field of knowledge to achieve the critical mass needed to transform it into a discipline of science it is not sufficient to operate only in the realm of academia, even if

source”, which is also known as F/OSS, FOSS, or FLOSS (free/libre/open source). It should be noted that there are many licensing schemes for open source software, for example GNU General Public License (GPL), GNU Lesser General Public License (LGPL), Affero General Public License (GNU AGPL), The Mozilla Public License, etc.

the researchers and academics working in the field are talented and committed to make their vision a reality. The discipline also needs to make an impact amongst the public at large, for example by introducing its basic principles to a broad range of higher education level students.

An excellent opportunity to reach students with no assumed technical skills or knowledge is PC Workshop, an introductory computing subject offered by the SCM as an open elective. The subject was not used to teach the students how to use computer forensics tools, but to make them aware that such a field exists and to expose them to a few general concepts specific to this field.

PC Workshop aims to guide the students towards becoming power users of computers by giving them the ability to explore, learn, and solve problems they will face in the future when using personal computers. There is a growing trend in educational institutions at all levels to offer introductory computing courses to every student, even those enrolled for non-technical degrees. Academics realise that computers are tools commonly used in practically all disciplines, and thus relevancy and importance of computer literacy for every educated person is unquestionable [121].

There are many educational institutions which changed their curricula to reflect this view, for example Georgia Institute of Technology, USA, in 1999 adopted a requirement that all students must take an introductory computing course [62]. It is reasonable to assume that this is a growing trend, and more introductory computing courses are being designed to reach students at all levels.

PC Workshop was a case study to prove and demonstrate that selected concepts of computer forensics can be successfully taught to students with no previous exposure to advanced computing concepts (see Appendix O p.6). When designing the subject two

basic approaches were considered: traditional, or content-based teaching, and the exploratory learning technique.

Content-based teaching is still the most common teaching technique; it places heavy emphasis on lecturing, and leaves very little flexibility for experimenting beyond prescribed activities. There are two disadvantages of this approach. Firstly there are no introductory personal computing textbooks which include computer forensics concepts. Secondly while the content-based, structured approach has educational advantages in certain fields (for example introductory mathematics subjects) it is a poor choice as a method of introducing new concepts which students can discover themselves by experimenting.

In designing the PC Workshop we adopted the exploratory learning technique, which is still not used commonly in university undergraduate courses. The textbook adopted was a comprehensive and well illustrated book which explains basic concepts of all aspects of computing [16], but not surprisingly it does not mention computer forensics. PC Workshop was designed with heavy emphasis on practical laboratory exercises. During the laboratories the students were encouraged to discover and learn by experimenting beyond the tasks described in the laboratory work sheets, and to explore new areas leading to the realisation that there is always more beyond the obvious. This approach gave us the opportunity to successfully introduce selected concepts and techniques inherent to computer forensics, for example multi partitioning of a hard disk drive or data recovery techniques. Such concepts are usually considered to be advanced and are rarely included in entry level computing courses. The students responded very well and with curiosity when exposed to new, unorthodox topics not covered in standard introductory texts. This approach succeeded, and the students left with a general understanding of the concepts of computer forensics and the ability to apply them in

their professional field. Another step towards strengthening the position of computer forensics as a discipline of science was achieved.

6.4 Summary of Contributions

This chapter makes a significant contribution to the design, development and implementation of computer forensics curricula at the tertiary level. Sharing knowledge and improving educational techniques is the best mechanism to create an influx of new researchers and practitioners, and in turn to sustain the momentum of progress which can transform the body of knowledge and practices into a discipline of science.

To systemise how teaching is conducted an overview of current computer forensics education in Australian tertiary institutions has been performed (see Appendix L). We also discussed the position of computer forensics in the body of knowledge and the issues of curriculum development, including the involvement of professional societies.

Research into higher degree teaching methodologies was applied in practice to create successful computer forensics curricula for university level computer subject, Computer Forensics Workshop (see Appendix M). The subject was developed and delivered repeatedly over the years with full success and a consistently very high level of student satisfaction¹⁷. The project fully succeeded, and the experience gained teaching computer forensics at the tertiary level and the lessons learned in the process were shared with the international academic community through publishing our findings and through extensive personal contacts.

A significant contribution was made by researching the application of open source software tools in computer forensics education at the tertiary level. The argument was

¹⁷ As measured by the UWS anonymous student surveys.

put forward that open source tools are more suitable in education than commercial tools, as they provide the opportunity for students to gain an in-depth understanding and appreciation of the computer forensic process as opposed to familiarity with one software product. A comprehensive case study was presented to illustrate this point (see Appendix N).

A contribution was made to advance the recognition of computer forensics as a field of science by introducing its basic concepts to a broader audience. By analysing the design and delivery of “PC Workshop” as another case study, we demonstrated that selected concepts of computer forensics can be taught to students at the introductory level (see Appendix O).

7 Summary of Contributions and Future Work

This thesis makes a material contribution to strengthening and advancing the position of computer forensics as a discipline of science by:

- presenting a comprehensive overview of its history,
- identifying new areas crucial to further development of the discipline,
- developing a series of new methodologies and providing exemplars illustrating new approaches,
- designing, developing and implementing computer forensics curricula at the tertiary level.

Chapter 2 of the thesis (see Appendices A, B, C, D and E) contributes significantly to the discipline of computer forensics by presenting a comprehensive analysis of the emergence of computer forensics, setting current research directions and identifying future challenges. Today it is easy to maintain a Web site located beyond local jurisdiction [59], to use free and powerful data encryption tools [145] or to use practically unlimited on-line storage [5] located in a remote place. None of those technologies existed when the discipline of computer forensics was formed, and the challenge for the discipline is to understand the latest developments in the computer field, and to anticipate how they can be used by lawbreakers (see Appendix A).

The analysis concludes that computer technology reached a stage when the conventional, mechanistic approach to collecting and analysing data is no longer sufficient. Two areas crucial to further development of the discipline were identified:

technical (memory forensics – see Appendices B and C) and organisational (computer forensics vs. information security (IS) audit— see Appendices D and E).

The thesis presents methodology quantifying persistence of user data in physical computer memory (see Appendix C p.70), and proposes a new technique which allows for forensically sound acquisition of memory based on the persistence paradigm (see Appendix B p.134). Research into similarities and differences between computer forensics and IS audit (see Appendix E p.113) resulted in developing a hybrid methodology which ensures that the evidence found in the course of an IS audit is forensically sound (see Appendix D p.7).

Chapter 3 of the thesis (see Appendices F, G and H) proves that there is very little finalised, and a lot still to be discovered in the conventional computer forensics methodology. As an example, the new NTFS file format system introduced many peculiarities and issues which had to be urgently addressed by computer forensic investigators. The thesis makes an important contribution to a better understanding of the internal NTFS mechanisms.

A formalised methodology for hiding data in the NTFS file system and the techniques to detect and recover hidden data has been described (see Appendix F p.118). An exemplar of this methodology was proposed where the system backup software was tested and, depending on its capability to save and restore NTFS Alternate Data Streams, was classified as belonging to one of five classes (see Appendix G p.450). A new methodology for extraction of EFS (Encrypted File System) files (see Appendix H p.3) was presented, and an exemplar of formalised procedures for extracting EFS encrypted files from live systems in a forensically sound way was described (see Appendix H p.5).

Chapter 4 of the thesis (see Appendices I, J and K) describes innovative research connecting virtualisation (which is aimed mainly at server applications) and forensics. A methodology reducing the time of the analysis phase of computer forensics investigation was developed (see Appendix I p.3) where conventional and virtual computing environments are used in parallel. Formalised exemplars of analysing a hard disk images (see Appendix I p.5) and USB flash drives in virtual environments (see Appendix J p.3) were presented.

As a result of extensive research experience with free and open source tools an approach was proposed leading to a gradual shift from closed source software tools to open source software (OSS) tools in computer forensics investigations (see Appendix K p.3) An exemplar case was presented which demonstrates a strong advantage of using a dual Windows / Linux virtual environment to access disk images with different operating systems (see Appendix K p.6).

Chapter 5 of the thesis (see Appendices L, M, N and O) makes a pioneering contribution to the methodology of computer forensics tertiary education by creating curricula and describing in detail the design and development of the Computer Forensics Workshop subject. In order to systemise how teaching is conducted a comprehensive overview of current computer forensics education in Australian tertiary institutions had been performed (see Appendix L p.1386), contributing to a better understanding of the position of computer forensics in the body of knowledge.

Computer Forensics Workshop (see Appendix M p.29) is an exemplar of suitable teaching methodologies for creating a comprehensive computer forensics curriculum for a university level computer subject. The subject was delivered repeatedly over the years with full success and a consistently very high level of student satisfaction (see Appendix M p.32). The actual application of open source software tools in computer

forensics education at the tertiary level led to the conclusion that such tools provide a better opportunity for students to gain an in-depth understanding and appreciation of the computer forensic process as opposed to familiarity with one software product (see Appendix N p.15).

Finally, a significant contribution was made to advance the recognition of computer forensics by introducing its basic concepts to a broader audience. Selected concepts of computer forensics were included in the curricula designed to teach students in PC Workshop, an introductory level subject (see Appendix O p.10).

Further research is needed in all areas of computer forensics. The recent realisation that the data which a computer investigator intends to capture is often dynamic and volatile (see Chapter 3) has led to increased interest in live system investigations, and subsequently created a need to develop better tools for live system memory capture and analysis. While various software products already exist they are mostly command line based, experimental or proof of concept, and are not ready for forensic investigation work.

More work is needed in the area of IS audit and computer forensics to better understand the common requirements of both fields and to develop common approaches and more mature tools. A hybrid methodology combining computer forensics and computer audit tools needs further refinement and more exemplars.

New, recently introduced file systems are complex and full documentation of their internals is not readily available. Furthermore their metadata structures provide an almost unlimited scope for hiding information (see Chapter 4). The body of research presented in the thesis is a foundation for creating an easy to use software tool which could comprehensively scan NTFS structures for hidden data. However the

methodologies rely on command line tools, for example Robocopy [10]. An unavoidable danger when using such tools is that selecting an incorrect parameter may result in damaging or invalidating all the evidence being collected. To minimise the chance of errors and make the tools suitable for use by investigating technicians all procedures should be at least partly automated by preparing a series of scripts which reduce the chance of entering wrong parameters. These procedure should preferably be based on improved graphics interfaces [86]. Collecting selected utilities and creating one toolkit with a simple to use graphics interface would help less experienced investigators to acquire the evidence from a live system in a reliable and efficient manner.

Virtualisation is a rapidly maturing technology which was not created for the computer forensics field, but it has tremendous potential in helping to analyse the computer related evidence and assisting in presenting the findings in a court environment (see Chapter 5). There is a large community of computer forensics investigators, thus virtual machine configurations should be further developed and specifically tailored for unique computer forensics needs. More work is also required to create exemplars of complete environments to demonstrate the accuracy and time saving advantages of dual Windows / Linux platforms when used as tools of forensics.

Further research is required into specific requirements and techniques of booting a forensic image in a virtual environment. This appears to be a very attractive application of virtualisation, but at this stage a lot of work would be required to make the process reliable and universally applicable to a wide range of computer configurations an investigator may encounter. Work done by Carnegie Mellon University resulted in the release of the Live View software tool [6] which creates a VMware virtual machine out of raw disk images. However the tool still fails to work in some cases and its application in professional investigations is somewhat limited at best.

Work also needs to continue in the field of education (see Chapter 6). Each year brings new technological developments and teaching a university level course in a new growing discipline has to be based on continuously updated curricula. The curriculum developed for this purpose would soon become obsolete unless scholarship and research is continued [94]. At the same time there is also an almost unlimited scope to use computer forensics techniques illustrated with various practical examples when teaching introductory level computing subjects to undergraduate students.

Computer forensics is impacting not only academics and researchers, but also the general public. All organisations and individuals intuitively accept that the security of their computers or computer networks is not infallible, and when it fails computer forensics is indispensable to determine how the system was compromised [97]. A large and growing community of professional investigators are continuously updating their knowledge and have a keen interest in specific solutions based on scenarios likely to occur in their daily practice. Organisations are also keen to know how to increase the protection of their computer networks in the future. For this reason the significance and pervasiveness of computer forensics continues to grow, and its universal importance can not be overestimated.

References

- [1] *Britannica online*, Britannica Online, Chicago, IL, 1994.
- [2] *EnCase Forensic Modules*, 2009,
http://www.guidancesoftware.com/products/ef_modules.asp.
- [3] *Handbook of computer crime investigation forensic tools and technology*, in E. Casey, ed., Academic Press, San Diego, Calif., 2002.
- [4] *IDC Corporate Headquarters*, 2009, <http://www.idc.com/home.jhtml>.
- [5] *Internet Virtual Storage*, 2007,
<http://www.cryer.co.uk/resources/virtualstorage.htm>.
- [6] *Live View*, CERT, Software Engineering Institute, 2007.
- [7] *NTFS-3G Stable Read/Write Driver*, <http://www.ntfs-3g.org/>.
- [8] *openSUSE*, <http://www.opensuse.org/>.
- [9] *Princeton University Center for Information Technology Policy*,
<http://citp.princeton.edu/memory/>, 2009.
- [10] *Robocopy.exe Robust File Copy Utility Version XP010*, 2003,
<http://www.microsoft.com/downloads/details.aspx?FamilyID=9d467a69-57ff-4ae7-96ee-b18c4790cffd&DisplayLang=en>.
- [11] *The Secret History of Hacking*, SBS, Sydney, N.S.W., 2003.
- [12] *Techno security's guide to e-discovery and digital forensics*, in J. Wiles, ed., Syngress Publishing, Burlington, MA, 2007.
- [13] *UWS Handbook - units*, 2009,
<http://handbook.uws.edu.au/hbook/unit.aspx?unit=300447.1>.
- [14] A. Abbate, J. Walker, S. Chimner and R. Morimoto, *Microsoft Windows Vista management and administration*, Sams Pub., Indianapolis, Ind., 2007.
- [15] A. Al Mamun, G. Guo and C. Bi, *Hard Disk Drive Mechatronics and Control*, CRC Press, Boca Raton, Fla., 2007.

- [16] J. Andrews, *A+ Guide to Managing and Maintaining Your PC*, Thomson Course Technology, Boston, Massachusetts, U.S.A., 2007.
- [17] J. Ashcroft, *Electronic Crime Scene Investigation: A Guide for First Responders*, U.S. Department of Justice Office of Justice Programs, Technical Working Group for Electronic Crime Scene Investigation, July 2001.
- [18] M. Atkinson and R. Welland, *Fully Integrated Data Environments: Persistent Programming Languages, Object Stores, and Programming Environments*, Springer, Berlin, 2000.
- [19] R. N. Barger, *Computer Ethics: a Case-Based Approach*, Cambridge University Press, Cambridge, 2008.
- [20] A. Bartoli and G. Dini, *Mechanisms For Application-Level Recoverable-Persistence in a Single Address Space*, *Microprocessors and Microsystems*, 22 (1998), pp. 247-261.
- [21] A. Basta and W. Halton, *Computer Security and Penetration Testing*, Thomson, United States, 2008.
- [22] P. Bijaoui, *Designing Storage for Exchange 2007 SP1*, in J. Hasslauer and ScienceDirect, eds., *Digital Press/Syngress/Elsevier*, Amsterdam, 2008.
- [23] P. M. Blau and W. R. Scott, *Formal organizations: a comparative approach*, Stanford Business Books, Stanford, Calif., 2003.
- [24] A. Blyth, *Information Assurance Security in the Information Environment*, in G. L. Kovacich, ed., *Springer-Verlag London Limited*, London, 2006.
- [25] P. J. Bowler and I. R. Morus, *Making modern science a historical survey*, University of Chicago Press, Chicago, 2005.
- [26] M. Brain, *Win32 system services: the heart of Windows 95 and Windows NT*, Prentice Hall PTR, Upper Saddle River, N.J., 1996.
- [27] M. Britz, *Computer Forensics and Cyber Crime: An Introduction*, Pearson/Prentice Hall, Upper Saddle River, NJ, 2004.
- [28] D. A. Bronstein, *Law for the Expert Witness*, CRC Press, Boca Raton, FL, 2007.
- [29] C. L. T. Brown, *Computer Evidence: Collection & Preservation*, Charles River Media, Hingham, MA, 2005.
- [30] S. Bunting, *EnCase Computer Forensics: the Official EnCE EnCase Certified Examiner Study Guide*, Wiley Pub., Indianapolis, Ind., 2008.

- [31] M. Burdach, *Digital forensics of the physical memory*, 2007, <http://forensic.seccure.net/>.
- [32] C. Burgess, *Secrets Stolen, Fortunes Lost. Preventing Intellectual Property Theft and Economic Espionage in the 21st Century*, in R. Power, ed., Syngress, Rockland, Mass., 2008.
- [33] B. Burns, J. Granick and et al., *Security Power Tools*, O'Reilly Media, Inc., Sebastopol, CA, 2007.
- [34] T. Bynum, *Computer Ethics: Basic Concepts and Historical Overview*, Winter 2001, <http://plato.stanford.edu/archives/win2001/entries/ethics-computer/>.
- [35] B. Carrier, *File System Forensic Analysis*, Addison-Wesley, Upper Saddle River, NJ, 2005.
- [36] B. D. Carrier and J. Grand, *A Hardware-Based Memory Acquisition Procedure for Digital Investigations*, Digital Investigation, 1 (2004), pp. 50-60.
- [37] H. Carvey, *Windows Forensic Analysis*, Syngress, Rockland, Massachusetts, U.S.A., 2007.
- [38] H. Carvey, *Windows Forensics and Incident Recovery*, Addison-Wesley, Boston, MA, 2004.
- [39] E. Casey, *Digital Evidence and Computer Crime*, Elsevier Academic Press, London, UK, 2004.
- [40] P. E. Ceruzzi, *A history of modern computing*, MIT Press, Cambridge, Mass., 1998.
- [41] P. E. Ceruzzi, *A History of Modern Computing*, in I. NetLibrary, ed., MIT Press, Cambridge, Mass., 1998.
- [42] S. Chakrabarti, *Data Mining: Know it all*, Elsevier/Morgan Kaufmann Publishers, Burlington, MA, 2009.
- [43] W. R. Cheswick and S. M. Bellovin, *Firewalls and Internet Security: Repelling the Wily Hacker*, Addison-Wesley, Reading, Mass., 1994.
- [44] F. Clark, *Investigating computer crime*, in K. Diliberto, ed., CRC Press, Boca Raton, Florida, 1996.
- [45] T. Clark, *Storage Virtualisation technologies for Simplifying Data Storage and Management*, Pearson Education, Upper Saddle River, NJ, 2005.

- [46] P. Crowley and D. Kleiman, *CD and DVD Forensics*, Syngress, Rockland, MA, 2007.
- [47] H. Custer, *Inside the Windows NT File System*, Microsoft Press, 1994.
- [48] M. A. Cusumano and R. W. Selby, *Microsoft secrets how the world's most powerful software company creates technology, shapes markets, and manages people*, Free Press, New York, 1995.
- [49] W. M. Dale and W. S. Becker, *The crime scene: how forensic science works*, Kaplan Publishing, New York, NY, 2007.
- [50] P. De Guise, *Enterprise systems backup and recovery a corporate insurance policy*, CRC Press, Boca Raton, 2009.
- [51] M. P. Dierks, *Computer Network Abuse*, Harvard Journal of Law & Technology, Volume 6 Number 2 (1993).
- [52] ECU, *Edith Cowan University Handbook*, 2009, <http://handbook.ecu.edu.au/>.
- [53] D. Farmer and W. Venema, *Forensic Discovery*, Addison-Wesley, Upper Saddle River, NJ, 2005.
- [54] C. Fehily, *Microsoft Windows Vista*, Peachpit Press, Berkeley, Calif., 2008.
- [55] M. M. Ferraro, M. McGrath and E. Casey, *Investigating Child Exploitation and Pornography: the Internet, the Law and Forensic Science*, Elsevier/Academic Press, Amsterdam, 2005.
- [56] S. Fleischmann, *WinHex/X-Ways Forensics*, 2007, <http://www.x-ways.net/forensics/>.
- [57] R. E. Gaensslen, H. A. Harris and H. C. Lee, *Introduction to Forensic Science and Criminalistics*, McGraw-Hill Higher Education, New York, 2008.
- [58] A. Gillen and B. Waldman, *Linux Adoption in a Global Recession*, IDC, Framingham, USA, 2009, pp. 12.
- [59] J. Gordon, *Illegal Internet Networks in the Developing World*, 2004, http://cyber.law.harvard.edu/home/research_publication_series.
- [60] P. N. Grabosky, *Electronic Crime*, Pearson Prentice Hall, Upper Saddle River, N.J., 2007.
- [61] B. J. Grundy, *The Law Enforcement and Forensic Examiner's Introduction to Linux, A Beginner's Guide*, 2007.

- [62] M. Guzdial, *Teaching computing to everyone*, Communications of the ACM, 52 (2009), pp. 31-33.
- [63] L. Hadfield, D. Hatter and D. Bixler, *Windows NT server 4 security handbook*, Que Corp., Indianapolis, IN, 1997.
- [64] D. W. Hagy, *Digital Evidence in the Courtroom: A Guide for Law Enforcement and Prosecutors*, U.S. Department of Justice Office of Justice Programs, Technical Working Group for Electronic Crime Scene Investigation, 2007.
- [65] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum and E. W. Felten, *Lest We Remember: Cold Boot Attacks on Encryption Keys*, USENIX Security Symposium, USENIX Association, 2008, pp. 45-60.
- [66] M. J. Handel, *The sociology of organizations. classic, contemporary, and critical readings*, Sage Publications, Thousand Oaks, 2003.
- [67] S. V. Hart, *Forensic Examination of Digital Evidence: A Guide for Law Enforcement*, U.S. Department of Justice Office of Justice Programs, Technical Working Group for Electronic Crime Scene Investigation, April 2004.
- [68] S. A. Haslam, *Psychology in organizations the social identity approach*, SAGE, London, 2001.
- [69] J. L. Heilbron, *The Oxford guide to the history of physics and astronomy*, Oxford University Press, Oxford, 2005.
- [70] G. Henriksen, *Windows NT and UNIX integration*, Macmillan Technical Publishing, Indianapolis, IN, 1998.
- [71] J. B. Hewson, *A history of the practice of navigation*, Glasgow, 1951.
- [72] T. Howlett, *Open Source Security Tools Practical Applications for Security*, Prentice Hall Professional Technical Reference, Upper Saddle River, N.J., 2004.
- [73] D. J. Icové, K. A. Seger and W. VonStorch, *Computer Crime: a Crimefighter's Handbook*, O'Reilly & Associates, Sebastopol, California, 1995.
- [74] G. Ifrah, *The Universal History of Computing From the Abacus to the Quantum Computer*, John Wiley, New York, 2000.
- [75] N. S. W. International Workshop on Persistent Object Systems: Newcastle, J. Rosenberg, D. Koch and S. British Computer, *Persistent Object Systems. Proceedings of the Third International Workshop 10-13 January, 1989, Newcastle, Australia*, Springer-Verlag, London, 1990.

- [76] B. Jacob, S. Ng and D. Wang, *Memory Systems Cache, DRAM, Disk*, Morgan Kaufmann Publishers, Burlington, MA, 2008.
- [77] JISC, *The Data Deluge: Preparing for the Explosion in Data*, 2004, <http://www.jisc.ac.uk/>.
- [78] K. Jones, R. Bejtlich and R. Curtis, *Real Digital Forensics: Computer Security and Incident Response*, Penguin Books Pearson Publishing, 2005.
- [79] P. Kanellis, *Digital crime and forensic science in cyberspace*, Idea Group Pub., Hershey PA, 2006.
- [80] A. V. Kelly, *The curriculum: theory and practice*, Sage Publications, London, 2004.
- [81] D. E. Knuth, *The Art of Computer Programming*, Addison-Wesley, Upper Saddle River, NJ, 2005.
- [82] A. G. Konheim, *Computer security and cryptography*, John Wiley & Sons, New Jersey, 2007.
- [83] G. L. Kovacich and A. Jones, *High-Technology Crime Investigator's Handbook*, Butterworth-Heinemann, Burlington, 2006.
- [84] J. Kratochvil, *Captive: The first free NTFS read/write filesystem for GNU/Linux*, <http://www.jankratochvil.net/project/captive/>.
- [85] W. G. Kruse II and J. G. Heiser, *Computer Forensics: Incident Response Essentials*, Addison Wesley Professional, 2002.
- [86] C. LaVelle and A. Konrad, *FriendlyRoboCopy: A GUI to RoboCopy for computer forensic investigators*, Digital Investigation, 4 (2007), pp. 16-23.
- [87] S. Levy, *Hackers: Heroes of the Computer Revolution*, Penguin, London, 1994.
- [88] A. Lewis, *Child Sexual Abuse*, Greenhaven Press, Farmington Hills, MI, 2005.
- [89] R. Love, *Linux kernel development*, Novell Press, Indianapolis, Ind., 2005.
- [90] S. Mahan and P. L. Griset, *Terrorism in Perspective*, Sage, Thousand Oaks, Calif., 2003.
- [91] K. Mandia, C. Prosie and M. Pepe, *Incident Response & Computer Forensics, Second Edition*, McGraw-Hill/Osborne, Emeryville, CA, 2003.
- [92] W. Mao, *Modern cryptography theory and practice*, Prentice Hall PTR, Upper Saddle River, NJ, 2004.

- [93] J. Markoff, *What the Dormouse said - How the Sixties Counterculture Shaped the Personal Computer Industry*, Viking, New York, 2005.
- [94] C. J. Marsh and G. Willis, *Curriculum: alternative approaches, ongoing issues*, Merrill/Prentice Hall, Upper Saddle River, N.J., 2003.
- [95] R. McKemmish, *What is Forensic Computing?*, *Trends & Issues in Crime And Criminal Justice*, Australian Institute of Criminology, 1999, pp. 6.
- [96] G. McKnight, *Computer Crime*, Michale Joseph, London, 1973.
- [97] J. McNamara, *Secrets of Computer Espionage, Tactics and Countermeasures*, Wiley, Indianapolis, Ind., 2003.
- [98] M. Meyers and M. Rogers, *Computer Forensics: The Need for Standardization and Certification*, *International Journal of Digital Evidence*, 3 (2004).
- [99] Microsoft, *Product Activation for Windows Vista® and Windows Server® 2008*, 2008, pp. 8.
- [100] Microsoft, *Windows Genuine Advantage, Reported OEM BIOS Hacks*, 2007, <http://blogs.msdn.com/wga/archive/2007/04/10/reported-oem-bios-hacks.aspx>.
- [101] Microsoft, *Windows Vista Security Guide*, 2006, <http://www.microsoft.com/technet/windowsvista/security/guide.msp#EJBA C>.
- [102] G. Miller, *The career coach: winning strategies for getting ahead in today's job market*, Currency/Doubleday, New York, 2001.
- [103] K. D. Mitnick and W. L. Simon, *The art of Intrusion: the Real Stories Behind the Exploits of Hackers, Intruders, and Deceivers*, Wiley, Indianapolis, IN, 2005.
- [104] G. M. Mohay, *Computer and intrusion forensics*, Artech House, Boston, 2003.
- [105] J. Moskowitz and T. Boutell, *Windows and Linux Integration: Hands-on Solutions for a Mixed Environment*, Wiley, Hoboken, NJ, 2005.
- [106] S. Mueller, *Upgrading and Repairing PCs*, Que Pub., Indianapolis, Ind., 2008.
- [107] A. Muller, *Scripting VMware Power Tools for Automating Virtual Infrastructure Administration*, Syngress, Rockland, MA, 2006.
- [108] Y. F. Musaji, *Auditing and security: AS/400, NT, UNIX, networks, and disaster recovery plans*, John Wiley, New York, 2001.

- [109] B. L. Nacos, *Terrorism and Counterterrorism: Understanding Threats and Responses in the Post-9/11 World*, Pearson Longman, New York, 2008.
- [110] D. C. Naik, *Inside Windows storage: server storage technologies for Windows 2000, Windows Server 2003, and beyond*, Addison-Wesley, Boston, 2004.
- [111] B. Nelson, A. Phillips, F. Enfinger and C. Steuart, *Guide to Computer Forensics and Investigations, Second Edition*, Thomson Course Technology, Boston, MA, 2006.
- [112] R. C. Newman, *Computer forensics evidence collection and management*, Auerbach Publications, Boca Raton, Fla., 2007.
- [113] M. Newton, *The Encyclopedia of Crime Scene Investigation*, Facts On File, New York, 2008.
- [114] M. Newton, *The Encyclopedia of High-tech Crime and Crime-fighting*, Facts On File/Checkmark Books, New York, 2004.
- [115] R. E. Overill, *Computer Crime - an Historical Survey*, 1998,
<http://www.kcl.ac.uk/orgs/icsa/Old/crime.html>.
- [116] D. B. Parker, *Crime by Computer*, Scribner, New York, 1976.
- [117] D. B. Parker, *Fighting Computer Crime, a New Framework For Protecting Information*, J. Wiley, New York, 1998.
- [118] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design The Hardware/Software Interface*, Elsevier Morgan Kaufmann, Boston, 2009.
- [119] S. Peisert, M. Bishop and K. Marzullo, *Computer Forensics in Forensics*, SIGOPS Oper. Syst. Rev., 42 (2008), pp. 112-122.
- [120] M. A. Penhallurick, *Methodologies for the use of VMware to Boot Cloned/Mounted Subject Hard Disk Images*, 2005,
<http://www.e5hforensics.com/Downloads/VMware%20Forensic%20Cloning%20Methodology.pdf>.
- [121] E. F. Provenzo, A. Brett and G. N. McCloskey, *Computers, curriculum, and cultural change: an introduction for teachers*, L. Erlbaum, Mahwah, N.J., 2005.
- [122] N. Provos and P. Honeyman, *Detecting Steganographic Content on the Internet*, CITI Technical Report 01-11, 2001, pp. 13.
- [123] E. W. Pugh, *Building IBM: shaping an industry and its technology*, MIT Press, Cambridge, Mass., 1995.

- [124] P. Rojas, *Encyclopedia of computers and computer history*, Fitzroy Dearborn, London, 2001.
- [125] T. Rude, *dd and Computer Forensics*, 2000, <http://www.crazytrain.com/dd.html>.
- [126] M. E. Russinovich and D. A. Solomon, *Microsoft Windows internals: Microsoft Windows Server 2003, Windows XP, and Windows 2000*, Microsoft Press, Redmond, WA, 2005.
- [127] R. Russon and Y. Fledel, *NTFS Documentation*, <http://www.linux-ntfs.org/content/view/103/42/>.
- [128] R. Saferstein, *Forensic Science Handbook*, Prentice Hall, 2001.
- [129] D. Samson and R. L. Daft, *Fundamentals of management*, Cengage Learning Australia, South Melbourne, Vic., 2009.
- [130] B. Schneier, *Applied cryptography: protocols, algorithms, and source code in C*, Wiley, New York, 1996.
- [131] J. M. Shafritz and J. S. Ott, *Classics of organization theory*, Dorsey Press, Chicago, Ill, 1987.
- [132] A. Silberschatz, P. B. Galvin and G. Gagne, *Operating System Concepts*, John Wiley & Sons, Hoboken, NJ, 2009.
- [133] D. P. Simpson, *Cassell's Latin Dictionary*, Wiley Publishing, Inc., New York, 1968.
- [134] A. Singh, *Mac OS X internals: a systems approach*, Addison-Wesley, Upper Saddle River, NJ, 2007.
- [135] J. E. Smith and R. Nair, *Virtual Machines, Versatile Platforms for Systems and Processes*, Morgan Kaufmann Publishers, San Francisco, CA, 2005.
- [136] D. J. Solove, *The Digital Person, Technology and Privacy in the Information Age*, New York University Press, New York, 2004.
- [137] J. G. Speight, *The Scientist or Engineer as an Expert Witness*, CRC Press, Boca Raton, 2009.
- [138] A. M. St. Laurent, *Understanding open source & free software licensing*, O'Reilly Media, Inc., Sebastopol, CA, 2004.
- [139] G. Stamatellos, *Computer Ethics, a Global Perspective*, Jones and Bartlett Publishers, Sudbury, Mass., 2007.

- [140] P. Stephenson, *Investigating Computer-Related Crime*, CRC Press, Boca Raton, Florida, 2000.
- [141] D. Tapscott, *Grown up Digital: How the net Generation is Changing Your World*, McGraw-Hill, New York, 2009.
- [142] A. Tharp, *File Organization and Processing*, Wiley, Hoboken, NJ 1988.
- [143] J. W. Toigo, *Disaster recovery planning Preparing for the unthinkable*, Prentice Hall PTR, Upper Saddle River, N.J., 2003.
- [144] P. S. Tolbert and R. H. Hall, *Organizations: structures, processes, and outcomes*, Pearson/Prentice Hall, Upper Saddle River, N.J., 2009.
- [145] TrueCrypt, *True Crypt - Free Open-Source On-The-Fly Disk Encryption Software*, 2007, <http://www.truecrypt.org/>.
- [146] J. R. Vacca, *Computer Forensics Computer Crime Scene Investigation*, Charles River Media, Hingham, Mass., 2005.
- [147] D. Verton, *The Hacker Diaries: Confessions of Teenage Hackers*, McGraw-Hill/Osborne, New York, 2002.
- [148] R. Vieler, *Professional Rootkits*, Wiley Pub., Indianapolis, IN, 2007.
- [149] L. Volonino, R. Anzaldua and J. Godwin, *Computer Forensics Principles and Practices*, Pearson Education, Upper Saddle River, N.J., 2007.
- [150] L. Volonino, H. J. Watson and S. Robinson, *Using EIS to Respond to Dynamic Business Conditions*, Decision Support Systems, 14 (1995), pp. 105-116.
- [151] J. Wack, M. Tracy and M. Souppaya, *Guideline on Network Security Testing*, NIST Special Publication 800-42 (2003).
- [152] D. Wall, *Cyberspace Crime*, Ashgate, Aldershot, 2003.
- [153] M. Wark, *A Hacker Manifesto*, Harvard University Press, Cambridge, MA, 2004.
- [154] T. J. Watson and P. Petre, *Father, Son & Co. my life at IBM and beyond*, Bantam Books, New York, 1990.
- [155] P. A. Watters, *Solaris 10 the complete reference*, McGraw-Hill/Osborne, Emeryville, Calif., 2005.
- [156] J. T. Wells, *Computer Fraud Casebook the Bytes That Bite*, Wiley, Hoboken, N.J., 2009.

- [157] M. E. Whitman and G. Holden, *Guide to Firewalls and Network Security: Intrusion Detection and VPNs*, Course Technology Cengage Learning, Boston, Mass., 2009.
- [158] M. R. Williams, *A history of computing technology*, IEEE Computer Society Press, Los Alamitos, Calif., 1997.
- [159] A. Yasinsac, R. F. Erbacher, D. G. Marks, M. M. Pollitt and P. M. Sommer, *Computer Forensics Education*, IEEE Security and Privacy, 1 (2003), pp. pp. 15-23.
- [160] M. Zald, *Sociology as a discipline: Quasi-science and quasi-humanities*, The American Sociologist, 22 (1991), pp. 165-187.

List of Appendices

Important note on page numbering in appendices: all appendices are papers which were previously published, and they are paginated differently to this thesis. Thus each appendix page has two numbers: one as in the original publication, and the second one corresponding to the consecutive page of this thesis. The references to page numbers in this thesis correspond to the respective original paper pagination.

Appendix A

D. Bem, F.Feld, E. Huebner, O. Bem “Computer Forensics - Past, Present And Future”

Published in *Journal of Information Science & Technology*, JIST 5(3), pp 43-59, 2008

The entire paper was written by the present author, under general supervision of E. Huebner and with legal issues input by F. Feld and O. Bem.

Appendix B

E. Huebner, **D. Bem**, F. Henskens and M. Wallis “Persistent Systems Techniques in Forensic Acquisition of Memory”

Published in *Digital Investigation Journal*, Volume 4, Issue 3-4, pp 129-137, September-December 2007

The present author contributed and developed fundamental ideas exposed in this paper, and participated significantly in the writing.

Appendix C

J. M. Solomon, E. Huebner, **D. Bem** and M. Szezynska “User Data Persistence in Physical Memory”

Published in *Digital Investigation Journal*, Volume 4, Issue 2, pp 68-72, June 2007

The present author contributed and developed fundamental ideas exposed in this paper, and participated significantly in the writing.

Appendix D

E. Huebner, M. Szezynska, **D. Bem** “Computer Forensics in IS Audit – a Case Study”

Published in *Recent Advances in Computing and Management Information Systems*, ed. P. Petratos & G.A. Marcoulides, Athens, Greece, 2009

The present author contributed and developed fundamental ideas exposed in this paper, and participated significantly in the writing.

Appendix E

M. Szezynska, E. Huebner, **D. Bem**, C. Ruan “Methodology and Tools of IS Audit and Computer Forensics - the Common Denominator”

Published in *Advances in Information Security and Assurance*, ed. JH Park et al., pp 110-121, Lecture Notes in Computer Science 5576, Springer-Verlag Berlin Heidelberg 2009

The present author contributed and developed fundamental ideas exposed in this paper, and participated significantly in the writing.

Appendix F

E. Huebner, **D. Bem**, C. K. Wee “Data Hiding in the NTFS File System”

Published in *Digital Investigation Journal*, Volume 3, Issue 4, pp 211-226, December 2006

The present author contributed and developed fundamental ideas exposed in this paper, and participated significantly in the writing.

Appendix G

E. Huebner, **D. Bem** “Forensic Extraction of EFS Encrypted Files in Live System Investigation”

Published in *Journal of Digital Forensic Practice*, Volume 2, Issue 1, pp 1-12, 2008

The present author was an equal partner in the research presented in this paper and in the writing.

Appendix H

D. Bem, E. Huebner “Alternate Data Streams in Forensics Investigations of File Systems Backups”

Published in *Current Computing Developments in E-Commerce, Security, HCI, DB, Collaborative and Cooperative Systems*, ed. P. Petratos, pp 449-460, Athens, Greece 2006

The entire paper was written by the present author, under general supervision of the co-author.

Appendix I

D. Bem, E. Huebner “Computer Forensic Analysis in a Virtual Environment”

Published in *International Journal of Digital Evidence*, Volume 6, Issue 2, pp 1-13, Fall 2007

The entire paper was written by the present author, under general supervision of the co-author.

Appendix J

D. Bem, E. Huebner “Analysis of USB Flash Drives in a Virtual Environment”

Published in *The Small Scale Digital Device Forensic Journal*, Volume 1, Number 1, pp 1-6, June 2007

The entire paper was written by the present author, under general supervision of the co-author.

Appendix K

D. Bem “Open Source Virtual Environments in Computer Forensics”

Published in proceedings of *4th International Conference on Open Source Systems (OSS 2008)*, pp 1-13, Milan, Italy, 7-10 September 2008

The entire paper was written by the present author.

Appendix L

E. Huebner, **D. Bem**, C. Ruan “Computer Forensics Tertiary Education in Australia”

Published in proceedings of *International Conference on Computer Science and Software Engineering (CSSSE 2008)*, pp 1383-1387, Wuhan, China, December 12-14, 2008

The present author was an equal partner in the research presented in this paper and in the writing.

Appendix M

D. Bem and E. Huebner “Computer Forensics Workshop for Undergraduate Students”

Published in proceedings of *Tenth Australasian Computing Education Conference (ACE2008)*, pp 29-34, Wollongong, Australia, January 2008

The entire paper was written by the present author, under general supervision of the co-author.

Appendix N

E. Huebner, **D. Bem**, H. Cheung “FLOSS Tools for Computer Forensics Tertiary Education”

Published in proceedings of *4th International Conference on Open Source Systems (OSS 2008)*, pp 14-22, Milan, Italy, 7-10 September 2008

The present author was an equal partner in the research presented in this paper and in the writing.

Appendix O

D. Bem, E. Huebner “Generating computer forensics awareness in exploratory learning of personal computing”

Published in *Recent Advances in Computing and Management Information Systems*, ed. P. Petratos & G.A. Marcoulides, Athens, Greece, 2009

The present author was an equal partner in the research presented in this paper and in the writing.