University of Pretoria etd – Fei, B K L (2007)

DATA VISUALISATION IN DIGTIAL FORENSICS

BENNIE KAR LEUNG FEI

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE MAGISTER SCIENTIA (COMPUTER SCIENCE) IN THE FACULTY OF ENGINEERING, BUILT ENVIRONMENT AND INFORMATION TECHNOLOGY, UNIVERSITY OF PRETORIA, SOUTH AFRICA.

University of Pretoria etd – Fei, B K L (2007)

Data Visualisation in Digital Forensics

by

B.K.L. Fei

Abstract

As digital crimes have risen, so has the need for digital forensics. Numerous state-of-the-art tools have been developed to assist digital investigators conduct proper investigations into digital crimes. However, digital investigations are becoming increasingly complex and time consuming due to the amount of data involved, and digital investigators can find themselves unable to conduct them in an appropriately efficient and effective manner. This situation has prompted the need for new tools capable of handling such large, complex investigations. Data mining is one such potential tool. It is still relatively unexplored from a digital forensics perspective, but the purpose of data mining is to discover new knowledge from data where the dimensionality, complexity or volume of data is prohibitively large for manual analysis.

This study assesses the self-organising map (SOM), a neural network model and data mining technique that could potentially offer tremendous benefits to digital forensics. The focus of this study is to demonstrate how the SOM can help digital investigators to make better decisions and conduct the forensic analysis process more efficiently and effectively during a digital investigation. The SOM's visualisation capabilities can not only be used to reveal interesting patterns, but can also serve as a platform for further, interactive analysis.

Keywords: Digital forensics, digital investigation, self-organising map, analysis, visualisation

Degree: Magister Scientia

Supervisor: Prof. J.H.P. Eloff

Co-supervisors: Dr. H.S. Venter and Prof. M.S. Olivier

Department of Computer Science

Acknowledgements

I wish to express my deepest gratitude to:

- My supervisors, Prof. Jan Eloff, Dr. Hein Venter and Prof. Martin Olivier for their professional supervision.
- Prof. Andries Engelbrecht, for his assistance and useful insight.
- My dearest family, especially Paul and Judy, for their forbearance, encouragement and continuous support.
- Shan, for her encouragement and support.
- Christina, my dear friend, for her inspiration and support.
- My friends, especially the members of the ICSA research group, for their assistance and ideas.
- University of Pretoria, for the use of their facilities.
- The National Research Foundation for the financial support.

University of Pretoria etd – Fei, B K L (2007)

Contents

University of Pretoria etd – Fei, B K L (2007)

University of Pretoria etd – Fei, B K L (2007)

University of Pretoria etd – Fei, B K L (2007)

University of Pretoria etd – Fei, B K L (2007)

List of figures

University of Pretoria etd – Fei, B K L (2007)

List of tables

University of Pretoria etd – Fei, B K L (2007)

Corroborating material

A CD-ROM has been included at the end of this dissertation, containing supporting data. The contents of this CD-ROM include the following:

- This dissertation in electronic format (fei_dissertation_2006.pdf).
- Adobe Acrobat Reader 7.0 for Windows (AdbeRdr70_enu_full.exe).
- Microsoft .NET Framework Version 1.1 Redistributable Package (dotnetfx.exe).
- The SOMFA Tool (SOMFA.exe)

University of Pretoria etd – Fei, B K L (2007)

Chapter 1

Introduction

In response to the continuous increase in digital crimes, research is being conducted into ways of improving the quality and efficiency of digital investigations. An important reason for the proliferation of digital crimes has been the remarkable growth of the Internet. The Internet began in 1969 when the Advanced Research Project Agency (ARPA), which later became the Defense Advanced Research Project Agency (DARPA), linked four mainframe computers to experiment with the potential of a network. Growth has accelerated ever since: in 1984 there were in excess of a thousand computers connected together, by 1989 the number had reached 100,000, and the latest estimates are in the millions (Dix et al., 1998).

Today, countless personal and business transactions are conducted electronically. People use the Internet for numerous reasons that include: chatting; e-mailing; transferring and sharing files; searching for information; exchanging ideas through various forums and playing games. The Internet offers computer users access to a wealth of information and reaches into the hearts of many organisations. The Internet continues to evolve and offer potential digital criminals increased opportunity through communication capabilities that did not exist previously. Nonetheless, its evolution also provides equally many new sources of potential evidence of digital crimes.

Internet-related crimes are on the rise (CERT Coordination Center, 2006). Furthermore, excessive usage of the Internet for non-job purposes and even blatant misuse of the Internet (for example, employees accessing Web sites that promote pornography or other illegal activities) has become a problem for many organisations.

Pornography has become a huge business and causes problems for numerous organisations. Not only is it widely available on the Internet, but it is often unavoidable as pornographers take advantage of the latest Internet technology to bombard computer users with unsolicited e-mail and unwanted pop-up advertisements. While some forms of pornography may be more distasteful and intrusive than illegal, child pornography is a serious offence and a growing problem for digital investigators. Nonetheless, the posting of child pornography on the Internet often leads digital investigators to the victims because, along with threatening letters, fraud and theft of intellectual property, it is a crime that leaves digital tracks (Kruse II and Heiser, 2002).

The data held on computer systems and networks can tell us a lot about an individual's interests, patterns of behaviour and even their whereabouts at a specific time. As computer systems, networks and other computing devices become more widely used and prevalent, the chances of such computing devices and networks being involved in criminal activity will also naturally increase. It is due to this increase in crimes and incidents relating to the Internet and computing devices that the field of digital forensics has rapidly emerged. The concepts and ideas behind digital forensics are well established, but the discipline is still a nascent one. It addresses the specific need to be able to extract legally admissible evidence from computer systems, networks and other computing devices that can be used to successfully prosecute digital criminals.

Digital forensics can be classified into two key areas, namely, computer and network forensics. In addition to this, in some contexts (Barbara, 2005; Marsico, 2005), digital forensics can be classified into other areas such as video, image and audio forensics. Computer forensics identifies evidence that particular computers have been used in the perpetration of specific crimes (Marcella and Greenfield, 2002) and network forensics serves both as a means of preventing attacks or 'hacks' into systems and a means of inspecting for potential evidence after an attack or incident has occurred (Mukkamala and Sung, 2003).

1.1 Motivation

As digital crimes continue to rise, the need for digital forensics also increases. Digital forensics is utilised to conduct investigations into digital crimes or incidents. The aim of such investigations is to expose and present the truth, which often leads to prosecution and conviction.

Dramatic increases in the numbers of digital crimes committed have led to the development of a whole slew of computer forensic tools. These tools ensure that digital evidence is acquired and preserved properly and that accuracy of results regarding the processing of digital evidence is maintained (Marcella and Greenfield, 2002). Such tools exist in the form of computer software and have been developed to assist digital investigators conduct a digital investigation.

1.2 Problem statement

Digital forensics as a discipline faces several problems, among the more acute and limiting are the following:

- Digital investigations are becoming more time consuming and complex as the volumes of data requiring analysis continue to grow (Davis et al., 2005; Stephenson, 2003).
- Digital investigators are finding it increasingly difficult to use current tools to locate vital evidence within the massive volumes of data (Slay and Jorgensen, 2005). As a result, it is difficult for digital investigators to conduct the forensic analysis process in an effective and efficient manner, despite using state-of-the-art computer forensic tools (Roussev and Richard III, 2004).
- Log files are often large in size and multi-dimensional, which makes the digital investigation and search for supporting evidence more complex.

These problems have prompted the following research questions for this study:

- Are there ways to improve the efficiency and quality of forensic analysis?

As stated previously, digital investigators are facing major challenges created by the increasingly large volumes of data available. Under these circumstances, it is often impractical to perform a complete forensic analysis due to time constraints and limited human resources. As a result, uncovering new methods that will improve the quality of the decisions made, lessen the human processing time required and reduce the monetary costs of digital investigations is of paramount importance.

- What are the current capabilities of computer forensic tools?

  At present, computer forensic tools are unable to present a visual overview of all the data found on storage media. Yet this visual overview can prove crucial in a digital investigation. It will reveal the overall pattern of the data set, which can help digital investigators decide what steps to take next in their search. Furthermore, it can assist digital investigators in locating their points of interest. Both of these enhancements will improve the efficiency and quality of the forensic analysis being conducted.

  The data offered by computer forensic tools can often be misleading due to the dimensionality, complexity and amount of the data presented. In such cases, the data generated by computer forensic tools is effectively meaningless.

- Can the complexity that exists within log files be reduced?

  In terms of network forensics, the digital evidence recovery process may require data to be gathered from a variety of sources, whether at the server, proxy or other levels. The required data often resides in the logs of the equipment dispersed throughout the network, for example, routers, switches, firewalls, Web servers, Web proxies. Consequently, the forensic analysis of logs plays a major role in modern computer security and digital forensics. Current network forensics practices involve manually examining these logs, which can be a time-consuming and error-prone process (Wang and Daniels, 2005). Therefore, developing methods or tools for reducing the dimensionality, complexity or volume of log data is a matter of some urgency.

In addition to the research questions for this study, the following broader questions are also considered:

- Have there been developments in other disciplines that could benefit digital forensics? Moreover, if there are any developments pertinent to the proposed solution, what are they?

1.3 Research methodology

The first step will be to asses the current state of the field of digital forensics by thoroughly exploring all the available literature. Subsequently, some of the commonly deployed computer forensic tools will be investigated. The results of this investigation will be used to create a basic classification of computer forensic tools. From this classification, the limitations of current computer forensic tools will be identified and recommendations for improvements will be made.

Based on the recommendations, a solution will be proposed and tested against the problems identified in the previous section. Finally, experiments will be conducted to confirm that the proposed solution will appropriately benefit the field of digital forensics.

1.4 Terminology

Some of the terms commonly used within the context of digital forensics are defined below to avoid any potential misunderstandings:

- **Digital forensics** is the use of scientific methods for the identification, preservation, extraction and documentation of digital evidence derived from digital sources to enable successful prosecution (Kruse II and Heiser, 2002).
- A **digital investigation** is the application of scientific methods to extract data of evidentiary value from digital sources.
- The **self-organising map** is a neural network model for mapping high-dimensional data to a low-dimensional space (Kohonen, 1990).
- **Analysis** is the process of interpreting data and placing it in a logical and useful format.

- **Visualisation** is the graphical representation of data to provide a qualitative understanding of the data.

1.5 Layout

Chapter 1 has introduced the research problem and the motivation for this study. As part of this introduction, a detailed, specific problem statement has been outlined and an appropriate research methodology suggested. Chapter 2 expands the discussion to include background information on the field of digital forensics; this includes its history and an overview of the two sub-disciplines of digital forensics, namely, computer and network forensics. Chapter 3 concludes the initial assessment of research problem by examining some of the more commonly deployed computer forensic tools and providing a basic classification of said tools. Based on this classification, limitations and recommendations are identified.

Chapter 4 uses a discussion of the field of data mining and specific data mining techniques to identify a potential solution to the limitations established previously. Chapter 5 furthers this by taking an in-depth look at the data mining technique in question, the neural network model of the self-organising map.

A prototype implementation of the proposed method is discussed in Chapter 6. The outline of the implementation includes its architecture and a functional overview. Chapter 7 assesses the prototype implementation from an experiential level and discusses the findings of various experiments applied to the implementation. Finally, Chapter 8 concludes the investigation with a summary and suggestions of future work.

University of Pretoria etd – Fei, B K L (2007)

Chapter 2

Digital forensics

Law enforcement agencies in the United States began working together to deal with the growth in digital crime in the late 1980s and early 1990s (Noblett et al., 2000). Rapid technological development, in conjunction with the increasing volume and sophistication of digital crimes, has made the need for digital forensics more and more acute. A situation that is unlikely to change in the foreseeable future as the number of incidents is continuously rising each year. Figure 2.1 shows the increase in the number of incidents reported each year by the CERT Coordination Center (CERT/CC) (CERT Coordination Center, 2006).

Digital forensics, in essence, answers the *when*, *what*, *who*, *where*, *how* and *why* concerning a digital crime (Beebe and Clark, 2004). When conducting an investigation on a computer system, for example, the '*when*' refers to the time interval the activities took place during. The '*what*' concerns the activities performed on the computer system. The '*who*' concerns the person responsible, the '*where*' refers to where the evidence is located, the '*how*' addresses the manner in which the activities were performed, and the '*why*' seeks to ascertain the motives behind the crime.

The aim of this chapter is to acquaint the reader with the terms and concepts used in the field of digital forensics. In the first section, digital evidence is discussed in detail. Subsequent to that, an overview of digital forensics is

provided. This includes the field's history and the two sub-disciplines of digital forensics, namely, computer and network forensics.

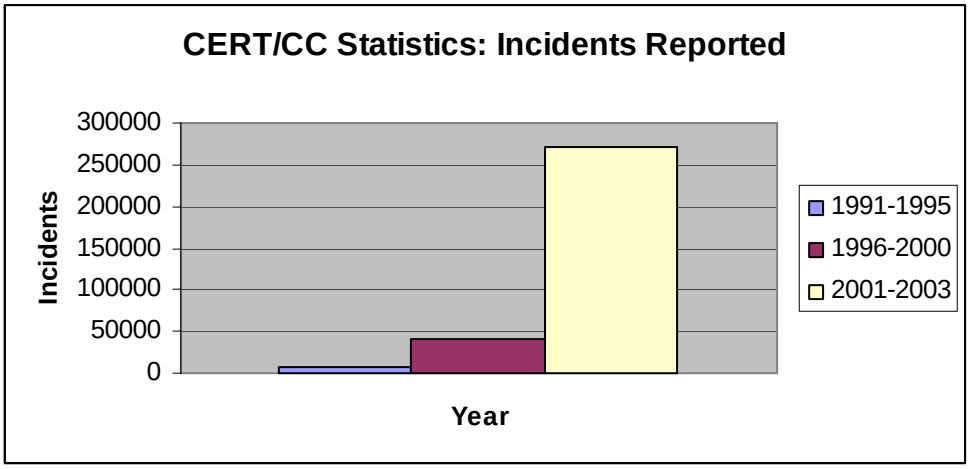


Figure 2.1: CERT/CC statistics for incidents reported.

2.1 Digital evidence

Today, an increasing amount of criminal evidence resides on a computer, even if the majority of such evidence is still used to commit traditional or conventional crimes. An example would be a threatening letter sent in electronic format, such as an e-mail or text file, instead of a traditional paper format.

Digital evidence by definition is information of probative value stored or transmitted in digital form (Pollitt, 2001). It is somewhat unique when compared to other forms of documentary evidence. For instance, it may be found in unusual locations, ones that are unknown to general computer users. It is also fragile in nature and can easily be altered or destroyed.

Another unique characteristic of digital evidence is that it can be duplicated. Consequently, it must be properly preserved, while analysis is performed on a duplicate copy, to ensure the original evidence is still acceptable in a court of law (Meyers and Rogers, 2004).

As technology is becoming more advanced and more integrated in everyday life, new challenges are constantly emerging. Since the appearance of computers,

sources of potential digital evidence have expanded to include areas such as networks (Casey, 2004b), mobile devices (Mellars, 2004; Willassen, 2003; Willassen, 2005), game consoles (Vaughan, 2004), digital media devices (Marsico, 2005; Marsico and Rogers, 2005) and many more.

Network traffic, for example, is a source of digital evidence that presents numerous challenges (Casey, 2004b). This is due to the limited opportunity for capturing network traffic. Adequate data capturing systems must be in place as data travels through a network otherwise the opportunity is lost. Furthermore, when dealing with network traffic, it is often difficult to locate and extract specific items from the large number of flows on a network.

Sources of digital evidence include (but are not limited to):

- compact discs (CDs);
- computer systems;
- digital cameras;
- digital media devices such as the iPod nano (Apple Computer, 2006) and Zen Vision (Creative Technology, 2006);
- digital versatile discs (DVDs);
- flash drives;
- floppy disks;
- game consoles such as the Xbox 360 (Microsoft Corporation, 2006);
- memory cards;
- mobile phones;
- network devices such as routers and switches;
- network traffic;
- notebooks;
- personal digital assistances (PDAs);
- zip disks.

When searching for evidence on storage media, there are various types of data to look for. The two main types of data, known as active and residual data, are discussed below.

2.1.1 Active data

Active data (Motion, 2005) is data residing on storage media that is readily visible to the operating system and accessible to computer users. Such data includes word processing and spreadsheet files and programs and operating system files. The latter including temporary files, temporary Internet files, cookie files, and file system metadata to name but a few.

2.1.2 Residual data

Residual data (Motion, 2005) is data that appears to have been deleted, but can still be recovered. The most common examples are swap files and file fragments found in slack or unallocated space. For example, when a computer user deletes a file, only the information that points to the location of the file on the media will be erased, therefore, the data still exists.

2.2 Digital forensics defined

Forensics involves the investigation of evidence by following scientific methods within the regulations of the law (Vacca, 2002). Digital forensics applies these principles to the process of digital evidence recovery. It is a sub-discipline of forensics that can be traced back more than two decades (Inman and Rudin, 2001). It can be classified into two key areas, namely, computer and network forensics. In general, computer forensics deals with data in a computer (Reith et al., 2002), whereas network forensics deals with data that may be spread over several databases residing on computers in one or more networks (Caloyannides, 2004). A broad overview of computer forensics is depicted in Figure 2.2 and a similar overview of network forensics is depicted in Figure 2.3.

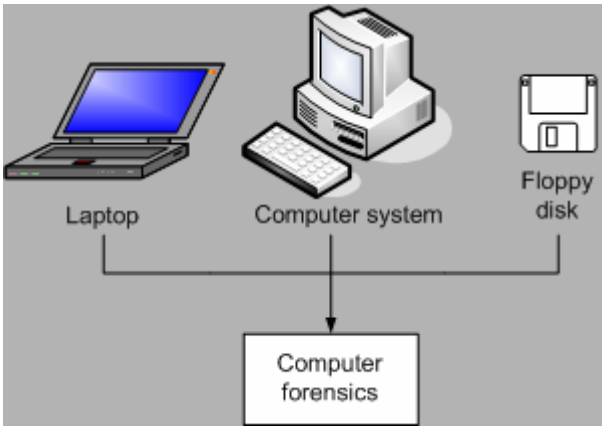


Figure 2.2: A broad overview of computer forensics.

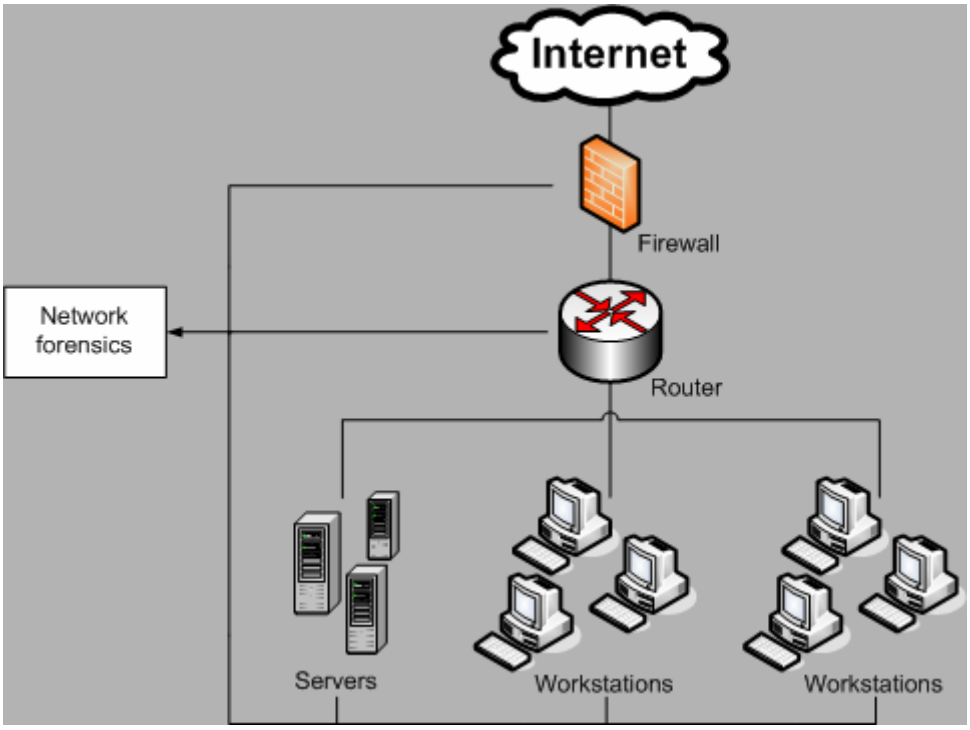


Figure 2.3: A broad overview of network forensics.

2.2.1 Computer forensics

Computer forensics can be traced back to as early as 1984 when the Federal Bureau of Investigation (FBI), as well as other law enforcement agencies, began

developing programs to assist in the examination and analysis of digital evidence (Noblett et al., 2000).

Computer forensics can be referred to the forensic examination of computer components and their contents (Casey, 2004a). These computer components can be printers and storage media such as hard drives or CDs. In general, computer forensics is used to identify evidence when personal computers are used in the commission of crimes (Marcella and Greenfield, 2002). In more practical terms, computer forensics deals with the identification, preservation, extraction and documentation of digital evidence (Marcella and Greenfield, 2002). These phases form the backbone of the digital investigations process and are discussed later on.

Computer systems and networks have one thing in common in terms of forensics: both can contain evidence that can be used to verify a crime has been committed and who the perpetrator is. However, the term computer forensics has different meanings in different contexts. For instance, in some contexts the term is referring to the forensic examination of all forms of digital evidence, including data over networks. With the rapid development of network technology and the increasing usage thereof, evidence is no longer confined to storage media. In response, network forensics has emerged as a sub-discipline of digital forensics and deals with data over networks (Caloyannides, 2004).

The unique needs of computer forensics have resulted in the creation of computer forensic tools in the form of computer software. These tools ensure that digital evidence is acquired and preserved properly to maintain the integrity of digital evidence. For example, copying and pasting data onto another storage medium may not be admitted in a court of law as forensically sound evidence (Wang et al., 2005b). This is because the process of copying and pasting data can modify it, for example, altering the timestamps of the data. As a result, a typical digital investigation requires the making of an exact bit by bit (or bit stream) copy of all the data on a storage medium. This exact bit by bit copy is called an image and the process of making an image is frequently referred to as imaging (Sammes and Jenkinson, 2000).

In most cases, once the imaging process has been completed, it is essential to have a mechanism or procedure to determine whether the evidence has been altered. This is to ensure the integrity of the evidence (Gollmann, 1999). A

common practice is to use a hash algorithm such as the message digest version 5 (MD5) (Rivest, 1992) or secure hash algorithm version 1 (SHA-1) (Eastlake III and Jones, 2001) to verify the integrity of the evidence.

Hash functions are used by digital investigators in two ways. First, hash functions can verify whether a file has been altered (Thompson, 2005). Second, they can serve as authentication since it is extremely unlikely that two files will have the same hash value. For instance, the hash value for a specific word processing file is "82E22E78711B5960AE6D8B61000AEA3A". If a blank space is added to the file, the new hash will be "65435F8ADEB763F1F39918664D654F2A". In other words, any alterations made to the file will result in the change of the hash value.

Once an image has been successfully acquired, it must be analysed to extract the evidence that the digital investigators wish to present. Analysis of digital evidence involves a mixture of techniques. For example, one technique is to perform keyword searches on digital evidence (Schweitzer, 2003). Other techniques include file signature analysis and hash analysis (Casey, 2002).

A last point to note is that the field of computer forensics has advanced from carrying out analysis within a command-line environment, such as DOS, to a graphical environment, such as Windows. Each environment has its own advantages and disadvantages.

The remainder of this section describes the four phases that form the backbone of the digital investigation process: identification; preservation; extraction, and documentation.

Identification

Every digital investigation begins with the identification of digital evidence followed by its preservation and extraction. The identification of digital evidence requires two steps. First, digital investigators have to identify the sources of data and, second, they must identify the relevant, legally admissible data available on those sources.

At the beginning of the investigation, the digital investigators must identify the sources of data. For example, this can include floppy disks, CDs, DVDs, zip

disks, memory cards, flash drives, PDAs or complete computer systems. Once identified, the task of acquisition begins. Acquisition is the process in which digital evidence is duplicated or imaged and is performed in order to avoid tampering with the original evidence.

Once the sources have been identified, digital investigators must differentiate between the relevant and irrelevant data since different crimes result in different types of digital evidence. For example, child pornographers will have graphical images or video files stored on their computer systems and fraudsters will have data such as company information stored on their computer systems.

Preservation

Once evidence has been recovered, there are strict requirements that must be met to preserve the evidence for later use in a court of law. If evidence is not preserved properly, it may turn out to be inadmissible in a court of law (Meyers and Rogers, 2004). Preservation involves the isolation, securing and safeguarding of digital evidence (Baryamureeba and Tushabe, 2004).

Digital evidence must be preserved to prevent alterations. As mentioned earlier, to verify the integrity of the evidence, a hash algorithm such as the MD5 or SHA-1 is used. This makes it difficult for a defence attorney to argue that the evidence was tampered if the hash value matches that of the original evidence. The way to properly preserve evidence physically is to place the storage media in anti-static bags and keep them away from magnetic fields.

Extraction

Extraction of digital evidence can be a lengthy and complex process. The requirements of digital investigations will differ from one another, depending on the unique nature of the investigation. For example, in the case of child pornography, the investigation involves the extraction of all graphical images or video files from the suspect's computer system. These extracted graphical images or video files will then be analysed to determine whether relevant to the case at hand. If the digital investigator knows what they are searching for, it is likely that the evidence can be extracted in a quicker manner. However, if the evidence is

hidden, for example, if the files that contain vital information have been deleted, it is possible to recover the deleted files, but it is likely to take longer. Even if a deleted file is partially overwritten, file fragments can be found in slack or unallocated space that can still be extracted and reconstructed to their original state.

The process of extraction involves the analysis of digital evidence. Analysis is the process of interpreting data and placing it in a logical and useful format. Once the data has been successfully gathered, it must be analysed to extract the evidence that is to be presented in a court of law. There are various types of analysis, for example, timeframe, data hiding, application and file, and ownership and possession analysis (National Institute of Justice, 2004).

Documentation

Documentation is an ongoing process throughout a digital investigation. It is the process through which digital evidence is documented as it is found and collected. It also includes recording the critical details of each step of the digital investigation, such as procedures followed and methods used to seize, document, collect, preserve, recover, reconstruct, organise, and search for key evidence.

As mentioned previously, digital forensics answers the *when*, *what*, *who*, *where*, *how* and *why*. Therefore, these concerns must be documented during each step of the digital investigation. Evidence documented clearly and accurately is crucial and often concludes the outcome of the prosecution. For example, documentation showing evidence in its original state and unaltered can create a solid case against the perpetrator. In addition, documentations can be useful when trying to reconstruct a crime and reconstructing digital crimes helps digital investigators to defend their hypotheses about why such evidence exists. This is particularly important since perpetrators often claim the evidence was planted by someone else and deny any wrong doing.

2.2.2 Network forensics

Network forensics was introduced in the early ‘90s (Mukkamala and Sung, 2003). It is an attempt to prevent attacks into systems and to search for potential

evidence after an attack or incident has occurred. Such attacks include probing, denial of service (DoS), user-to-root (U2R) and remote-to-local (R2L).

The network forensic discipline spans numerous activities and analytical techniques. For example, the analysis of the logs of intrusion detection systems (IDSs) (Sommer, 1999), the analysis of network traffic (Casey, 2004b) and the analysis of network devices themselves (Petersen, 2005) are all areas considered to be part of network forensics.

Network forensics involves capturing, recording and analysing network audit trails in order to discover the source of security breaches or other information assurance problems (Mukkamala and Sung, 2003). A practical approach is to archive all traffic and analyse subsets as necessary (Corey et al., 2002). In addition, digital investigators are required to be familiar with the underlying technology involved in digital crimes that occur within networks.

As mentioned earlier, computer systems and networks have one thing in common: that both areas may contain potential evidence after a crime has been committed or after an incident has been occurred. If, for instance, an attacker attacks a network, the attack traffic usually goes through a router. As a result of this, important information or evidence might be found by examining the router's logs. The forensic analysis of logs has always played an important role in modern computer security and it is increasingly playing an essential role in network forensics as well since logs now reside on many different sources such as routers, switches, IDSs, firewalls, Web servers and Web proxies.

Evidence can be gathered from various sources depending on the unique requirements and nature of the investigation. It can be gathered at the server level, proxy level or from several other sources. For example, at the server level, evidence can be gathered from Web server logs as they record the browsing behaviour of site visitors. These logs reflect which users have accessed the site and how. Other sources include the contents of network devices and traffic that passes through both wired and wireless networks. For example, evidence can be gathered from the usage data extracted by using packet sniffers (such as tcpdump (Jacobson et al., 2006)) to monitor incoming network traffic. This network traffic could be used to verify whether a suspect has been distributing child pornography, for instance.

2.3 Conclusion

Digital forensics is utilised to conduct investigations into digital crimes or incidents in order to derive evidence. Digital evidence has increasingly played an important role in the prosecution of perpetrators. In this chapter, an overview of the two sub-disciplines of digital forensics, together with the terms and concepts related to digital forensics were given in order to lay the foundation for the upcoming chapters.

Secure collection of digital evidence can be accomplished by making use of computer forensic tools. The following chapter provides an overview of some of the most commonly deployed computer forensic tools. In addition to that, it aims to classify the different computer forensics tools into a basic catalogue.

University of Pretoria etd – Fei, B K L (2007)

Chapter 3

Computer forensic tools

Computer forensic tools have been developed to assist digital investigators in conducting proper investigations into digital crimes. Such tools ensure that digital evidence is acquired and preserved properly.

With the increasing number of computer forensic tools available on the market, it is important to be aware of the different features that exist within the domain. The focus of this chapter is to develop a better understanding of some of the more commonly deployed computer forensic tools. The first section provides an overview of the tools. The most important features are identified and categorised for each tool and used to create a basic classification of computer forensic tools. Based on this classification, limitations and recommendations are then identified and discussed.

3.1 Overview of computer forensic tools

Computer forensic tools are utilised to either collect or analyse digital evidence. Such tools exist in the form of computer software (Marcella and Greenfield, 2002). Examples include EnCase Forensic (Guidance Software, 2005), Forensic Toolkit (AccessData Corporation, 2005) and SafeBack (Armor Forensics, 2006). Some tools are designed with a single purpose in mind. For example, SafeBack is intended only for imaging. Others offer a whole range of functionalities, such as

searching capabilities and report generation. Furthermore, there are some tools that provide similar functionalities, but with a different graphical user interface. These are only some of the differences. Other differences exist in terms of complexity, usability, operating environment support and cost.

Computer forensic tools have advanced from command-line environments to sophisticated graphical user interfaces that significantly enhance investigative activities. Since computer forensic tools are increasingly available commercially, they need to be supported by tests carried out in a scientific manner. To that end, a framework for testing computer forensics tools has been defined by the National Institute of Standards and Technology (NIST) (National Institute of Standards and Technology, 2005). This framework defines both the functional characteristics and requirements of such tools and the tests that can be used to determine whether a particular tool meets the specified requirements.

The following subsections provide an overview of the more commonly deployed and well-known computer forensic tools that exist on the market. Please note that a comprehensive discussion of the tools available is beyond the scope of this document. The selection is based on a survey of the available literature (Altheide, 2004; Buchholz and Falk, 2005; Caloyannides, 2001; Casey, 2002; Kruse II and Heiser, 2002; Marcella and Greenfield, 2002; Middleton, 2001; Morris, 2002; National Institute of Justice, 2004; Scott, 2003; Vacca, 2002). It should be noted that the following tools are presented in no particular order. Furthermore, the following subsections aim at providing the reader with an overview of the tools' latest capabilities and are not intended to provide a detailed investigation into the tools. Each of the following subsections starts with background information about the tool and concludes with a functional overview. Following the overviews, a summary of the mentioned tools is provided.

3.1.1 EnCase Forensic

Background

EnCase Forensic, from Guidance Software, Inc. (Guidance Software, 2006), is one of the most well-known and commonly deployed computer forensic tools. It

is widely used by both law enforcement and computer security professionals worldwide.

EnCase Forensic is a Windows-based computer forensic tool that is intended to be utilised for gathering and evaluating digital evidence. It is one of the most expensive commercial tools available on the market. However, it offers a comprehensive range of functionalities and features a sophisticated graphical user interface.

Functional overview

At first glance, EnCase Forensic is a solid imaging tool. No flaws in the imaging engine were discovered when it was tested by the NIST (National Institute of Standards and Technology, 2005). A main feature of EnCase Forensic is the ability to preview media (such as hard drives) before acquisition. This ability allows digital investigators to determine whether it is necessary to conduct a full investigation on a piece of media or not. In addition to that, it enables digital investigators to view the contents of a piece of media without altering them in any way. Such contents include both active and residual data.

To operate EnCase Forensic, a security key is required. This security key (also known as a dongle) is a hardware device that controls access to the application. Once the tool is initiated, the investigator must create a new case to start the investigation (illustrated in Figure 3.1). Note that the current version of EnCase Forensic is version 4.14. The screen in Figure 3.1 allows the investigator to enter the information that is related to the case and set the path of the working folder.

The next step is to add the media for preview. Potential media sources include hard drives, flash drives or CDs. After adding the media, the investigator can immediately preview its contents and choose whether to conduct a full investigation of the device or not. A full investigation of the device or media requires performing the task of acquisition. In the demonstration detailed here, acquisition was performed on a flash drive. After the acquisition process, the MD5 hash value of the flash drive is generated (see Figure 3.2).

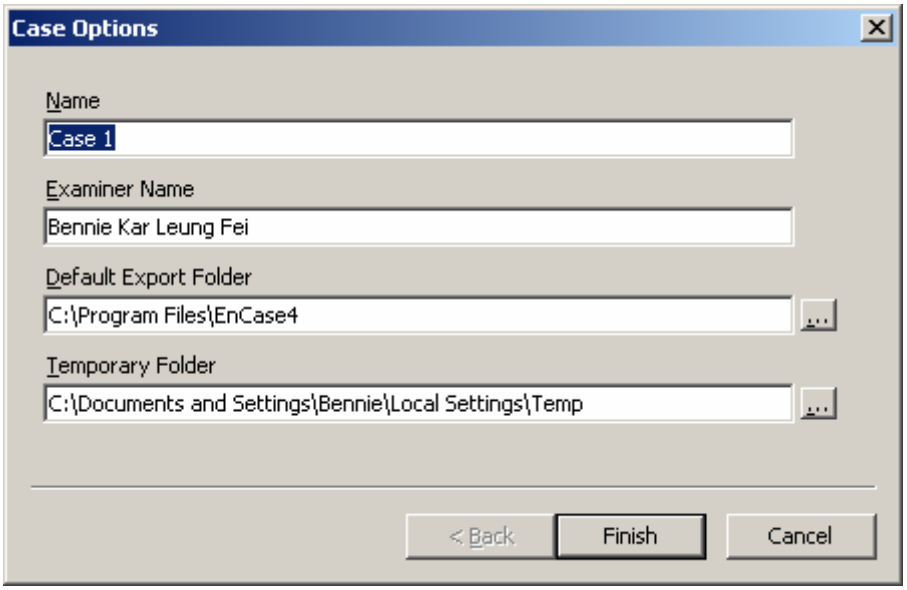


Figure 3.1: Creating a new case in EnCase Forensic.

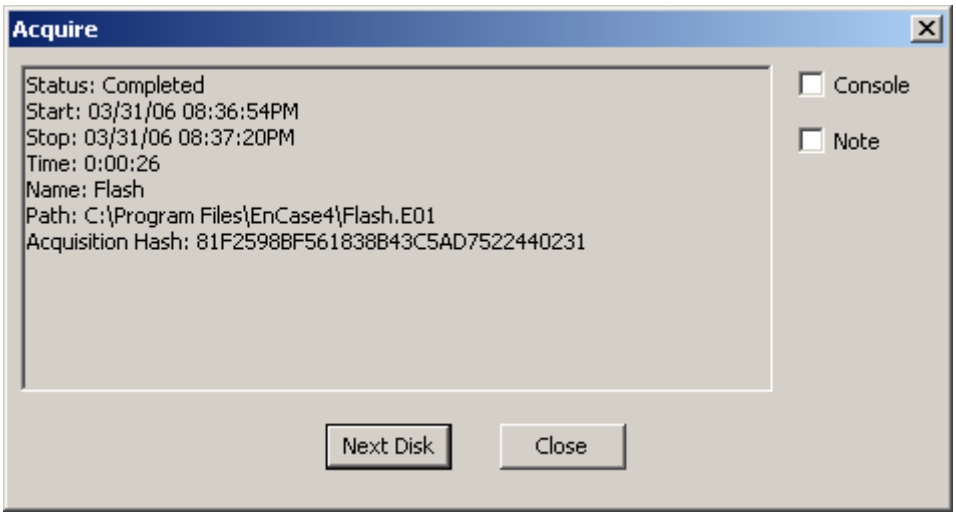


Figure 3.2: Results of the acquisition process for EnCase Forensic.

In terms of analysis, EnCase Forensic offers a whole range of functionalities including advanced searching capabilities, filtering capabilities, hash analysis, file signature analysis, registry analysis and many more. Furthermore, it features several views which are accessible through a tab-driven interface. These views are as follows:

- **Table view**: The Table view permits the investigator to view the files found on a device, including their attributes, in a spreadsheet-style format. These attributes may include file name, file extension, file type and file category (see in Figure 3.3).
- **Gallery view**: The Gallery view enables the investigator to view the graphical images found on a device. The graphical images are displayed as thumbnails, which facilitates quicker analysis and access. Once a graphical image is selected, the graphical image is then displayed in the viewer located at the bottom of the screen.
- **Timeline view**: The Timeline view provides a calendar view for looking at patterns of file creation, editing and last accessed times. In addition, it offers five check-boxes for filtering the kind of file activity that might be of interest (see Figure 3.4).
- **Report view**: The Report view presents various reports regarding the investigation (see Figure 3.5). This view can display information regarding the device including files and folders found on the device. In addition to that, it typically includes reference information about the acquisition, bookmarked files and bookmarked graphical images. The report view illustrated in Figure 3.5 displays information regarding the flash drive that was acquired. It includes general information regarding the flash drive, for instance, file system, sectors per cluster, total sectors, total clusters and so on. Furthermore, it includes other information regarding the investigation. Examples include the name of the investigator, the size of the flash drive, the total number of sectors in the flash drive, the version of EnCase Forensic, the operating system employed and, more importantly, the MD5 hash value of the flash drive.

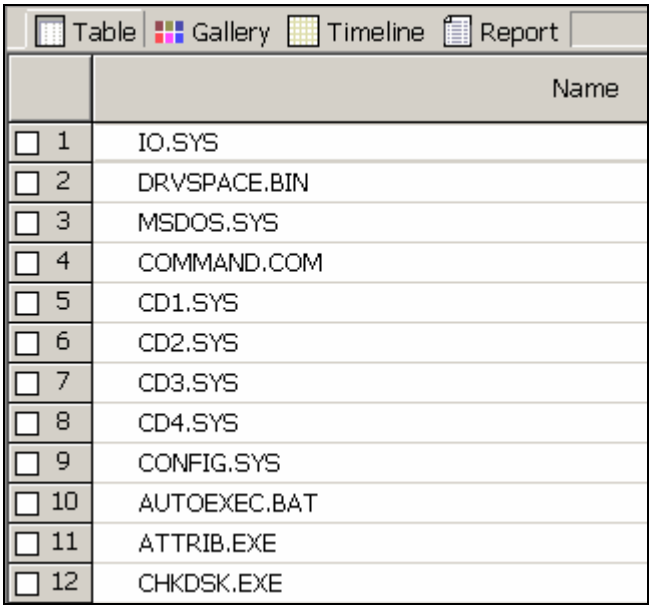


Figure 3.3: The Table view.

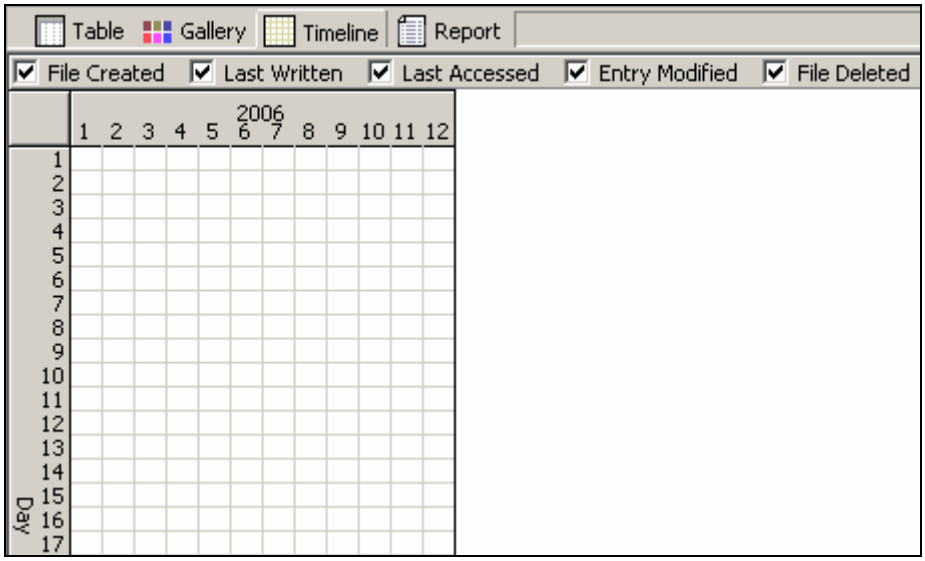


Figure 3.4: The Timeline view.

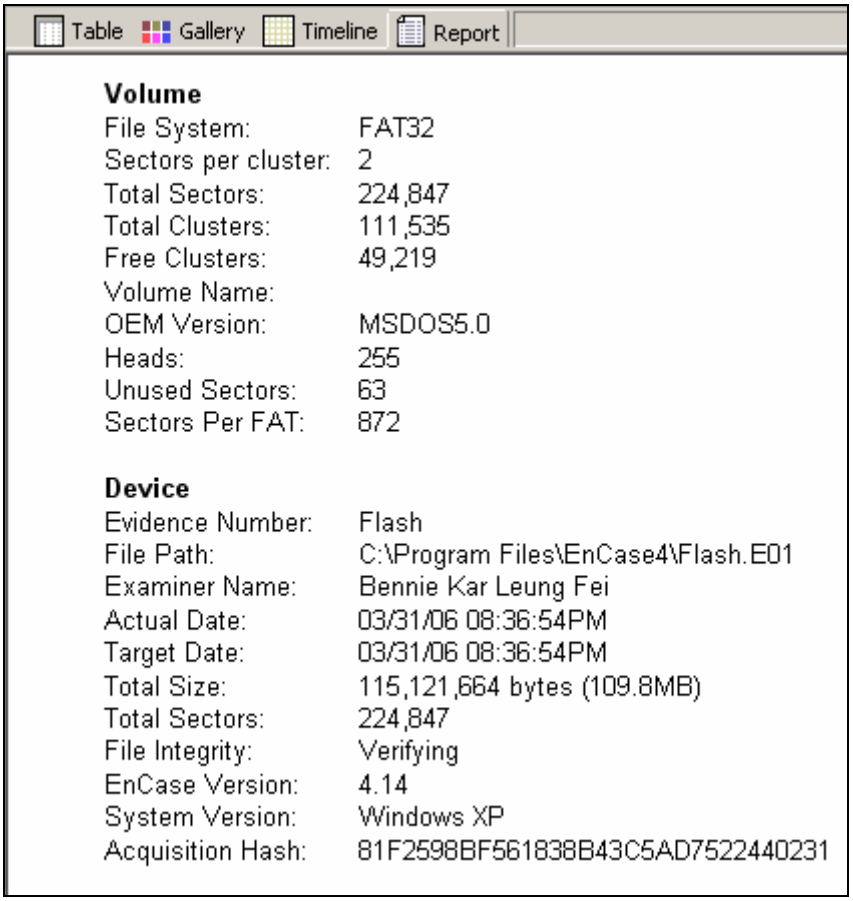


Figure 3.5: The Report view.

3.1.2 Forensic ToolKit

Background

Forensic ToolKit (FTK) from AccessData Corporation (AccessData Corporation, 2006) is another well-known and commonly deployed computer forensic tool. This tool is recognised as one of the leading tools for performing e-mail analysis (Prosise et al., 2003). Similarly to EnCase Forensic, FTK operates on a Windows platform such as Windows 2000 or Windows XP. However, in terms of cost, it is less expensive when compared to EnCase Forensic.

Functional overview

FTK is made up of several components, such as the FTK Imager, the Registry Viewer and the Known File Filter (KFF). It should be noted that each component requires its own installation and can be used on its own.

The FTK Imager is essentially an imaging component for acquisition although it has the ability to preview the source of the evidence, which can be a piece of media, an image file or the contents of a folder, as well. Integrated into FTK, the Register Viewer is a component for examining registry files. This includes searching for information in a registry file and accessing areas which contain passwords and other information. Lastly, the KFF is a component that can compare file hashes against a database of hashes from known files such as standard operating system and program files. It will identify all the known file formats automatically in order to reduce the time of the analysis process and ensure that the integrity of files remains unchanged.

Once FTK is launched, there is a choice to either preview or acquire the source of the evidence. It should be noted that a security key is required to operate FTK and the current version of FTK is version 1.61. Here, the source of the evidence is acquired using FTK Imager version 2.4 (a component of FTK). Once the source of the evidence has been acquired, the investigator is presented with the outcome of the acquisition process, which includes both the MD5 and SHA-1 hash values for verification (see Figure 3.6).

After acquisition, examination can begin by starting a new case. A new case is started using the new case wizard depicted in Figure 3.7. This wizard consists of several steps that allow the investigator to specify various options. For instance, the investigator has the ability to choose which processes will be performed (see Figure 3.8). These processes include computing both the MD5 and SHA-1 hash values. Processes like the “KFF Lookup”, which will identify all known files, and the “Full Text Index”, which will index all keyboard-related characters (to simplify the search process at a later stage), are also included as options. The wizard also enables the investigator to specify the source of the evidence, whether it is an image file, a local drive or a folder in a local drive.

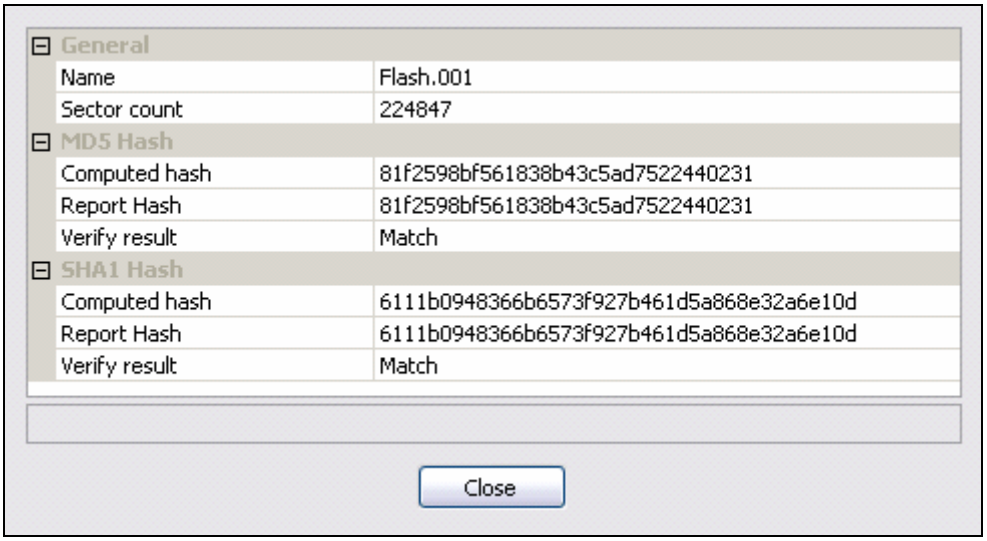


Figure 3.6: Results of the acquisition process for FTK.

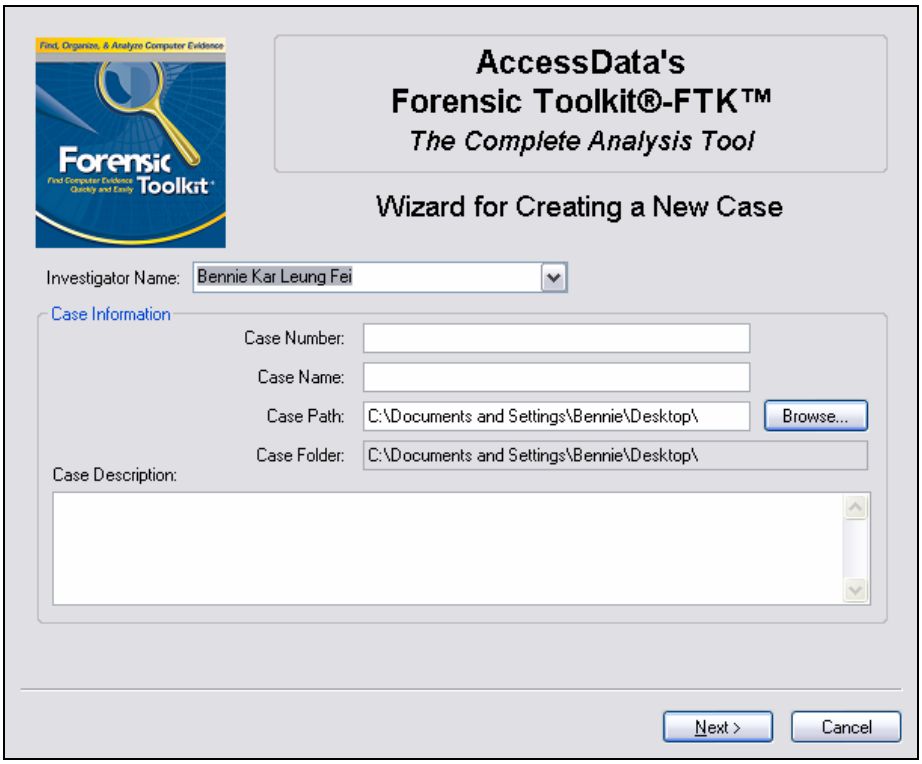


Figure 3.7: Wizard for creating a new case in FTK.

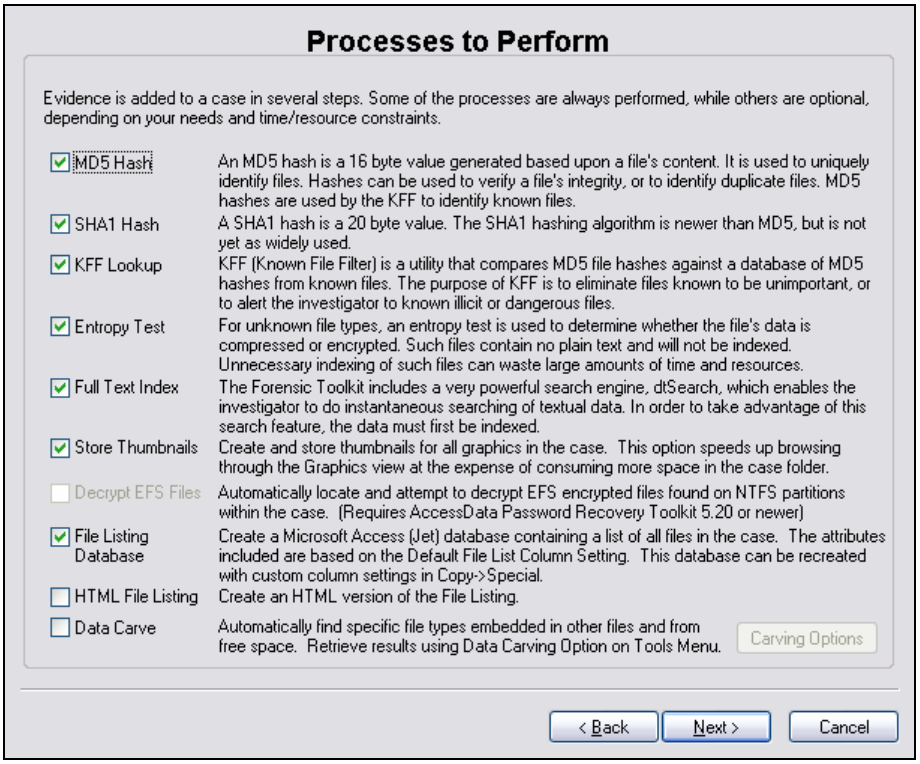


Figure 3.8: Processes to perform in FTK.

Once the wizard has been completed, FTK will start to process the evidence accordingly. After this process has been completed, analysis of the evidence can begin. FTK offers numerous analytical functionalities. From an interface perspective, FTK features a sophisticated graphical user interface, which includes six key tabbed windows, each with a particular focus or function. These windows essentially support the analysis and are discussed below:

- **Overview window**: The Overview window provides an overview of the case at hand. Here, files are presented in a spreadsheet-style format that enables the investigator to scroll and view those of interest. Furthermore, information regarding the source of evidence is presented in this window (see Figure 3.9). It should be noted that the files are categorised into various helpful classifications. For example, to see the deleted files, the investigator can simply click on the “Deleted Files” category and a list of deleted files will appear.

- **Explore window**: Using the Explore window, the investigator can view the hierarchical structure of files, folders and storage media on a computer system in a similar way to that used by Windows Explorer. In addition, it includes a viewer that can display the contents of a selected file as shown in Figure 3.10.
- **Graphics window**: The Graphics window displays graphical images as thumbnails which facilitates quicker analysis and access (see Figure 3.11). Note that the graphical window has the ability to display both deleted and undeleted graphical images. Furthermore, FTK also provides a viewer where the selected graphical image can be displayed.
- **E-Mail window**: The E-Mail window enables the investigator to view e-mail mailboxes, including their related messages and attachments. This includes both deleted and partially deleted messages. In addition, FTK supports Outlook, Outlook Express and many more e-mail clients. Figure 3.12 illustrates the E-Mail window listing three mailboxes, one of which contains a message.
- **Search window**: FTK offers two separate search modes: live and indexed. The live search involves an item-by-item comparison with search terms specified by the investigator, while the indexed search involves the use of a powerful search engine known as dtSearch (dtSearch Corporation, 2006). This search method creates a full index of the data and vastly speeds up keyword searches. Figure 3.13 illustrates executing keyword searches with the indexed search.
- **Bookmark window**: In this window, the investigator can view any items that have been bookmarked. Viewing the bookmarked items enables the investigator to review items important to the case. From here, the investigator can insert comments for each bookmark. In addition to that, the investigator can indicate whether to include the bookmarked items in the report. An example of the Bookmark window is shown in Figure 3.14. In Figure 3.14, the bookmarked items are all the deleted graphical images that will be included in the report.

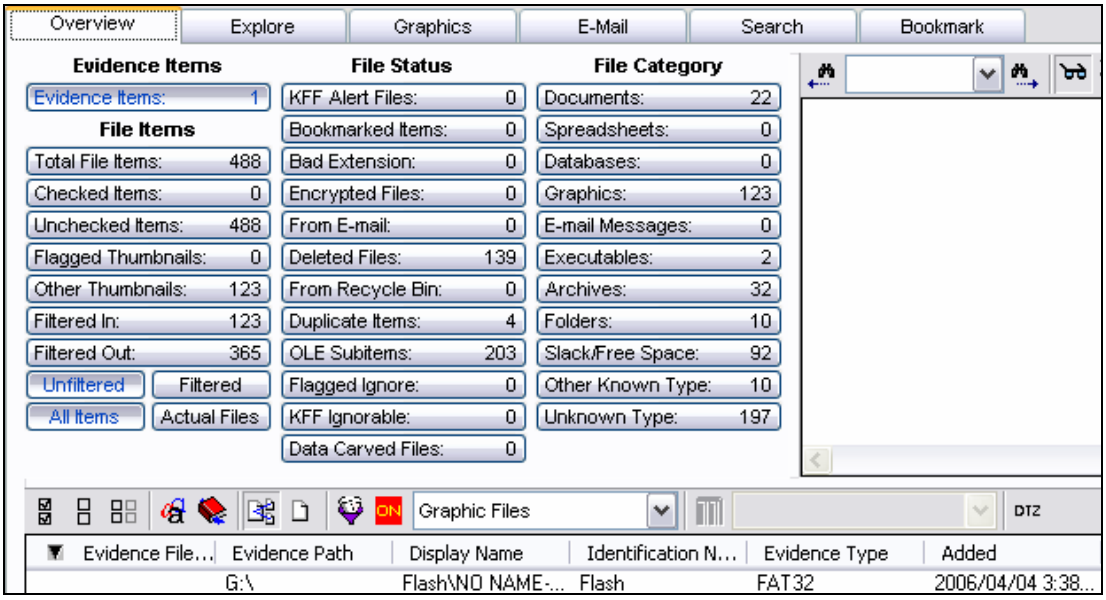


Figure 3.9: Overview window.

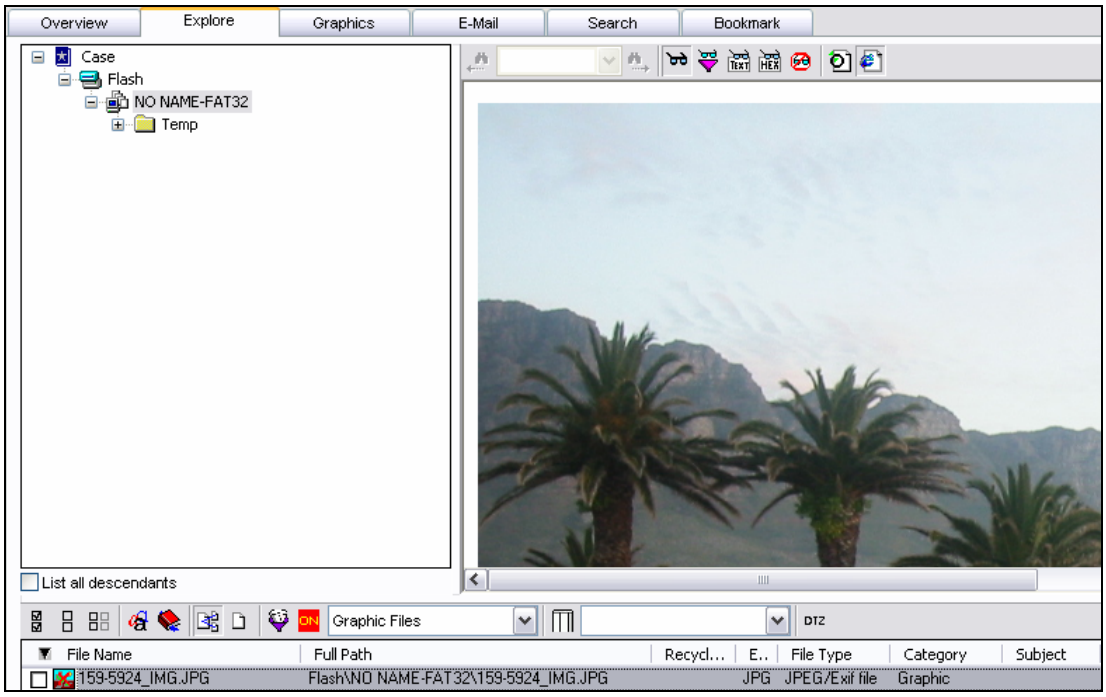


Figure 3.10: Explore window.

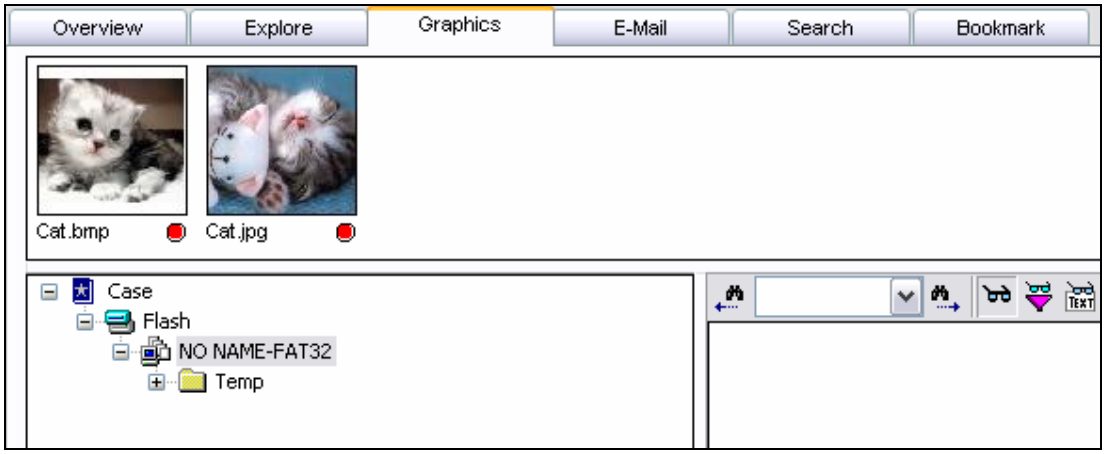


Figure 3.11: Graphics window.

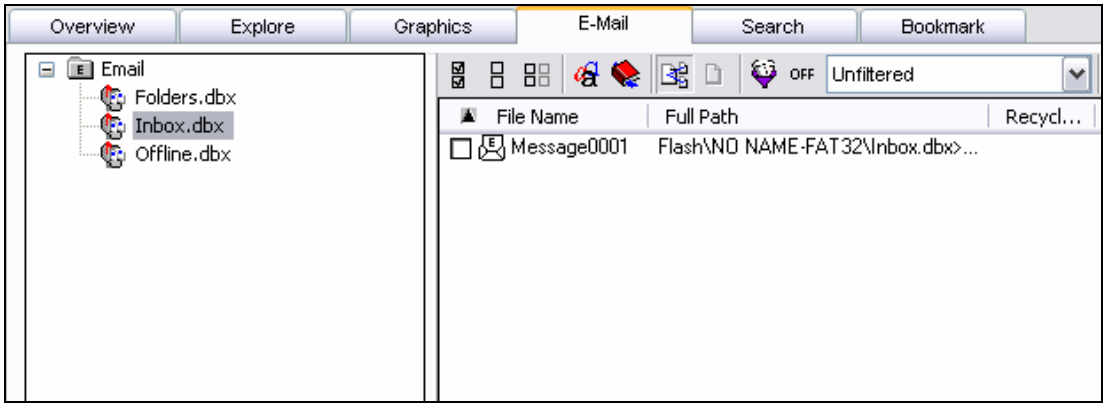


Figure 3.12: E-Mail window.

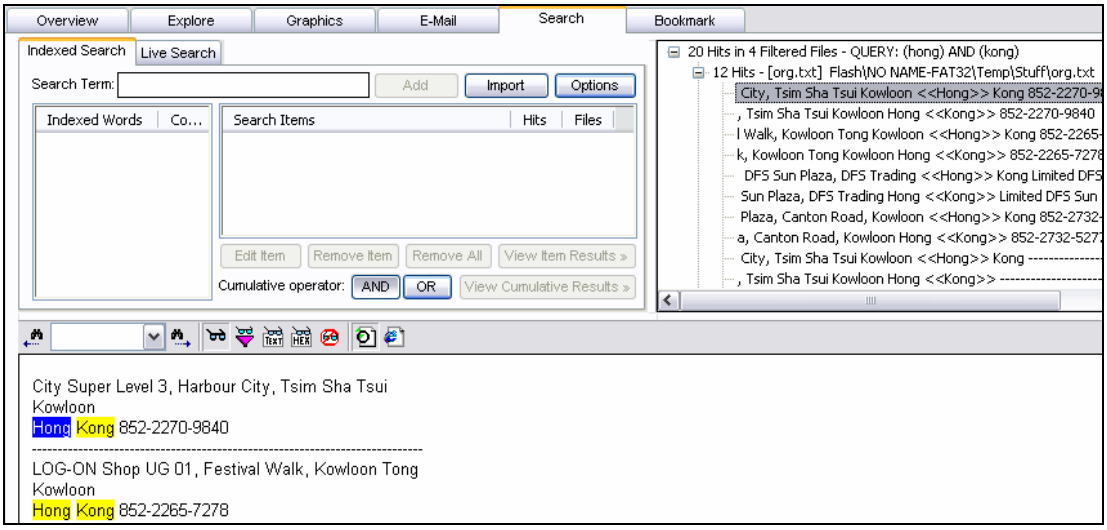


Figure 3.13: Search window.

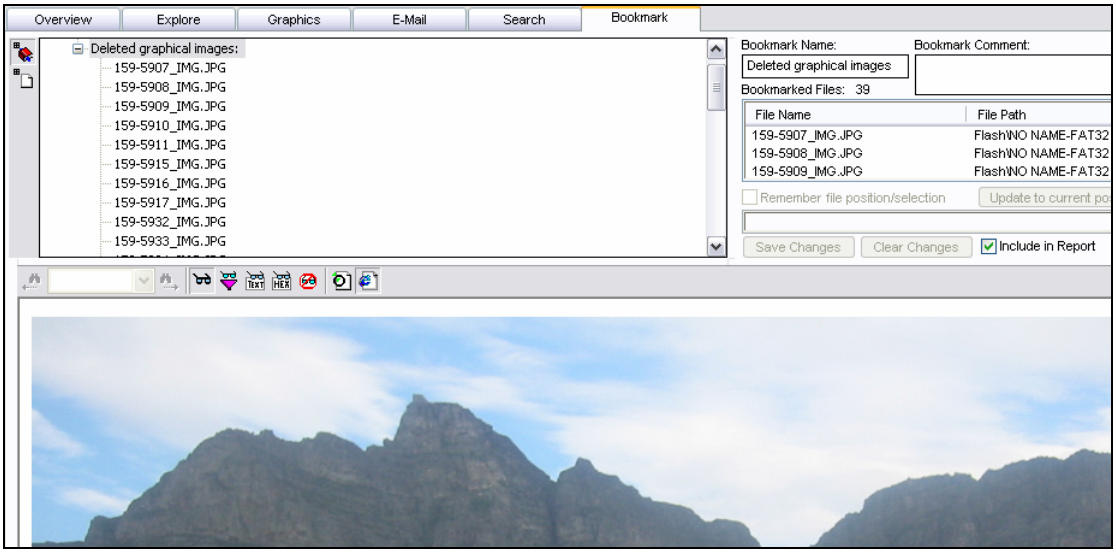


Figure 3.14: Bookmark window.

3.1.3 SafeBack

Background

SafeBack, currently version 3.0, from Armor Forensics (Armor Forensics, 2006), is another computer forensic tool commonly used by law enforcement, computer security and military professionals worldwide. It is a DOS-based computer

forensic tool that is capable of creating bit stream copies of media (National Institute of Standards and Technology, 2005). It is primarily used for imaging media and is not equipped with functionalities such as searching capabilities or report generation.

Functional overview

SafeBack utilises the SHA-256 algorithm (National Institute of Standards and Technology, 2002) to ensure the integrity of images. Furthermore, it offers a self-authenticating format for images, whereby SHA-256 hashes are stored along with the data.

It should be noted that a security key is not required to operate SafeBack and that it can be run from a floppy disk. To run Safeback, the disk must contain the file "Master.exe" which is the main SafeBack utility program. Typing "master" in the command-line will execute the utility and load an interface from which an investigator can select several options prior to starting the acquisition process. Once the desired storage media and name for the output file have been selected, the acquisition process will begin.

3.1.4 Storage Media Archival Recovery Toolkit

Background

Storage Media Archival Recovery Toolkit (SMART) is a Linux-based computer forensic tool from ASR Data Acquisition and Analysis, LLC (ASR Data Acquisition and Analysis, 2006). It has been utilised and accepted by law enforcement, computer security and military professionals worldwide. SMART features a simple, easy-to-use interface. In addition to that, it offers numerous functionalities similar to its counterparts such as EnCase Forensic and FTK.

Functional overview

SMART is primarily an imaging tool that uses the SHA-1 algorithm by default for creating hashes. In addition, it provides other algorithms such as the MD5

algorithm and the cyclical redundancy check 32 algorithm (Deutsch, 1996) to ensure the integrity of images. Tests have been performed to show that SMART is capable of creating a bit stream copy of media (Rude, 2002; Scott, 2003).

SMART offers preview, searching, report and even remote acquisition capabilities. To operate SMART, a security key is required. Once the tool is launched, it provides an overview of all the devices found. This overview includes information such as the size of each device, the file systems residing on each device and the residual data found on each device (see Figure 3.15).

To start with the acquisition process, simply right click on a device and select “Acquire”. The acquisition window will then appear. This window enables the investigator to choose options such as the hash algorithm to be used (shown in Figure 3.16). In addition, the investigator can specify the name of the target image, a description of the image and other details.

Once the image has been created, the investigator can begin the examination of the image. Such an examination usually includes searching through both active and residual data. This can be, for example, a simple keyword search which can be performed by right clicking on the image, selecting “Search” and adding the desired keyword. SMART offers a whole range of functionalities such as searching capabilities, filtering capabilities and hash analysis for performing analysis.

SMART features a built-in graphic viewer which allows investigators to view graphical images as thumbnails. The graphical images can be viewed in a slideshow. In addition, graphical images that are of interest can be flagged. Note that the flagged items will appear within the Report Builder and be available for selection and inclusion in a report if required.

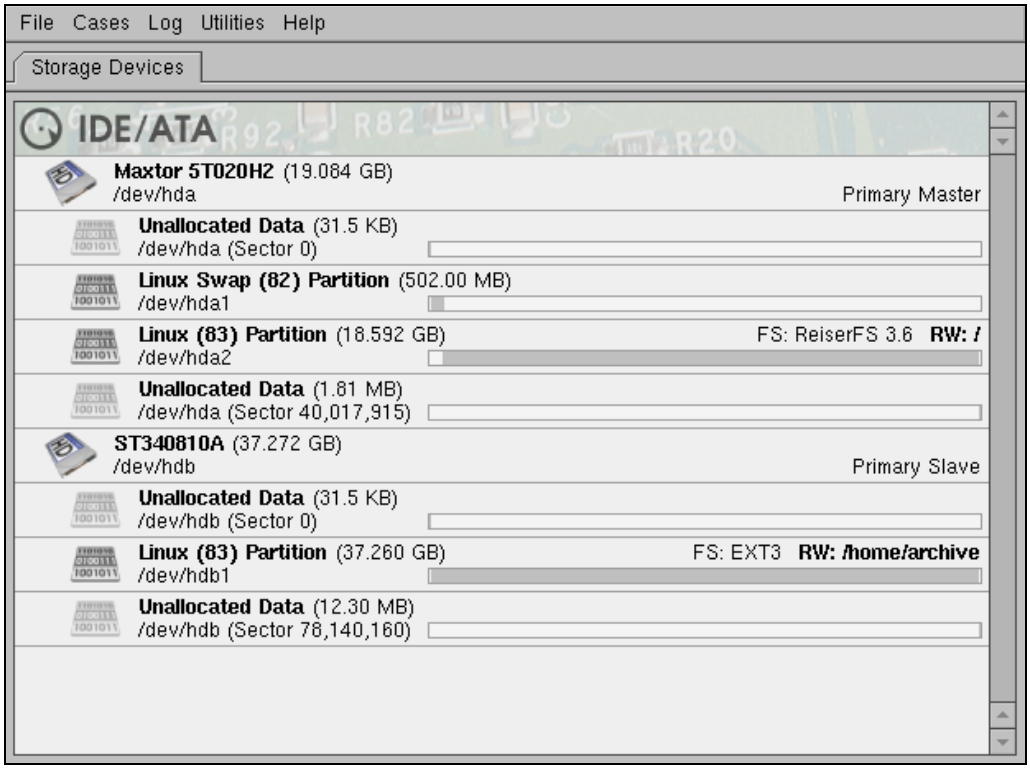


Figure 3.15: Screen-capture of SMART showing all the devices found.

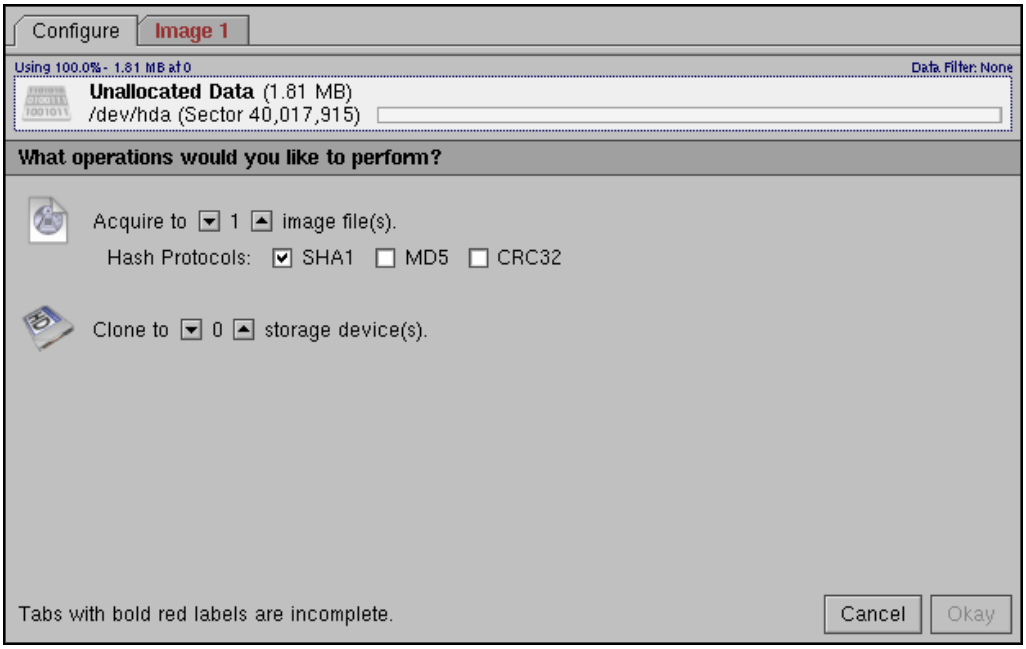


Figure 3.16: Screen-capture of SMART presenting the options for acquisition.

Report generation is performed by the Report Builder, where all logged and bookmarked events are shown and can be included in the reports. The Report Builder enables investigators to preview and modify reports prior to generation. It also saves reports in hypertext mark-up language (HTML) format. An example of the Report Builder interface, displaying the events which took place during the investigation, is shown in Figure 3.17.

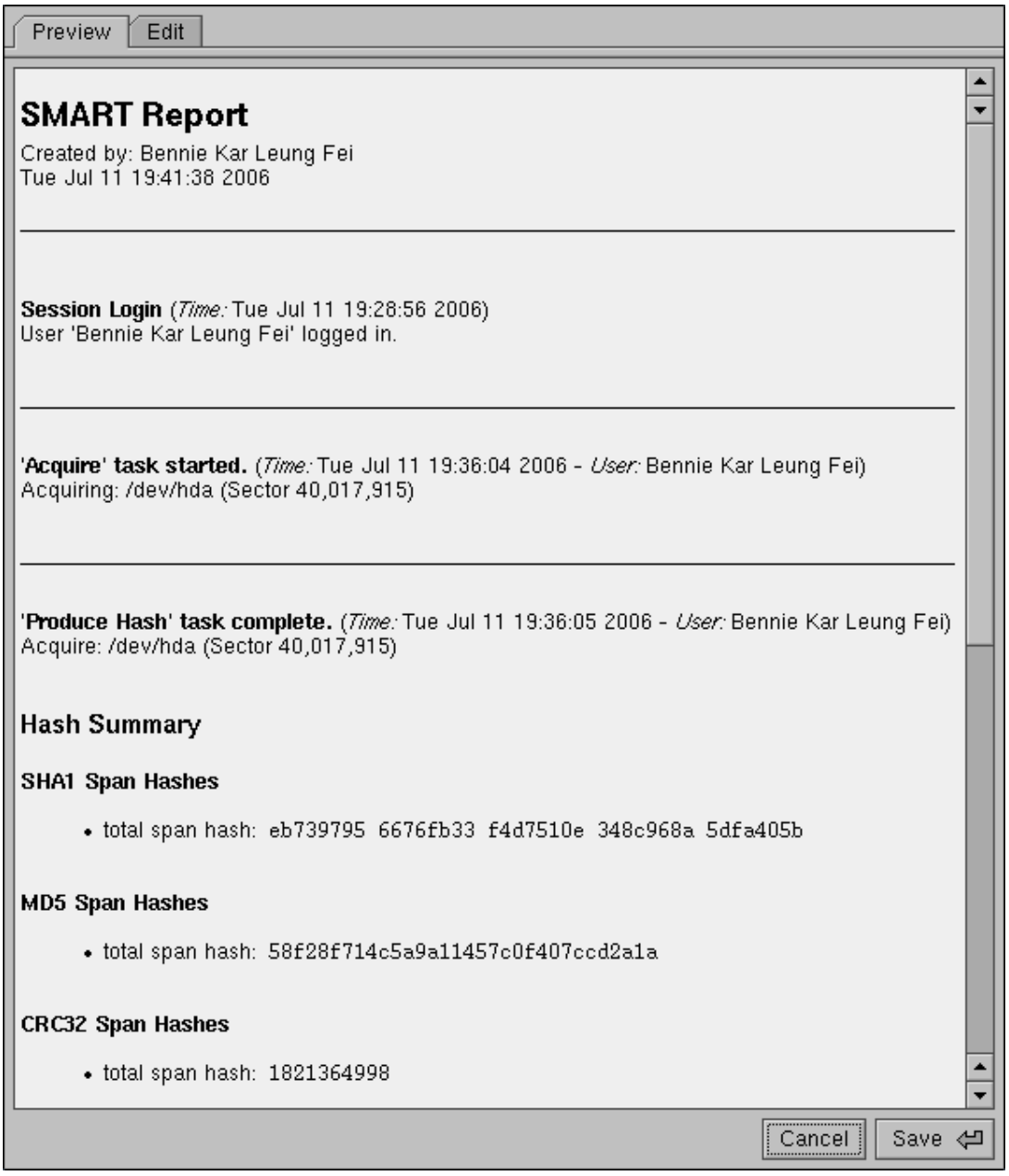


Figure 3.17: Report Builder interface.

3.1.5 Summary on computer forensic tools

The focus of this subsection is to provide the reader with a summary of the mentioned tools. Based on the above study, the key capabilities of the mentioned tools have been identified. As stated earlier, computer forensic tools have advanced from using command-line environments to providing sophisticated graphical user interfaces that significantly enhance investigative activities. This is clearly revealed by comparing SafeBack to the other featured tools.

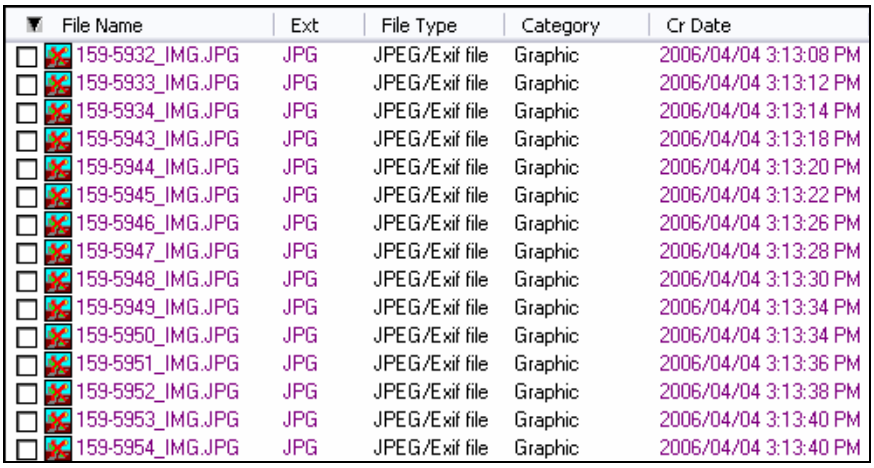
The first capability noticed is the ability to image media. This is a common capability that exists in all the mentioned tools. The primary focus of computer forensic tools is to ensure the accuracy of results and maintain the integrity of digital evidence. On that note, the NIST has provided some requirements for computer forensic tools when performing imaging (Lyle, 2003). These requirements state that when performing imaging, the tools should produce a bit stream copy of a piece of media without altering it in any way. Furthermore, the tools should verify the integrity of the image file and log any errors that might have occurred during acquisition. During the investigation, it was found that the mentioned tools all met the requirements stated above.

The next capability is the ability to conduct analysis. Essentially, once the acquisition process has been completed, it is necessary to analyse the bit stream copy. The majority of the mentioned tools offer analysis capabilities. The only exception is SafeBack. Such capabilities include the ability to perform hash analysis, file signature analysis, keyword searches and many more. It should be noted that a useful feature when conducting analysis is the presentation of files in a spreadsheet-style format. This ability permits investigators to view all the files found on a particular media as well as information regarding each file (see Figure 3.18, for example). The details include file name, file creation date and time, logical size and so on.

The ability to preview media before acquisition without altering it in any way is another common feature that exists in the majority of the mentioned tools. These tools offer previewing capabilities that can assist in locating specific data. Such previewing capabilities include viewing the content of a file. In general, when a file is selected, the content of the file is displayed by means of a viewer (see Figure 3.19 for example).

Reporting every event of the investigation, including the findings, is crucial and often concludes the outcome of the prosecution. Most tools have the ability to build reports that can include information regarding acquisition, bookmarked files and graphical images, and so on.

The knowledge gained through the overview of the more commonly deployed and well-known computer forensic tools in this section will be used to classify the tools and create a basic catalogue in the next section. This basic catalogue will then be used as the baseline from which improvements to computer forensic tools will be proposed in Section 3.3.



File Name	Ext	File Type	Category	Cr Date
159-5932_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:08 PM
159-5933_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:12 PM
159-5934_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:14 PM
159-5943_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:18 PM
159-5944_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:20 PM
159-5945_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:22 PM
159-5946_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:26 PM
159-5947_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:28 PM
159-5948_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:30 PM
159-5949_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:34 PM
159-5950_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:34 PM
159-5951_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:36 PM
159-5952_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:38 PM
159-5953_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:40 PM
159-5954_IMG.JPG	JPG	JPEG/Exif file	Graphic	2006/04/04 3:13:40 PM

Figure 3.18: Screen-capture showing the presentation of files.

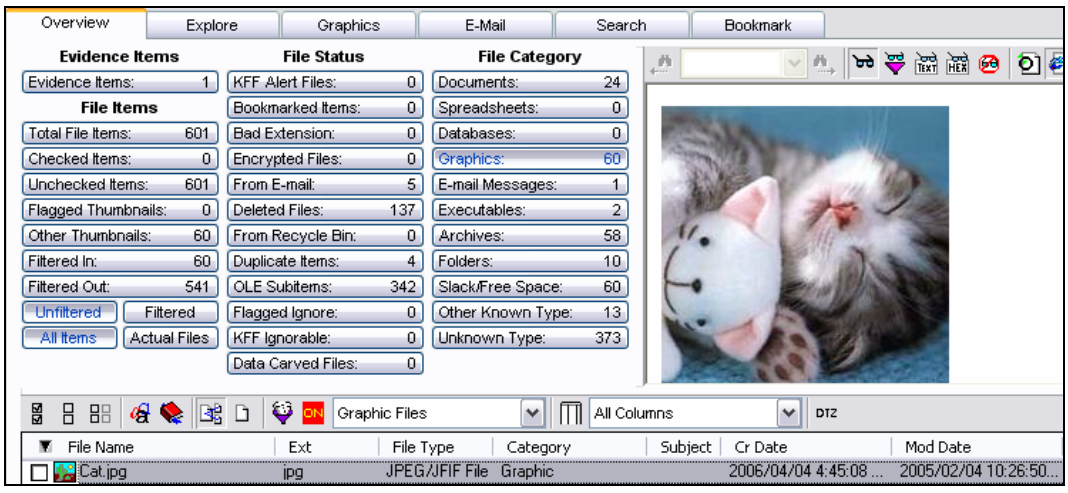


Figure 3.19: Screen-capture showing when a file is selected.

3.2 Classification of computer forensic tools

The study of computer forensic tools in the previous section has enabled the creation of a basic classification, which is presented in this section. Several categories of features have been identified and are referred to as the capabilities of computer forensic tools. In this section, the classification is presented and briefly discussed. A table is provided which lists some of the computer forensic tools and their capabilities. Note that this table lists tools other than those already mentioned with the purpose of confirming the capabilities most computer forensic tools are offering.

The classification shown in Figure 3.20 is based primarily on two characteristics:

- The operating environment, namely, Windows, DOS and UNIX;
- The capabilities of the computer forensic tool, namely, imaging, analysis, viewing, and reporting.

The aim of this classification is to also provide an overview of the current capabilities of computer forensic tools. It can be of assistance when selecting the right tool and also to motivate research into new computer forensic tools. For the purposes of this study, the classification is taken as the baseline from which improvements are proposed.

A brief overview of the different capabilities of computer forensic tools is provided in the following subsections.

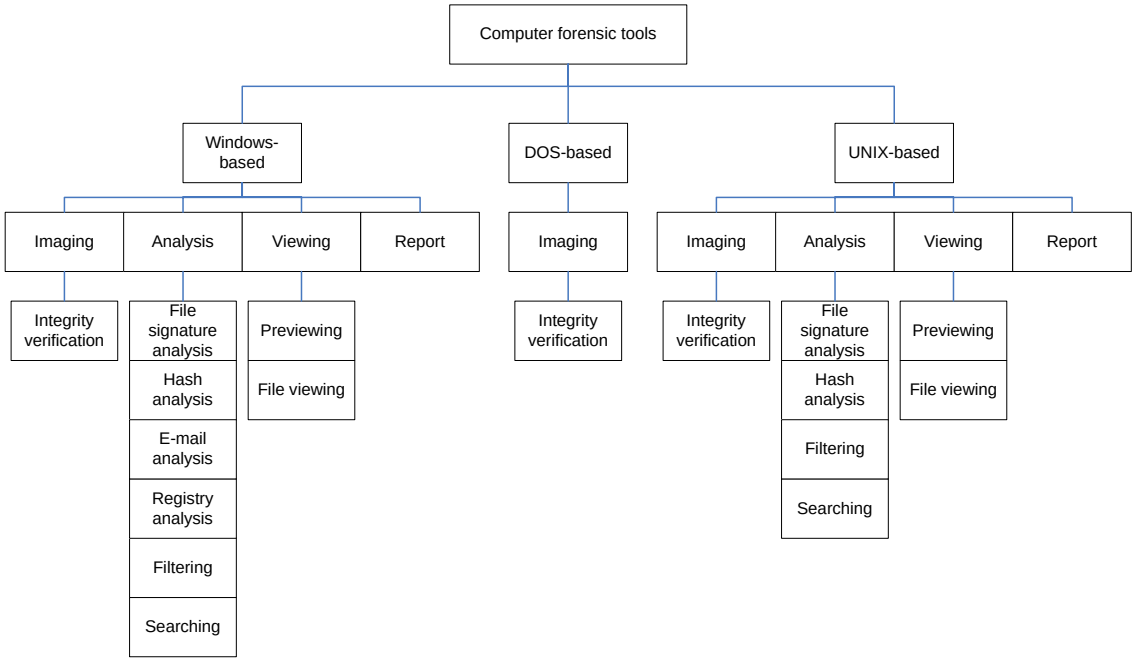


Figure 3.20: Classification of computer forensic tools.

3.2.1 Imaging

Imaging is the process of making an image. To be more precise, it is the process of copying data sector-by-sector from a piece of media to create a bit stream copy, otherwise known as an image of the media (Kenneally and Brown, 2005).

To acquire an image, specialist software is used to read a piece of media from its very beginning to its very end and create an image file that contains all the data in exactly the same order as it was read (Sammes and Jenkinson, 2000). This image file has to be the exact duplicate of the media, including both active and residual data. Once the source of evidence has been acquired, the digital investigator is given the outcome of the acquisition process which includes both the MD5 and SHA-1 hash values for verification.

Imaging can be a lengthy process, depending on the size of the media. Once completed, the image file can then be examined for items of interest. For example, e-mail messages sent or received on a suspect's computer system are subject to discovery and examination by law enforcement officials in criminal investigations.

3.2.2 Analysis

Once the data has been successfully gathered, it must be analysed to extract any relevant evidence. As stated previously, analysis is the process of interpreting data and placing it in a logical and useful format. It is also the process of determining the importance of the data and drawing conclusions from it (Carrier and Spafford, 2003; Reith et al., 2002). Part of the analysis stage may involve recovering residual data. This relies on specialist software that has the ability to recover and analyse both active and residual data.

Analysis of digital evidence involves a mixture of techniques. One common technique is to perform keyword searches on digital evidence. When performing a keyword search, the word list should be kept as short as possible. Furthermore, common words should be avoided. Keyword searches are generally used to locate occurrences of words or strings of text in data stored in files or slack and unallocated space (Schweitzer, 2003).

Another common technique is to perform file signature analysis. Changing the extension of a file can obfuscate its content. File signature analysis involves looking at the unique hex header signature of a file and checking whether it matches with the signature that is associated with its file type. Other techniques include hash analysis, registry analysis, e-mail analysis and filtering.

3.2.3 Viewing

The ability to view a piece of media or an image is invaluable in any digital investigation. For instance, it is often necessary to locate graphical images and determine their contents (Middleton, 2001); especially since child pornography is a typical digital crime.

Since there is no software that can be used to search and classify graphical images, digital investigators are required to extract all graphical images from the source media and manually search through them (Wang et al., 2005b). This can be a time consuming process, particularly if there are thousands of graphical images. However, as has been shown earlier in this section, the viewing capability of most computer forensic tools can help greatly reduce the human processing time required during this part of the investigation.

In general terms, viewing is the process of observing data on a piece of media. This process can simply be taking a preview of a piece of media before creating an image or viewing the contents of an image file or a piece of media. The viewing process also includes viewing the contents of any type file once it has been selected.

3.2.4 Reporting

Reporting is the process of capturing the findings of an investigation. Final reports should contain critical details from each step of the investigation, including reference to procedures followed and methods used to seize, document, collect, preserve, recover, reconstruct, organise, and search key evidence (Casey, 2004a).

The significance of the reporting process for a digital forensic investigation is often underestimated. Clues, analysis, and search results are sometimes mismanaged or neglected, particularly when relying upon a separate word-processing program that the digital investigator must cut and paste bits and pieces of numerous output files into or supplementing a report with further information entered manually at a later time (Casey, 2002).

Relevant evidence, comments, recovered pictures, search criteria, search results, and the date and time of the search process should be included in the report. This report will be presented in legal proceedings and often determines the outcome of the prosecution.

The following table (Table 3.1) lists some of the computer forensic tools and their capabilities. A tick mark appears only when a particular capability is available in the tool.

Table 3.1: Capabilities of computer forensic tools.

	Imaging	Analysis	Viewing	Reporting
Windows-based				
EnCase Enterprise	✓	✓	✓	✓
EnCase Forensic	✓	✓	✓	✓
FTK	✓	✓	✓	✓
FTK Imager	✓		✓	
ILook Investigator	✓	✓	✓	✓
Paraben Forensic Replicator	✓		✓	
ProDiscover Forensics	✓	✓	✓	✓
Ultimate Toolkit	✓	✓	✓	✓
X-Ways Forensics	✓	✓	✓	✓
DOS-based				
Byte Back	✓			
SafeBack	✓			
UNIX-based				
Autopsy Forensic Browser		✓	✓	✓
dd command	✓			
SMART	✓	✓	✓	✓

3.3 Limitations and recommendations

This section aims at extending the classification presented above by discussing some of the limitations of contemporary computer forensic tools. This discussion will then be used to proposed recommendations that could be made to the tools to enhance the task of performing digital investigations.

As demonstrated previously, a fully featured computer forensic tool offers capabilities such as imaging, analysis, viewing and reporting. Such capabilities are critical to conducting digital investigations. However, digital investigations

are becoming more time-consuming and complex as the volumes of data involved increase (Davis et al., 2005; Stephenson, 2003).

This is also partly because digital investigators are finding it increasingly difficult to use current tools to locate vital evidence within the massive volumes of data (Slay and Jorgensen, 2005). For example, one useful feature of computer forensic tools is the presentation of files in a spreadsheet-style format, but the process of scrolling through many rows of data can be extremely tedious when working with large data sets. Also, it can be difficult to locate specific information of interest to the investigation.

As observed, computer forensic tools are unable to present a visual overview of all the data found on a piece of media. Having this overview of the entire data set can be crucial since it reveals the overall pattern of the data set. This limitation makes it difficult for the investigator to locate the points of interest which could guide them to the next step of their search.

Storage media are steadily growing in size and analysis of a single computer system has become cumbersome in itself. This has made the process of analysing or investigating a large number of computer systems very difficult or practically impossible. However, the important aspect remains the detection of suspicious behaviour, which can help investigators narrow their subsequent searches for evidence.

It is often impractical to perform a complete examination of all the devices encountered in an investigation (Slay and Jorgensen, 2005). Therefore, it is necessary to narrow the search space down into smaller, more easily managed areas. Narrowing the search space makes it more feasible to locate the specific data which could corroborate the suspicious behaviour.

Computer forensic tools focus primarily on digital evidence recovery, in other words, on recovering residual data from a piece of media. These tools usually have limited abilities to assist in the analysis of the recovered data. The presentation of data offered by computer forensic tools is deceptive at times. The reason is that the dimensionality, complexity and volume of data still exists because the computer forensic tools merely present it to investigators. The digital investigators still have to examine the presented data and draw conclusions.

At present, computer forensic tools are not ideal for the following tasks:

- Association: identifying correlations among data;
- Classification: discovering and sorting data into groups based on similarities of data;
- Clustering: finding and visually presenting groups of facts previously unknown or left unnoticed;
- Forecasting: discovering patterns and data that may lead to reasonable predictions.

The above limitations of computer forensic tools also can be related to the forensic analysis of logs in network forensics. Logs residing in, for example, routers, Web servers and Web proxies are often manually examined, which can be a time-consuming and error-prone process. Again, the dimensionality, complexity and the amount of data is often prohibitively large.

Reducing multi-dimensional data into a two-dimensional visualisation can reduce the complexity that exists within the data. Furthermore, analysis of the two-dimensional visualisation will reveal the underlying structure of the data, any interesting and unusual patterns and potential outliers to investigators. All of which will improve the efficiency and quality of the forensic analysis process.

Extending the classification of computer forensic tools shown in Figure 3.20 reveals that, in most cases, the data presented by the tools still requires digital investigators to conduct lengthy manual examinations of the presented data before they can draw reliable conclusions. It also suggests that enabling the tools to reduce the complexity of the data they present will help digital investigators to draw faster and better conclusions. One of the best ways of achieving this is to provide them with a new capability that enables them to produce enhanced visualisations of the forensic data.

The similarity between the analysis of forensic data and the analysis of data in several other fields is striking. One related field is known as data mining. Data mining has been successfully applied across many different fields, but is still fairly unexplored in digital forensics. The use of data mining techniques in digital forensics, to identify new scenarios of interest and frequent or infrequent patterns of behaviour, can offer many potential benefits (Mohay, 2005). Such benefits include improving the quality of decisions, reducing human processing time and

reducing monetary costs (Beebe and Clark, 2005) and are discussed in detail in the next chapter.

3.4 Conclusion

With the increasing number of computer forensic tools available on the market, it is important to be aware of the different features that exist within the domain. In this chapter, the study of computer forensic tools has enabled the creation of a basic classification. The aim of this classification was to provide an overview of the current capabilities of computer forensic tools. It is also taken as the baseline from which limitations and recommendations were identified.

The next chapter discusses the field of data mining and the specific data mining techniques in order to identify a potential solution to the limitations established in this chapter.

University of Pretoria etd – Fei, B K L (2007)

Chapter 4

Data mining

Data mining is about finding answers from data (Vesanto, 2000). It has produced good results when gaining insight from large volumes of data. Data mining is the synthesis of statistical modelling, database storage and artificial intelligence technologies (Mena, 2003). The purpose of data mining is to discover new knowledge from data where the dimensionality, complexity or the amount of data is prohibitively large for manual analysis.

Data mining is part of the interdisciplinary field of knowledge discovery in databases (Palmerini, 2004). Research on data mining began in the 1980s and grew rapidly in the 1990s (Piatetsky-Shapiro, 1999). Specific techniques that have been developed within disciplines such as artificial intelligence, machine learning and pattern recognition have been successfully employed in data mining (Han and Kamber, 2005; Witten and Frank, 2005).

Data mining has been successfully introduced in many different fields (Perner, 2006). An important application area for data mining techniques is the World Wide Web (Han and Kamber, 2005). Recently, data mining techniques have also being applied to the field of criminal forensics (Chen et al., 2004; Mena, 2003). Examples include detecting deceptive criminal identities (Chen et al., 2004), identifying groups of criminals who are engaging in various illegal activities (Chen et al., 2004) and many more (Mena, 2003; Oatley and Ewart,

2003; Xue and Brown, 2006). In addition, recent research has focused on applying data mining techniques to digital forensics.

The focus of this chapter is to briefly introduce the field of data mining and provide an overview of data mining functionalities. The overview will be followed by a brief discussion of how such techniques can be applied in the field of digital forensics. The discussion will pay particular attention to the field of Web mining because of the similarities in approach between it and digital forensics. The chapter will then conclude with a brief discussion of a potential method to overcome some of the problems mentioned previously in this study.

4.1 Data mining functionalities

Data mining techniques typically aim to produce insights from large volumes of data. Such tasks involve supporting the discovery of patterns. Data mining functionalities are used to specify the various types of patterns to be looked for. The following subsections discuss the data mining functionalities, namely, classification, clustering, association and prediction in more detail.

4.1.1 Classification

Classification is the process through which the common properties among data are found and classified into different classes (Chen et al., 1996). The field of classification has been studied substantially in data mining (Fayyad et al., 1996; Lu et al., 1996). Classification techniques are used for both descriptive and predictive data mining. Descriptive data mining characterises the general properties of the data, whereas predictive data mining performs inference on the data in order to make predictions.

The classification process involves analysing the data and developing a model (or classifier) for each class using the features available in the data. Classification is also called supervised learning (Han and Kamber, 2005). In supervised learning, the classes which an object belongs to are either known or specified in advance. Classification techniques include decision tree induction (Osei-Bryson, 2004), Bayesian classification (Fayyad et al., 1996), neural

networks (Engelbrecht, 2003), genetic algorithms (Davis, 1991). Some of these techniques are discussed in detail in the next section.

4.1.2 Clustering

Similar to classification, clustering (or cluster analysis) is the process of grouping data into meaningful clusters (or classes) in a way that maximises the similarity within clusters and minimises the similarity between two different clusters (Karypis et al., 1999). In some contexts, clustering is also called data segmentation because it partitions large data sets into groups according to their similarity (Han and Kamber, 2005).

In machine learning, clustering is often referred to as unsupervised learning since the classes which an object belongs to are not specified in advance (Fayyad et al., 1996). Essentially, it establishes the classes based on the similarities and dissimilarities present within the data. In addition, a cluster technique has the ability to reduce the amount of data and induce a categorisation. During the clustering process, it identifies clusters which can also be used for outlier detection. These outliers are the values that are distant from any cluster and can also be detected using statistical tests or distance measures.

Clustering utilises unsupervised learning through techniques such as partitioning methods, hierarchical methods and density-based methods. The two most important clustering methods are the partitioning and hierarchical ones. The concept behind partitioning methods is to divide the data directly into a number of clusters. That is, to classify the data into a given number of clusters. Hierarchical methods, on the other hand, locate clusters one at a time. This method merges the groups that are close to one another until all of the clusters are merged into one.

4.1.3 Association

Association is the attempt to identify relationships or correlations among data (Agrawal et al., 1993b). In general, association can be viewed as a two-step

process. It involves finding patterns that occur frequently in a data set. These patterns can be represented in the form of association rules.

Association rules have been extensively studied (Fayyad et al., 1996). It originated as a tool for discovering sets within a large collection of items in a supermarket (Agrawal et al., 1993a). It has evolved since then and has been extended in various ways to include multi-level association rules and multi-dimensional association rules (Han and Kamber, 2005). Multi-level association rules involve concepts at various levels of abstraction and multi-dimensional association rules involve more than one dimension. An example of a multi-level association rule is when levels of abstraction are referenced. In the case of software and antivirus, for example, software is a higher-level abstraction of antivirus. An example of a multi-dimensional association rule is when a rule references two or more dimensions, such as age and occupation.

4.1.4 Prediction

Prediction is the process of discovering patterns whereby an outcome (such as the potential sales of a new product at a given price) can be forecasted (Han and Kamber, 2005). The commonly used approach for numeric prediction is regression.

Regression analysis is used when data observations can be modelled, and therefore forecasted, by a mathematical function using a given set of data characteristics. The task of regression is to map data onto a function. Methods for regression analysis include linear regression and nonlinear regression.

The standard technique for numeric prediction is linear regression (Han and Kamber, 2005). Linear regression involves locating the optimal line between two attributes in order to predict the one through its relationship to the other. In linear regression, the data are modelled to fit a straight line.

4.2 Data mining applied in digital forensics

Data mining techniques are designed for large volumes of data. Hence, they are able to support digital investigations. While such techniques have been employed

in other fields, their application in digital forensics is still relatively unexplored. Some of the data mining techniques applied in digital forensics are as follows:

- **Association rules** have been employed to profile user behaviour and identify irregularities in log files (Abraham and de Vel, 2002). Such irregularities can assist in locating evidence that might be crucial to a digital investigation. In this instance, the rule sets generated from the log files were considered to describe a profile contained within the data. The log files contained user login information for a computer system. By generating the rule sets, behavioural profiles of users were developed and used to detect behavioural anomalies.
- **Outlier analysis** has been utilised to locate potential evidence in files and directories that have been hidden or that are different from their surrounding files and directories (Carrier and Spafford, 2005). Outlier analysis can assist in locating hidden files and directories that might have been concealed by an attacker. In order to locate hidden files, the characteristics of each file within a directory are compared to detect potential outliers. This approach is similar to that used when locating hidden directories where the characteristics of directories at the same level are compared.
- **Support vector machines** have been utilised in several research areas in the field of digital forensics. A support vector machine (SVM) is an algorithm for classification that seeks categorised data based on certain fundamental features of the data (Joachims, 2002). In one instance, a support vector machine was applied to determine the gender of the author of an e-mail based on the gender-preferential language used by the author (Corney et al., 2002). In another instance, a support vector machine was applied to determine the authorship of an e-mail (de Vel et al., 2001). Based on the content of the e-mail, each e-mail was classified according to its likely author.

  Image mining is one of the many activities undertaken during a digital investigation. A support vector machine can also be used to recognise certain patches or areas of an image (Brown et al., 2005). Consequently, it can be used to detect and filter out images that are

irrelevant or unrelated to the case at hand. Other instances where support vector machines have been utilised include image retrieval (Brown and Pham, 2005) and in executing search queries on images containing suspicious objects (Brown et al., 2003).

Support vector machines have also been applied in network forensics to detect intrusions (Mukkamala and Sung, 2002; Mukkamala and Sung, 2003). In this instance, support vector machines are used for feature ranking in support of intrusion detection.

- **Discriminant analysis** has been employed to determine whether contraband images, such as child pornography, were intentionally downloaded or downloaded without the consent of the user (Carney and Rogers, 2004). Often, individuals prosecuted for crimes based on digital evidence claim that a Trojan horse or virus installed on their computer system was responsible. In this instance, discriminant analysis provided a mechanism for event reconstruction and enabled digital investigators to counter the Trojan defence by examining the characteristics of the data.
- **Bayesian networks** have been used to automate digital investigations. Bayesian networks are based on Bayes' theorem of posterior probability (Han and Kamber, 2005). A Bayesian network is a directed acyclic graph which models probabilistic relationships among a set of random variables (Pernkopf, 2005). In one instance, a Bayesian network has been used to model and reason about attacks on computer systems and networks (Duval et al., 2005). The aim was to gather information about likely attacks, actions performed by attackers, the most vulnerable software systems and the investigation techniques that should be used.

The data mining techniques mentioned above are only some of the data mining techniques applied in digital forensics. Other areas in digital forensics where data mining techniques are utilised include the use of wavelet transforms to analyse data in network security databases (Liu et al., 2003), wavelet transforms to analyse images in steganography detection (Farid and Lyu, 2003), attribute-oriented induction to identify irregularities in log files (Abraham et al., 2002) and possibly many more as researchers continue to explore the field.

In the next section, how Web mining is performed and how it can be related to network forensics is discussed.

4.3 Web mining

Data mining on the Internet is commonly referred as Web mining (Zaiane, 1999). A great deal of research has been conducted in the field of Web mining (Eirnaki and Vazirgiannis, 2003; Kolari and Joshi, 2004; Kosala and Blockeel, 2000; Li et al., 2002; Mobasher et al., 1996). It is the extraction of interesting and useful knowledge, as well as implicit information, from activities related to the World Wide Web (Abraham and Ramos, 2003). The rise of Internet-related crimes has created an overlap of interest between Web mining and network forensics. Web mining is categorised into three main areas: Web content mining; Web structure mining, and Web usage mining (Li et al., 2002).

Considerable work has been done in the area of Web usage mining (Abraham and Ramos, 2003; Berendt, 2000; Cooley et al., 1999; Gery and Haddad, 2003; Srivastava et al., 2000). In general, Web usage mining involves three phases: data pre-processing, pattern discovery and pattern analysis. Web usage mining seeks to reveal knowledge hidden in Web server log files, including statistical information about site visitors, and the preferences, characteristics and navigational behaviour of computer users.

The following subsections discuss the three main phases: data pre-processing, pattern discovery and pattern analysis. It should be noted that these phases are applied at a later stage.

4.3.1 Data pre-processing

Data pre-processing is mainly concerned with data cleaning, data transformation and data reduction. The goal of data cleaning is to remove irrelevant information. Data transformation converts the raw data into structured information. Lastly, data reduction reduces the representation of the data set into a smaller volume to make analysis more practical and feasible. An example of a strategy for data reduction is dimensionality reduction.

In general, data pre-processing can help reduce a search space into smaller, more easily managed parts. This can save valuable time during an investigation.

4.3.2 Pattern discovery

Once data pre-processing has been completed, the next step is to apply intelligent methods in order to extract patterns from the data. This process is commonly referred to as pattern discovery.

Pattern discovery involves the use of the data mining functionalities such as classification and clustering (Han and Kamber, 2005). Pattern discovery draws on algorithms used in data mining, machine learning and pattern recognition to detect interesting patterns. These patterns can be further analysed during the pattern analysis phase to gain better insights into the data.

The pattern discovery technique favoured in this study is the self-organising map (SOM). The SOM is a neural network model that has attracted a great deal of interest among researchers in a wide variety of fields (Brittle and Boldyreff, 2003; Deboeck, 1998; Payer et al., 2005; Tangsripairoj and Samadzadeh, 2004), but has yet to be applied to digital forensics.

In this study, the SOM is proposed for clustering, visualising and analysing forensics data. The characteristics of the SOM make it ideal for association, classification, clustering and forecasting. The SOM has the ability to map high-dimensional data onto a two-dimensional space. This allows for a better understanding of multi-dimensional data, as well as reducing the complexity that exists when performing a forensic analysis. Using the SOM, forensic data can be presented in a visual and graphical manner. This offers digital investigators a fresh perspective from which to study the data.

4.3.3 Pattern analysis

Expressing discovered knowledge in visual representations makes it easier to interpret. Nonetheless, the patterns discovered still need to be analysed in order to determine which ones are interesting and relevant. The purpose of the pattern analysis phase is to identify pertinent patterns. Such identification can be

supported with visualisation techniques. The visualisation of results obtained from the pattern discovery phase is simply the presentation of the results in visual forms, but can help to identify complex relationships within multi-dimensional data.

4.4 Conclusion

Data mining is the process of discovering new knowledge from data where the dimensionality, complexity or volume of data is prohibitively large for manual analysis. Data mining techniques can support digital investigations in various ways as digital investigations are increasingly experiencing large volumes of data.

This survey of data mining techniques that have been employed in the digital forensics arena, together with a general understanding of the data mining functionalities themselves, has suggested using the SOM for pattern discovery. The next chapter investigates this suggestion further by examining the SOM in detail.

University of Pretoria etd – Fei, B K L (2007)

Chapter 5

Pattern discovery: the self-organising map

Pattern discovery draws on algorithms used in, for example, data mining or machine learning, to detect interesting patterns. It is also useful for discovering new knowledge from data where the dimensionality and the volume of data are prohibitively large for manual analysis. To achieve this it uses data mining functionalities such as classification, clustering, association and prediction. Based on these characteristics, the self-organising map (SOM) (Kohonen, 1990; Kohonen, 2001) is proposed as an appropriate vehicle for pattern discovery.

The SOM is one of the most widely used neural network models. Since its introduction in the early 80s (Kohonen, 1981; Kohonen, 1982), it has attracted a great deal of interest among researchers in a wide variety of fields. At present, there are more than 5,000 research articles published on the SOM (Kaski et al., 1998; Oja et al., 2002). Data mining, in particular, is one of the areas where SOMs have been successfully employed (Deboeck, 1998; Kohonen et al., 2000; Smith and Ng, 2003; Yang et al., 2003). In general, the SOM is used to map high-dimensional data onto a low-dimensional space, typically two-dimensional, while preserving the topology of the input data. The preservation of the topology means that similar data in the input space are placed nearby on the map. The SOM has been extensively used in the clustering and visualisation of high-dimensional data. Clustering attempts to group data with similar characteristics.

Visualisation is the process of mapping complex data to a graphical representation to provide qualitative notions of its properties.

In the next section, the architecture of the SOM is discussed in detail. Subsequent to that, the learning process of the SOM is described in Section 5.2. Section 5.3 focuses on data visualisation and how the various visualisations of the SOM make it ideal for supporting forensic analysis. Finally, the chapter concludes in Section 5.4.

5.1 Architecture

The architecture of the SOM typically consists of two layers of units, namely, the input layer and the output layer (see Figure 5.1). These units are often referred to as neurons. Each unit in the input layer, which represents an input signal, is fully connected with the units in the output layer. The output layer generally forms a two-dimensional grid of units, where each unit represents a unit of the final structure. The connections between the layers are represented by weights. Therefore, each unit in the output layer is represented by a weight vector which is also referred to as a prototype vector. Essentially, each weight vector has the same dimension as the input data. The adjustments of the weights are done through the learning process of the SOM, which is described in the next section.

Often, the accuracy of the map depends on the number of units in the output layer. For example, too few units may result in a poor performance of the SOM. Too many units, however, may lead to an overfitted map and the SOM losing the ability to generalise, which leads to a poor performance on unseen input patterns and an increase in computational complexity. Therefore, in general, the number of units should be less than or equal to the number of input patterns (Engelbrecht, 2003).

5.2 Learning process

The SOM is based on unsupervised competitive learning, which means the learning process is entirely data driven and that the units in the output layer compete among one another. With unsupervised learning, it is not necessary to

know about the characteristics of the input data since there is no target involved and it does not require any human supervision.

The SOM can be viewed as a constrained version of k-means (Alsabti et al., 1998), which is a very popular clustering algorithm. The effect of the learning process is to cluster together similar patterns. Prior to the learning process, weights are initialised. These weights are then adjusted throughout the learning process. The proper initialisation of weights enables a fast convergence that leads to better results.

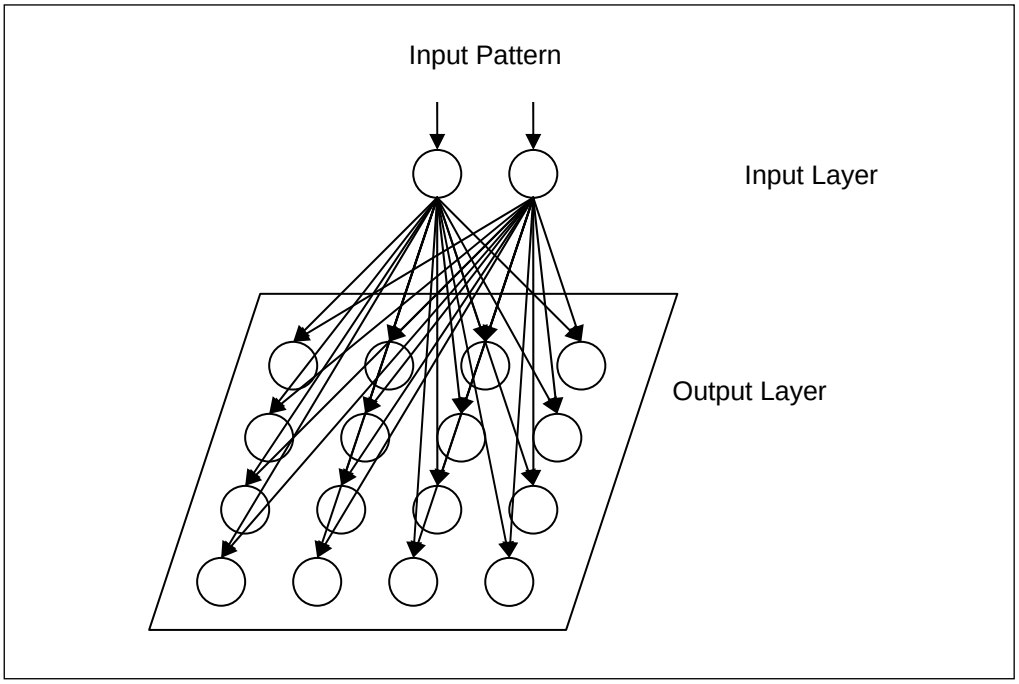


Figure 5.1: The architecture of the self-organising map.

There are three methods of weights initialisation: initialisation with random values; initialisation with patterns, and linear initialisation (Vesanto, 1997). Initialisation with random values is where random values, which are completely independent from the input data, are used to initialise the weights. Initialisation with patterns is, basically, where random input patterns are used for initialisation. Linear initialisation, on the other hand, is where weights are initialised in an ordered manner on the two-dimensional subspace spanned by the two principal components (or eigenvectors) of the input data (Kohonen, 2001). Theoretically,

in terms of digital forensics, very often digital investigators know very little or nothing at all about the forensic data, therefore, initialisation with random values can be a good approach for initialisation.

In addition to the initialisation of the weights, it should be noted that the performance of the SOM is also influenced by the choice of parameters such as the number of learning iterations, the initial value of the learning rate and the dimensions of the map.

Theoretically, the learning process involves two major steps: identifying the winning unit and updating unit weights. The learning is stochastic, where unit weights are updated after each input pattern is presented. When an input pattern is presented to the input layer, the units in the output layer will compete with one another. The winning unit in the output layer will be the one whose weights are closest to the input pattern in terms of Euclidean distance (Engelbrecht, 2003). After the winning unit is determined, the weights of that unit and its neighbouring units are adjusted. The weights are adjusted according to a pre-defined learning rate, which is a decreasing function of time. An appropriate learning rate must be chosen, one which ensures a fast convergence without adversely affecting the performance of the SOM. As a result, the learning rate should be neither too large nor too small. The learning process continues until the SOM produces acceptable results or a pre-set limit is reached on the number of learning iterations. A general algorithm for the SOM is summarised in Figure 5.2.

Another variation to the general algorithm for the SOM is the batch algorithm (Kohonen, 2001). This is where unit weights are updated only after all the input patterns are presented. As a result of this, the batch algorithm is much faster than the general algorithm mentioned above.

The next section, Section 5.3, discusses the visualisation of the SOM.

```
Initialisation of weights
For each learning iteration
begin
        For each input pattern
        begin
                For each output unit
                    begin
                        Calculate the Euclidean distance
                    end
                    Find the winning unit
                    Update the weights of the winning unit and its neighbouring units
        end
end
```

Figure 5.2: A general algorithm for the SOM.

5.3 Data visualisation

In addition to clustering, one of the key abilities of the SOM is to visualise high-dimensional data. The SOM achieves simplicity by reducing dimensionality via projecting high-dimensional data onto a two-dimensional grid. Hence, the SOM is particularly useful for visualisation. As mentioned previously, visualisation is the graphical representation of data to provide a qualitative understanding of the data. Data visualisation is the visual interpretation of high-dimensional data, which is particularly appropriate for obtaining an overall view of a data set and locating important aspects within a data set. This is particular useful in digital forensics because the data encountered in digital investigations is often large in size, multi-dimensional and complex. As a result, obtaining an overall view of the data can aid digital investigators to obtain a better understanding of the data and locate important aspects, which may result in the recovery of appropriate digital evidence.

Once the learning process has been completed, the SOM can be used as a convenient visualisation platform since it offers a powerful framework for visualising and analysing large volumes of data. There are a number of techniques for visualising the SOM which make it ideal for supporting forensic analysis. The various visualisations are discussed in the following subsections.

5.3.1 Visualisation of data

The SOM can be viewed as an ordered map which provides an overall view of the input data. When an input pattern is mapped to a particular unit on the map, the input patterns that have been mapped onto nearby units will have similar characteristics. As a result, an ordered map can be used to investigate the relationships among the input patterns. In addition, it enables the data to be interpreted in a fast and easy manner and valuable information ascertained quickly (Kaski, 1997). Such benefits can speed up digital investigations and save valuable time.

The SOM is much more illustrative than, for example, statistical charts or tables. It can be also visualised as a histogram. A histogram, by definition, is a synopsis data structure that can be used to approximate data distribution (Han and Kamber, 2005). The SOM has the ability to reveal the frequency of the data samples on the map. There are various ways to visualise the SOM as a histogram. The simplest way is to give units, for instance, a dark colouring for a large frequency of data samples and a light colouring for a smaller frequency of data samples (as shown in Figure 5.3).

5.3.2 Visualisation of components

Besides obtaining an overview of the data, it is possible to focus on smaller details such as insights into each of the dimensions. One of the advantages of a SOM is its ability to manifest possible correlations between different dimensions of input data in component maps. A component map can be thought of as a sliced version of the SOM (Vesanto, 2000). Each component map displays the spread of values of a particular dimension. Correlations are revealed by comparing component maps with each other because there are generally significant dependencies between dimensions. By doing so, digital investigators can detect frequent and infrequent patterns of behaviour. Such detection can provide digital investigators with further knowledge and assist them in reaching a decision.

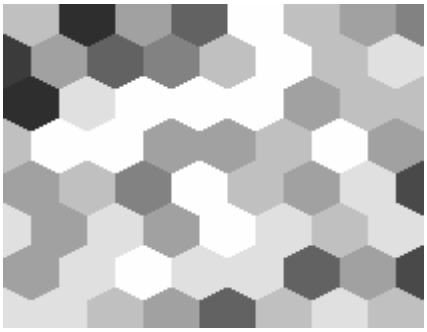


Figure 5.3: Representation of a data histogram.

In order to visualise the component maps, a potential technique is to colour code the map and ascribe similar colours to similar units. For example, the colour blue could be used to represent small values and the colour red to represent large values (see Figure 5.4). By doing so, the data is easily interpreted and recognised. Such colour coding can also facilitate the identification of possible clusters within the data and the linking together of different visualisations (Himberg, 1998).

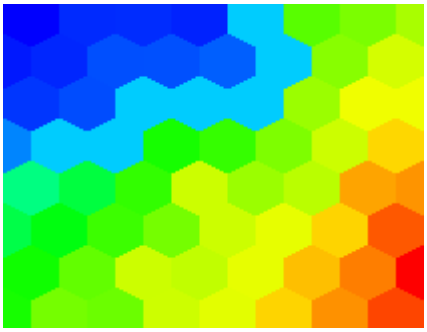


Figure 5.4: Representation of a component map.

5.3.3 Visualisation of clusters

In general, the effect of the learning process is to cluster similar patterns or cluster patterns with similar characteristics. Based on this, it is possible to locate groups of latent facts which offer digital investigators unobtrusive insights to the forensic data. There are various ways to visualise clusters on the SOM (Merkl and Rauber, 1997). In most cases, an additional step is required to determine the

cluster boundaries. This can be done by calculating the unified distance matrix (U-matrix) (Kohonen, 2001; Ultsh, 1993). The U-matrix is a representation of the SOM that visualises the distance between neighbouring units. The cluster boundaries are determined by calculating the distance between neighbouring units. Hence, large values within the U-matrix indicate the position of the cluster boundaries.

To visualise the U-matrix, each unit in the map is graded with a different shade. A light shading, for example, corresponding to a large distance and a dark shading corresponding to a small distance (see Figure 5.5). In this instance, dark areas will represent clusters and light areas the cluster boundaries. In Figure 5.5, there are two clusters separated by the units with light shading.

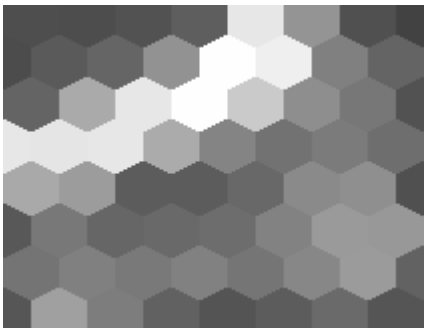


Figure 5.5: Representation of a U-matrix.

An additional method for locating cluster boundaries is Ward clustering (Drobics et al., 2000; Kohonen, 2001). This is where each unit initially forms its own cluster. Then, step-by-step, the clusters that are closest, according to a distance measure, are merged until optimal clusters have been constructed.

5.3.4 Visualisation of outliers

While clustering aims to group data with similar characteristics, these clusters can also be used for outlier detection. The outliers, in general, are the values that are distant from any cluster. Through the visualisation of the SOM, potential outliers can be easily detected by examining the clusters which are distant from one another. In addition, the use of colours can further facilitate the recognition

of potential outliers if the units that have been influenced by outliers are represented with similar colours. In terms of digital forensics, outlier detection is particularly useful when conducting a digital investigation because it can be used to detect anomalous behaviour. This allows investigators to narrow down their searches and concentrate on the areas where potentially anomalous behaviour has been detected.

5.4 Conclusion

The SOM can be used for both clustering and visualising high-dimensional data. Moreover, it is also ideal for association and classification. Association seeks to identify correlations in data. Classification maps data into predetermined classes. The learning process involves the clustering of data and produces an ordered map which can then be visualised. The SOM can be used as a convenient visualisation platform and can serve as a basis for further forensic analysis of data. The various potential visualisations of the SOM make it a powerful framework for analysing and visualising large volumes of data. As a result, its use can contribute to an increase in the quality of digital investigations by reducing the amount of effort required to analyse the often large quantities of data involved. Furthermore, its use can improve the quality of the forensic analysis being conducted because it can also perform interactive analysis. Therefore, it is not only about presenting the visualisation of the SOM, but also interacting with the SOM.

This advantage is demonstrated in detail in the next chapter, which describes a prototype implementation of the SOM.

University of Pretoria etd – Fei, B K L (2007)

Chapter 6

The SOM Forensic Analysis Tool

The SOM Forensic Analysis (SOMFA) Tool employs an unsupervised neural network based on the concept of the SOM as discussed previously in Chapter 5. It allows mapping of high-dimensional data onto a two-dimensional map. Furthermore, the SOMFA Tool has the ability to group data with similar characteristics and produce an ordered map which can then be visualised. Visualisation techniques are applied to the two-dimensional map and then displayed in the form of a hexagonal grid. The objective of the SOMFA Tool is not only about presenting an ordered map, but also about enabling an interactive analysis with the forensic data. This provides a more efficient way of analysing the forensic data and improves the quality of decisions in digital investigations.

The focus of this chapter is to acquaint the reader with the SOMFA Tool and its capabilities. It starts with a motivation for implementing the SOMFA Tool. Thereafter, the architecture of the SOMFA Tool is presented in Section 6.2. This is followed by the system requirements of the SOMFA Tool in Section 6.3. Following from that, the functional overview of the SOMFA Tool is described in Sections 6.4, 6.5 and 6.6 respectively. Finally, the chapter is concluded with some concluding remarks in Section 6.7.

6.1 Motivation

The primary motivation for the implementation of the SOMFA Tool is to demonstrate the application of data visualisation in digital forensics using the SOM. There are various implementations of the SOM available, for example, the SOM Toolbox for Matlab 5 (Vesanto et al., 2000). However, these either do not provide interactive capabilities or are prohibitively expensive. The ability to perform interactive analysis is central to the usefulness of the SOM from a digital forensics perspective and the distinctive ability provided by the SOMFA Tool.

Another motivation for the implementation of the SOMFA Tool is to enable features such as automated data cleaning, data transformation and data reduction to be incorporated into the tool itself. Such features can assist in improving the efficiency of digital investigations and narrowing down the search space. Other features such as simple statistics and documentation capabilities are also incorporated in the SOMFA Tool. Moreover, the possibility of future development is also allowed for. For example, the eventual inclusion of various features designed specifically to encompass all the major digital forensic processes.

6.2 Architecture

The architecture of the SOMFA Tool consists of three major components. Each component performs one of the three required phases, namely, data pre-processing, pattern discovery and pattern analysis, as discussed in Chapter 4. Figure 6.1 presents the architecture of the SOMFA Tool.

Once the source of the evidence has been acquired and a data file containing information regarding the files found created by a computer forensic tool, the data file can be processed using the SOMFA Tool. Evidence can also be gathered from sources such as Web server logs or Web proxy logs and processed directly by the SOMFA Tool. Nonetheless, data pre-processing will, typically, be performed first. Data pre-processing involves data cleaning, data transformation and data reduction.

Once data pre-processing has been completed, the next step will be pattern discovery. Pattern discovery aims at discovering new knowledge, in the form of patterns or anomalies, from the forensic data. It involves the use of the SOM to map forensic data onto a two-dimensional space, which can then be visualised and analysed during the pattern analysis phase.

The purpose of the pattern analysis phase is to identify potential correlations, associations and anomalies within the data. Such identification is supported with the various visualisations possible using the SOM. These visualisations provide digital investigators with a general understanding of the forensic data. Furthermore, they help identify the complex relationships that may exist within the forensic data.

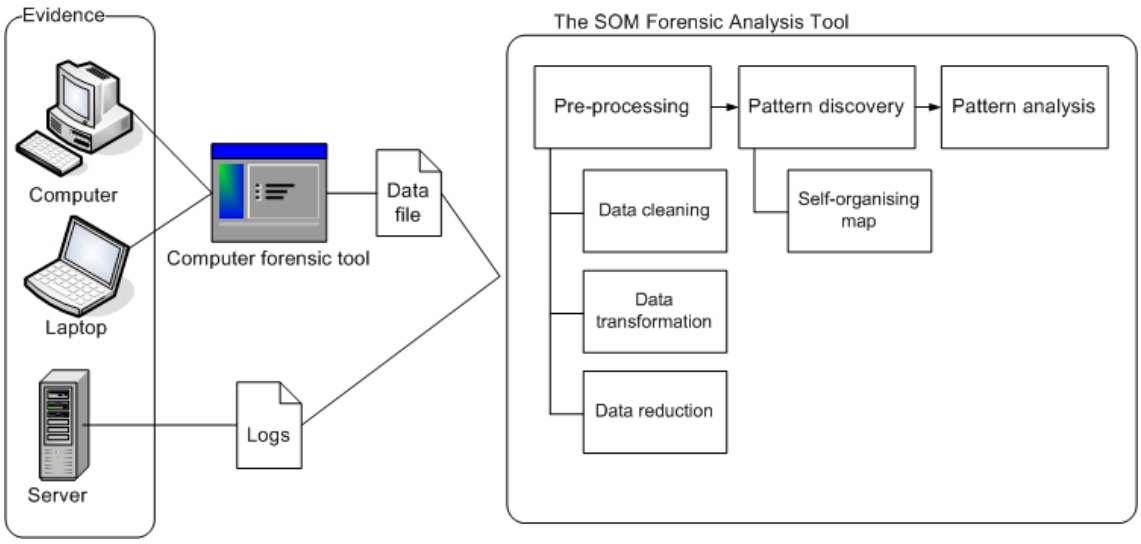


Figure 6.1: The architecture of the SOMFA Tool.

6.3 System requirements

The SOMFA Tool operates on a Windows platform (such as Windows 2000 or Windows XP) and was developed using Microsoft Visual C# .NET. As a result, one of the main requirements is that the Microsoft .NET Framework 1.1 must be installed onto the computer system before the SOMFA Tool can operate. Another requirement is to have sufficient memory. It is recommended to have at least 512

MB of memory and a fast processor, for example, a Pentium 4 processor, since the utilisation of the tool is computationally intensive.

To operate the SOMFA Tool, first install the Microsoft .NET Framework 1.1 by executing “dotnetfx.exe” followed by “SOMFA.exe”. It should be noted that these two files are available in the provided CD-ROM.

6.4 Data pre-processing

Data pre-processing is the process of processing raw data in order to improve the quality of data for further analysis. Data pre-processing involves data cleaning, data transformation and data reduction. The goal of data cleaning is to remove irrelevant information, while data transformation aims to transform the raw data items into structured information. The objective of data reduction is to then reduce the volume of the representation of the data set.

To start the data pre-processing process, click on **View** then **View Data File** in the menu. An interface, as shown in Figure 6.2, appears after clicking on the **View Data File** option. This interface allows the user to perform the data pre-processing tasks (such as data cleaning, data transformation and data reduction) on the specified data. To load the data file press **Browse**, then select the data file. Note that the data file can have any file extension as long as it is in the correct structure (see Section 6.4.1).

Once the data file has been loaded, the application determines the number of row entries and columns within the data file (see Figure 6.3). It should be noted that these columns can also be referred to as dimensions. Furthermore, the data in the data file is subsequently displayed to allow the viewing of the data before performing the data pre-processing tasks. The data displayed at the bottom of the data pre-processing interface allows the user to distinguish which of the data pre-processing tasks are required. In Figure 6.3, the setup of the data pre-processing tasks can be performed by selecting from the available options and entering the specific commands. To start the data pre-processing tasks, the user must first specify a name for the output file.

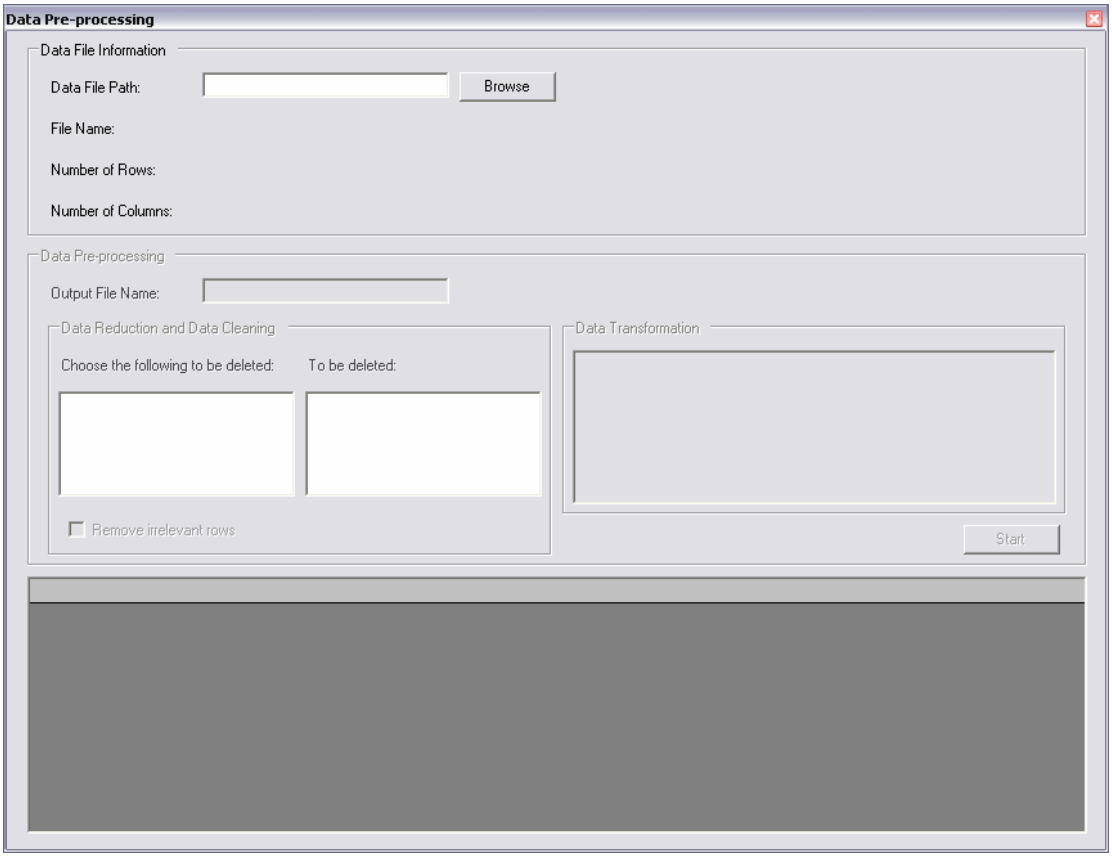


Figure 6.2: The SOMFA Tool – Data Pre-processing interface.

The setup of the data reduction process is performed by clicking on the items displayed in the list box under the **Choose the following to be deleted** label. This allows the user to specify which dimensions of the data should be deleted so that the volume of the data set can be reduced. When an item is clicked, the item will appear in the list box under the **To be deleted** label. Note that this list box gives an indication with regards to the deletion of the columns that are listed.

The data cleaning process will be performed when the check box next to the **Remove irrelevant rows** label is selected. This will remove any row entry that is either irrelevant or incomplete. A row entry will be classified as irrelevant if it contains attributes that will not contribute towards the learning process. For example, row entries containing attributes such as "-". Such attributes do not contribute towards the learning process because they contain no meaning. An incomplete row entry is, for example, a row entry containing missing values.

For data transformation, there are specific commands that must be entered. These commands specify the kind of operations to be performed on the various columns within the data set. It should be noted that each command statement must appear on a new line. In addition, the command statement must start with the name of the column, followed by "@", then the operation. For example, see Figure 6.3. The list of operations for data transformation is depicted in Table 6.1. An example of a command statement is as follows: "content@replace". This command statement specifies each data item in the column "content" to be replaced with a numerical value. For example, data items containing the value "text" are to be replaced with the value 1, "graphical image" are to be replaced with the value 2 and so on.

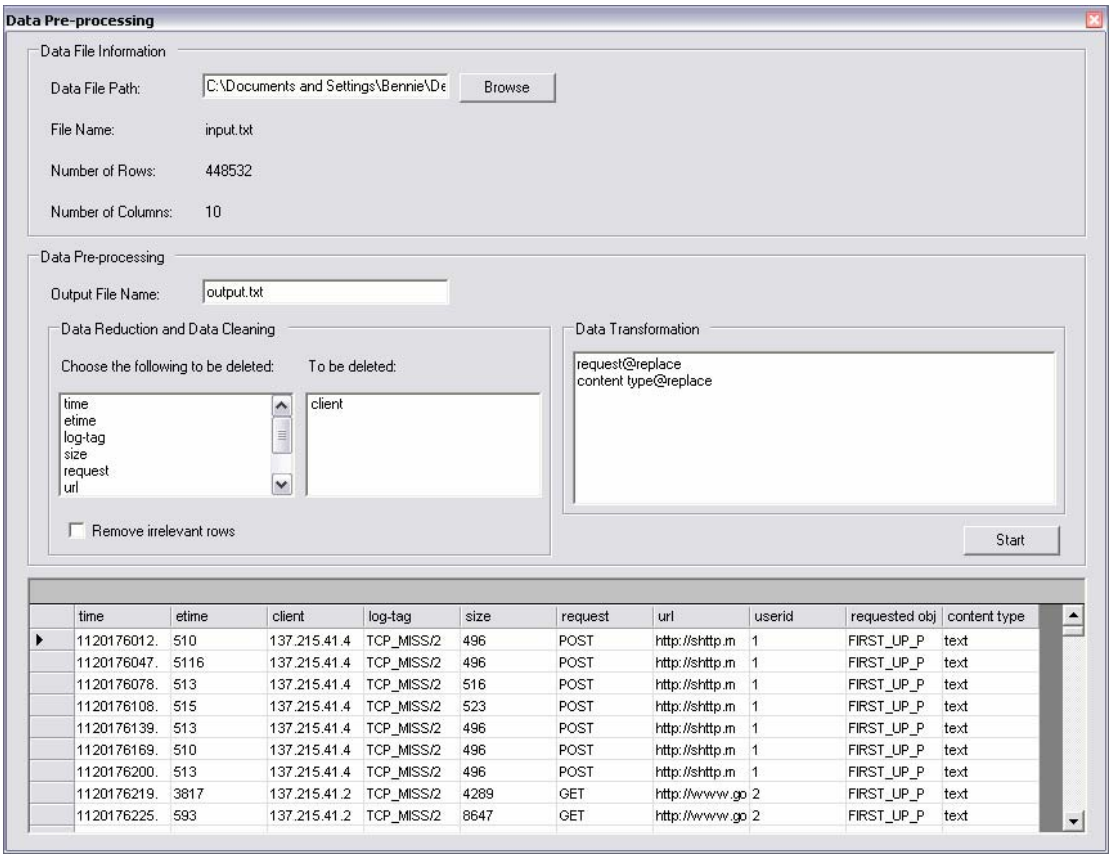


Figure 6.3: The Data Pre-processing interface after loading a data file.

It was mentioned earlier that documentation is an ongoing process throughout a digital investigation. Therefore, for documentation purposes, a

report containing the data pre-processing process, as well as additional information such as the name of the data file, the name of the output file and the date of the report, is generated after the data pre-processing process. This report helps investigators keep track of the changes made during the pre-processing process. The report is in the form of an HTML file and can be viewed by selecting **View** from the menu and clicking on **View Changes on Data File** in the sub-menu. An example of the report is shown in Figure 6.4.

Table 6.1: List of operations for data transformation.

Operation	Description
date	converts seconds since epoch into a readable format which contains three columns, namely, date, time and day and removes any forward slashes that appears in a date
replace	substitutes fields with numerical values
url	transforms the URL so that it is restricted to its domain name and subsequently substituted with numerical values
size	convert bytes to KB

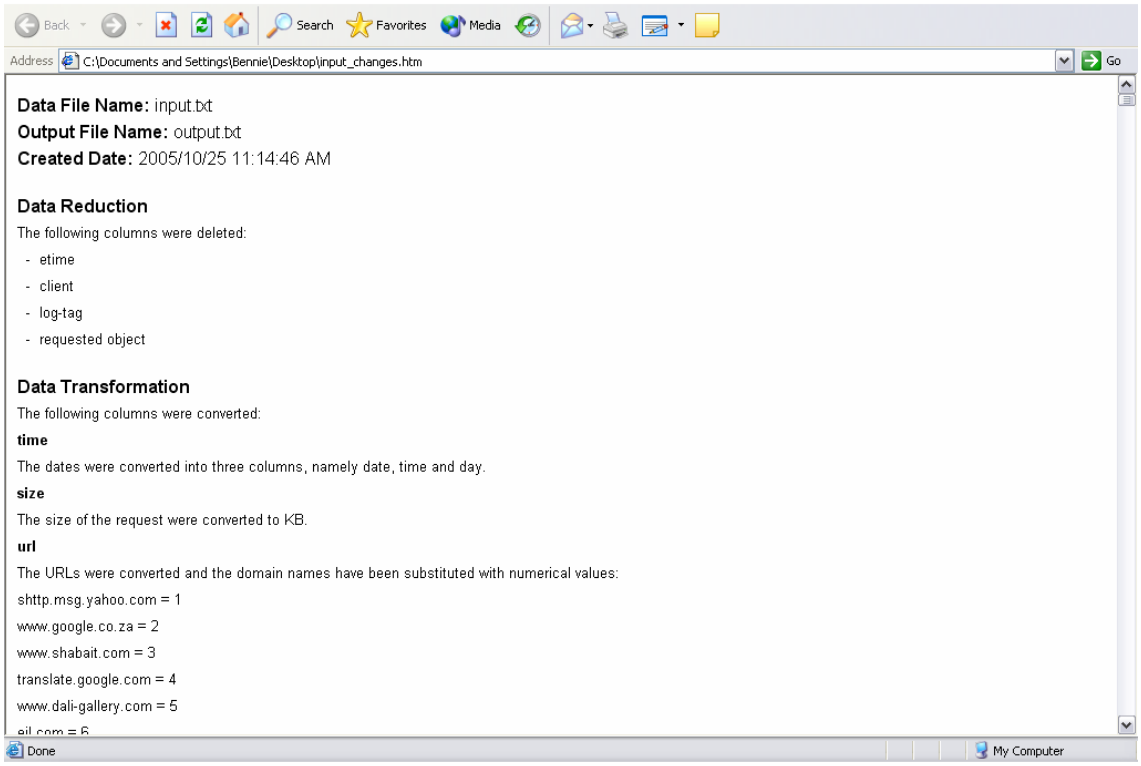


Figure 6.4: Example of a generated report after data pre-processing.

6.4.1 Data file structure

The structure of the data file has to be correct when using the SOMFA Tool. The data file can be obtained from numerous places, for example, it can be generated by a computer forensic tool, a Web server or Web proxy. The first line in the data file should contain the column headings or the names of the dimensions, thereafter, each data record in the data file must appear on a new line. It should be noted that a data record can be referred to as an input pattern. For each data record, the attributes must be separated by tab spaces. The attributes can be either a combination of letters or digits. However, it should be noted that the learning process of the SOMFA Tool requires the attributes to be numerical values. Therefore, the columns that contain strings should ideally be converted into numerical values through the data transformation process. An example of the structure of a data file is shown in Figure 6.5. In Figure 6.5, there are four dimensions. The first line contains the name of the dimensions, followed by the data records.

File type	File extension	Time created	Day created
Image	jpg	2332	5
Image	gif	2334	5
Document	pdf	2236	5

Figure 6.5: An example of the structure of a data file.

6.5 Pattern discovery

Pattern discovery draws upon methods and algorithms developed from various fields (such as data mining, machine learning, pattern recognition and many more) to detect interesting patterns (Han and Kamber, 2005). These patterns can be further analysed in the pattern analysis phase to gain unobtrusive insights of forensic data that will support digital evidence recovery. The SOM, discussed in Chapter 5, is the method utilised for pattern discovery.

Once data pre-processing has been completed, the pattern discovery process can begin. To start the pattern discovery process, select **File** then **New Case** from the menu. An interface, as shown in Figure 6.6, will appear. This interface allows the user to enter the information regarding the forensic analysis process and to select the data file for the pattern discovery process. Note that, as mentioned previously, the SOMFA Tool cannot process the strings existing within the data records. To be processed, the strings have to be converted to numerical values. This can be done by performing the data transformation as discussed in Section 6.4.

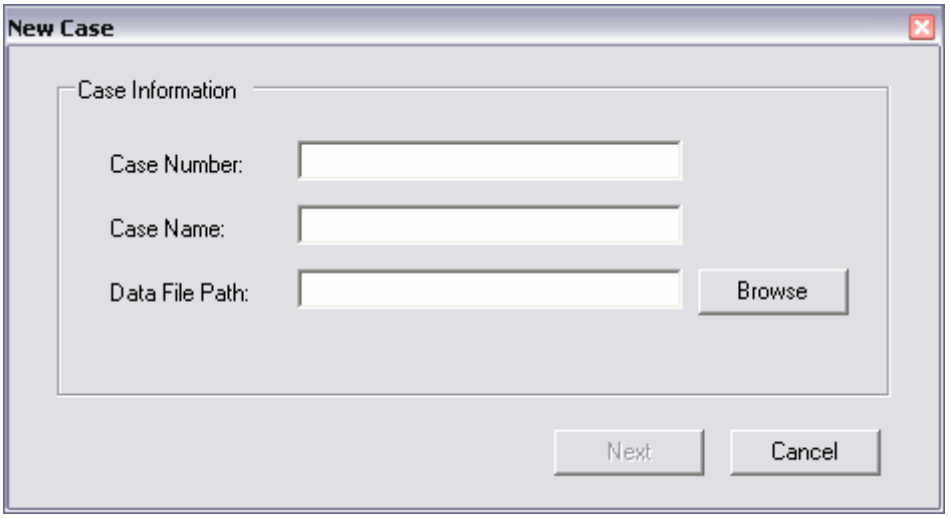


Figure 6.6: A screenshot of the new case interface.

The **Next** button, as shown in Figure 6.6, will be enabled once the data file has been selected. By clicking on the **Next** button, the next interface will appear (see Figure 6.7). This interface allows the user to select the dimensions of the map. In essence, the dimensions of the map (meaning the number of units in the output layer) specify the size of the SOM, which is often the size of a square. Furthermore, the interface shown in Figure 6.7 allows the user to choose between two learning methods. The preferred learning method can be selected from the list in the drop down box. Each learning method has its own advantages and disadvantages. The first learning method is the batch learning method, it allows for faster learning, but with lower accuracy. The second learning method is the stochastic learning method, which is slow compared to the batch learning method, but is more accurate. To proceed to the next and final interface prior to starting the learning process, click on the **Finish** button.

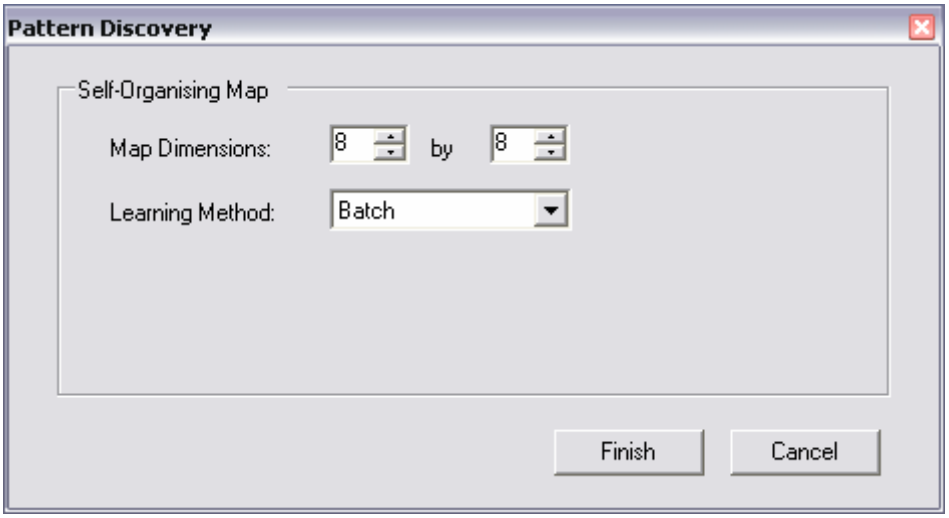


Figure 6.7: The SOMFA Tool – Pattern Discovery interface.

The **Learning** interface, as shown in Figure 6.8, allows the user to specify the various requirements for the learning process. It displays the dimensions in the data file, which enables the user to choose the dimensions to be included in the learning process. By clicking on a particular dimension, it will appear in the list box under the **To be included** label. In Figure 6.8, the text box next to the **Learning Iterations** label allows the user to specify the number of learning iterations. Note that the learning process will take some time to reach an accurate result. The learning process continues until the SOM produces acceptable results or the pre-set limit on the number of learning iterations is reached. The default amount of learning iterations is twice the number of input patterns. There is no specific rule for choosing the number of learning iterations. However, in general, the default number should allow the SOM to reach an acceptable result.

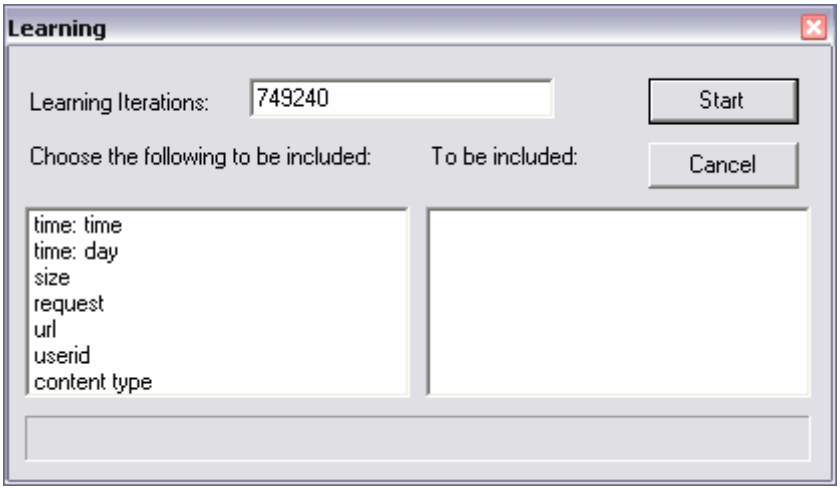


Figure 6.8: The SOMFA Tool – Learning interface.

6.6 Pattern analysis

From the pattern discovery process an ordered map is generated. This two-dimensional map, also known as the SOM, is displayed in the form of hexagonal grids. More importantly, it is used as a visualisation platform offering a powerful framework for visualising and analysing the data that was provided to the pattern discovery process. There are numerous techniques for visualising the SOM, which makes it ideal for supporting forensic analysis. The SOM has the ability to give digital investigators unobtrusive insights of the input data and assist them in digital evidence recovery.

The choice of visualisations of the SOM can be made using the **Display Options** interface (see Figure 6.9). This interface is displayed upon completion of the learning process. There are three types of visualisation which can be selected. To select the different visualisations, simply check the boxes next to them and click on the **OK** button. Note that the **Display Options** interface is also accessible through the menu by selecting **View** and clicking on **Display Options**.

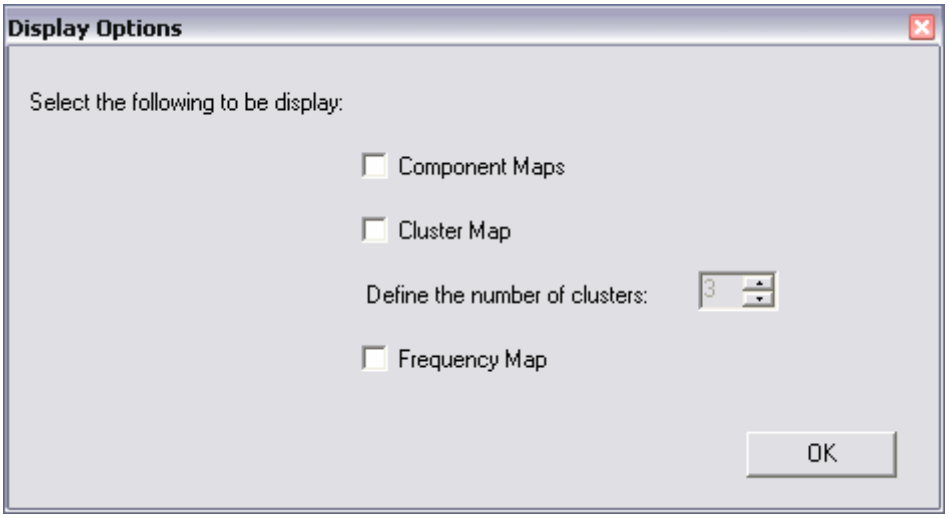


Figure 6.9: The SOMFA Tool – Display Options interface.

The first visualisation that can be selected is **Component Maps.** Each component map visualises the spread of values of a particular dimension. A component map can be thought of as a sliced version of the SOM that can be used to focus on the smaller details within each of the dimensions. For each component map, similar colours are ascribed to similar units. An example of a component map is given in Figure 6.10. The colour blue indicates small values, the red indicates large values, and the other colours (such as green and yellow) represent intermediate values. Furthermore, the solid black line indicates the cluster boundaries. Although it is not shown in Figure 6.10, it should be noted that any unit that there is no data or input patterns mapped onto will be depicted with a grey colour.

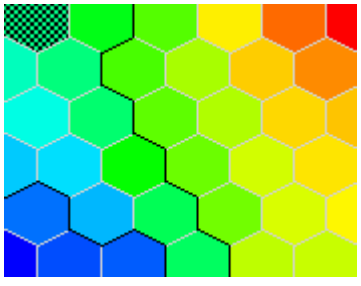


Figure 6.10: The component map.

The second selection is the **Cluster Map**. The effect of the learning process is to cluster similar patterns or cluster patterns with similar characteristics. The cluster map is used to reveal these clusters within the map. In the cluster map, each cluster has a distinct colour. In addition, each cluster is separated by a cluster boundary, which is indicated with a solid black line. Ward clustering is used to locate the cluster boundaries. The number of clusters can be specified by clicking the up and down buttons in Figure 6.9. This option gives the user the ability to view a desired number of clusters, which might be helpful during forensic analysis. An example of a cluster map is shown in Figure 6.11. The cluster map in Figure 6.11 reveals two clusters – one displayed in red and the other in cyan. The brightness of the colour reveals the distance between each unit and its 'centre of gravity', namely, the map unit that most closely represents the average of all the units within that particular cluster. In Figure 6.11, brighter colours indicate longer distances and darker colours indicate shorter distances.

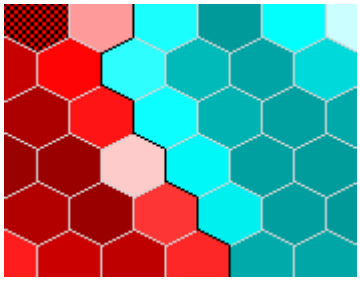


Figure 6.11: The cluster map.

The third and final selection is the **Frequency Map.** This reveals the frequency of input patterns mapped onto various units within the map. The frequency map can be classified as a data histogram to approximate data distribution. An example of a frequency map is shown in Figure 6.12. A darker red colour indicates a large frequency of input patterns mapped onto a unit and a lighter colour indicates a small frequency. The frequency map is used to support both determining the data distribution among the units in the map and to obtain an overall view of the data set.

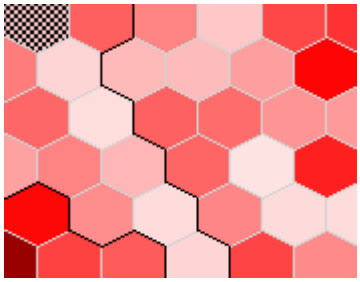
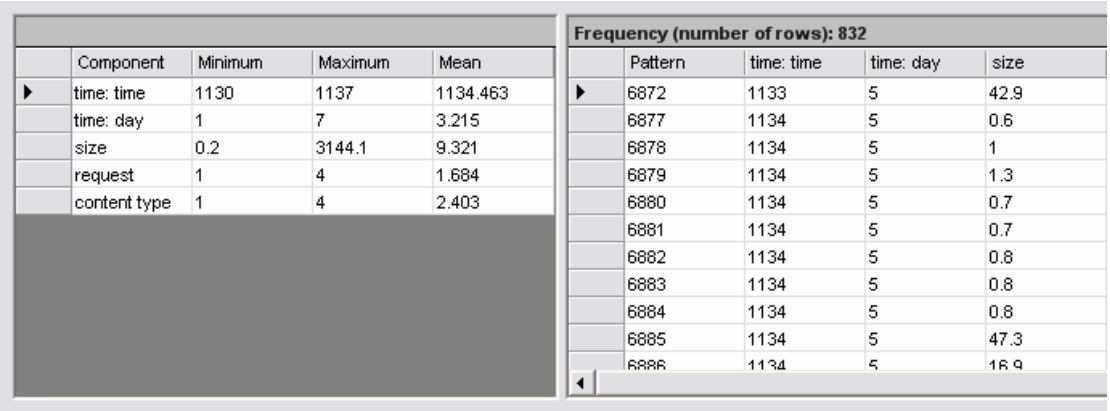


Figure 6.12: The frequency map.

Once the required visualisations have been selected, the two-dimensional maps are displayed and interactive analysis can begin. These maps provide an important visualisation aid since they give a complete picture of the data. Furthermore, the maps should be used in conjunction with each other because together they form the backbone of the forensic analysis. Moreover, smaller portions of the maps can be focussed in on to narrow down the investigation.

Each unit within the map contains information regarding the input patterns that have been mapped onto it. This information will appear when investigating a particular unit. In other words, when a unit is selected, the information will automatically appear. A selected unit is highlighted with a small checker board. When a unit is selected, the information regarding that particular unit is displayed in a dialog box, as shown in Figure 6.13. This dialog box consists of two grids. The grid on left displays general statistics and the grid on the right displays the actual data records mapped onto the unit. The statistics displayed by the grid on the left include the minimum value, the maximum value and the mean, for each of the components or dimensions. The actual data records displayed by the grid on the right include the total number of row entries mapped onto that particular unit. Both grids have the ability to perform sorting on each of the columns. This can be performed by clicking on the column headings.



Component	Minimum	Maximum	Mean
time: time	1130	1137	1134.463
time: day	1	7	3.215
size	0.2	3144.1	9.321
request	1	4	1.684
content type	1	4	2.403

Frequency (number of rows): 832

Pattern	time: time	time: day	size
6872	1133	5	42.9
6877	1134	5	0.6
6878	1134	5	1
6879	1134	5	1.3
6880	1134	5	0.7
6881	1134	5	0.7
6882	1134	5	0.8
6883	1134	5	0.8
6884	1134	5	0.8
6885	1134	5	47.3
6886	1134	5	16.9

Figure 6.13: The SOMFA Tool – Information interface.

An additional feature of the SOMFA Tool is that other statistics can also be displayed while analysing the two-dimensional maps. These statistics can be obtained by clicking **View** then **Statistics**. The **Statistics** interface displays two grids (see Figure 6.14). The grid on the left displays the frequency between the fields on a specific dimension. It gives the percentages for each field on the unit being investigated. For example, in Figure 6.14, the selected dimension is “File Type” and the grid on the left shows that for the map unit being investigated, there is an occurrence of 40.27% of value 1 and 59.73% of value 2.

The grid on the right, as shown in Figure 6.14, shows the frequency between the fields on a specific dimension for the entire map. For example, in Figure 6.14, the grid on the right shows that for the entire map, there is an occurrence of 49.92% of value 1 and 50.08% of value 2.

Finally, for documentation purposes, a report containing the various maps and other additional information such as the case number, case name and date of the report can be generated by clicking on **Tools** then **Generate Report** in the menu. The report is in the form of an HTML file and can be viewed by selecting **Tools** then **View Report** from the menu. This report documents the pattern analysis process and can be used as a reference at a later stage. An example of the report is shown in Figure 6.15.

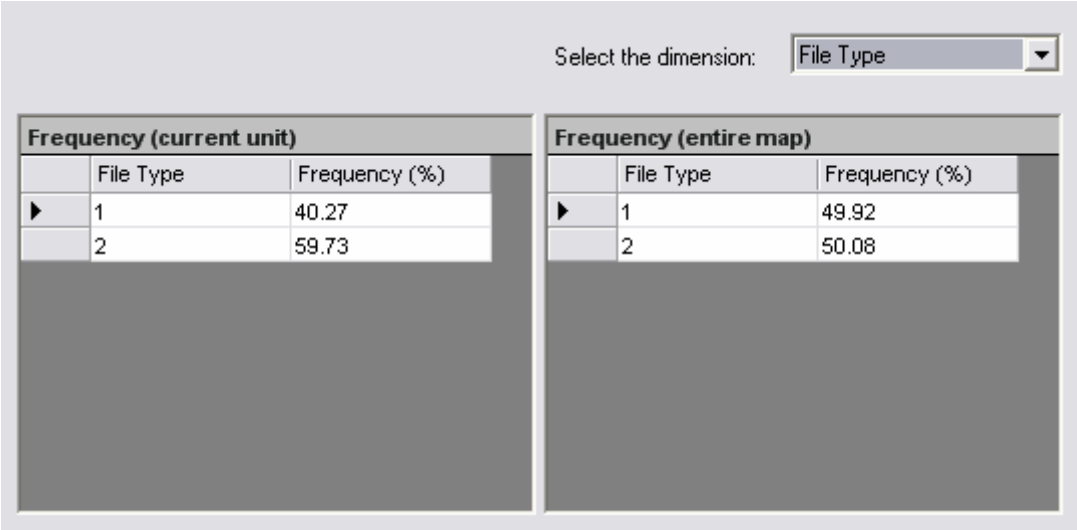


Figure 6.14: The SOMFA Tool – Statistics interface.

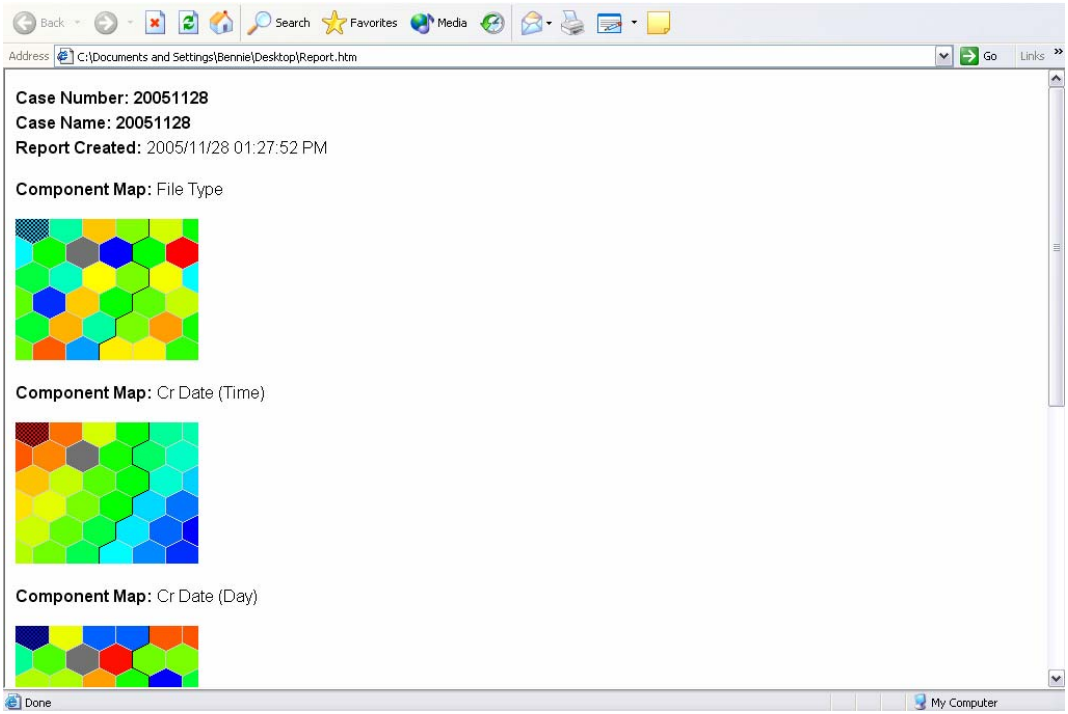


Figure 6.15: Viewing a report regarding pattern analysis.

6.7 Conclusion

This chapter has focussed on providing a functional overview of the SOMFA Tool and how it could potentially be applied to the forensic analysis process. An example of the forensic analysis process being conducted using the SOMFA Tool is shown in the screenshot in Figure 6.16.

The next chapter, Chapter 7, extends this discussion by experimenting with the SOMFA Tool. A discussion of the different experiments undertaken and the experimental results and findings is conducted. The objective is to demonstrate the application of the SOMFA Tool in the field of digital forensics.

Figure 6.16: Screenshot of the SOMFA Tool during forensic analysis.

University of Pretoria etd – Fei, B K L (2007)

Chapter 7

Experiments

This chapter describes the experiments conducted with the SOMFA Tool in order to demonstrate the application of the SOM in the field of digital forensics. To be more precise, the objective of this chapter is to demonstrate the application of data visualisation in digital forensics using the SOM, a well known neural network model. The experiments focus on the two main disciplines of digital forensics, namely, computer and network forensics. In addition, the ability to conduct interactive forensic analysis to improve the quality of digital investigations is shown.

The following sections, Sections 7.1, 7.2 and 7.3, discuss three independent experiments, including their experimental results and respective findings. Each of the following sections begins with a discussion of the setup of the experiment and concludes with a description of the three main phases: data pre-processing, pattern discovery, and pattern analysis.

7.1 Exploring forensic data

This section focuses on the application of the SOM to child pornography investigations and the forensic analysis of files found on a seized hard drive in particular. Obviously, these files may constitute evidence of illegal activity. In a child pornography investigation, a digital investigator must locate and examine

all available graphical images, discern possible patterns and study the behaviour of a suspect. In addition, the digital investigator is required to take into account any unethical activities found on the seized hard drive.

7.1.1 Experimental setup

The experimental setup was as follows:

- Forensic Toolkit (FTK) (AccessData Corporation, 2006) was used to create an image of the seized hard drive with a capacity of 100GB.
- Once the image had been created, it was analysed for evidence. Using FTK, the search was narrowed down to areas where the likelihood of evidence residing was high. In this case, it was narrowed down to graphical images. However, a large number of MP3 (music) files were also found on the seized hard drive. This information was also considered since downloading MP3 files, especially when large numbers of MP3 files are downloaded in a short period of time, is also deemed a potentially unethical activity. It should be noted that even if the file extensions are modified by the user, FTK is able to detect the correct format of each file.
- Once all the graphical images and MP3 files had been located, the information regarding these files was saved to a text file using FTK. This data file was subsequently processed using the SOMFA Tool. This process is discussed in detail in the following subsections.

7.1.2 Data pre-processing

First, the data file created by FTK, containing a total of 3,510 entries, was loaded into the SOMFA Tool. The data file contained data on all the graphical images and MP3 files found on the seized hard drive and included the three fields listed below:

- `File name` (used only for file identification).
- `File type`.

- `File creation date`.

The original state of the data file was such that data pre-processing was required. Data pre-processing simplifies the learning process of the SOM and reduces processing time. Data transformation was performed on the various fields through the data pre-processing capabilities of the SOMFA Tool. The `File type` fields were converted into numerical values, each numerical value representing a particular file type. In this instance, 1 represented graphical images and 2 represented MP3 files. The `File creation date` fields were converted and expanded into three separate fields, namely, `File creation date`, `File creation time` and `File creation day`. Afterwards, the `File creation date` and `File creation time` fields were reformatted as `yyyymmdd` and `hhmm` respectively.

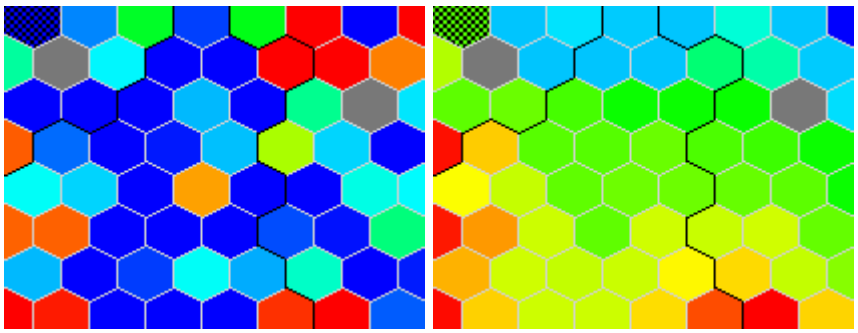
7.1.3 Pattern discovery

Pattern discovery using the SOM occurs after the data pre-processing phase. Only the fields required for analysis were included in the learning process. These were `File creation date`, `File creation time` and `File creation day`. The learning process of the SOM progresses until the SOM reaches an accurate result or until a given maximum number of learning iterations has been reached. This number is set to twice the number of input patterns by default. In this case, the number was 7,020. As mentioned earlier, there is no specific rule for choosing the number of learning iterations. However, the default number should allow the SOM to reach an acceptable result.

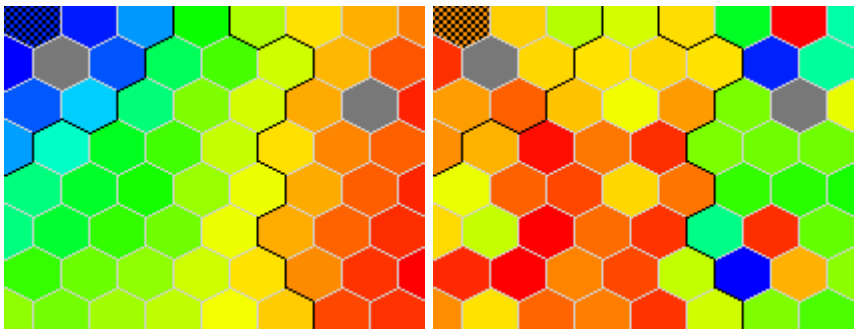
7.1.4 Pattern analysis

Two-dimensional maps, which are displayed as hexagonal grids in Figure 7.1, are generated after the SOM completes its learning process. These maps, known as component maps, reveal variations in the values of components across the map. Each component map visualises the spread of values in a particular dimension. In each map, the colour blue indicates low values, red high values and the other colours (such as green and yellow) represent intermediate values. The colour

grey indicates that no data is mapped to that particular unit (or hexagonal grid). The cluster boundaries are indicated with a solid black line.



(a) Component map (File type); (b) Component map (File creation date).



(c) Component map (File creation time); (d) Component map (File creation day).

Figure 7.1: Component maps generated from forensic data.

Figure 7.1(a) reveals variations in the file type. In this instance, blue indicates graphical images while red indicates MP3 files and the other colours indicate that a combination of graphical images and MP3 files are mapped onto that particular unit.

Figure 7.1(b) reveals variations in the file creation dates. Blue indicates small values (older files with earlier creation dates) while red indicates large values (new files). Therefore, the most recent files are displayed in the lower half of the map (represented in yellow and red). The upper half of the map reveals the files that were created earlier (indicated in blue).

Figure 7.1(c) reveals variations in the file creation times. The map has three portions: blue (top left), green (bottom left) and red (right). The blue portion denotes the 12am to 6am time period; green denotes 6am to 7pm time period, and red, 7pm to 12am. Upon viewing the map, it is immediately obvious that the green portion is significantly larger than the others, implying that files were mainly created between 6am to 7pm.

Figure 7.1(d) reveals variations in the specific days on which the files were created. The fact that the majority of the map is covered in red suggests that the majority of the files were created late in the week and over the weekend: Friday, Saturday and Sunday.

The four component maps form the backbone of the forensic analysis process and were used in conjunction. Interactive analysis was conducted by clicking on various units in the maps and reviewing the detailed information regarding the selected unit that appeared. This information included the data or input patterns mapped onto that specific unit and statistics regarding the selected unit (see Figure 7.2, for example). By comparing the different maps, correlations and patterns were revealed.

From a general perspective, it became clear that the majority of the file content was graphical images (as indicated in Figure 7.1(a)). This was confirmed by the statistics function of the SOMFA Tool, which showed that about 70% of the file content was graphical images (see Figure 7.3). Most of these images were either created or downloaded during Fridays, Saturdays and Sundays (as indicated in Figure 7.1(d)). By comparing Figures 7.1(b) and 7.1(c), a correlation between file creation dates and file creation times was detected. Most of the recent files were created between 6am to 12am, suggesting that the majority of recent activities took place during that time period. Also, by examining Figures 7.1(a) and 7.1(b), it was possible to discern a pattern concerning when the MP3 files were created. The MP3 files were either very recent or very old in terms of dates in which they were created or downloaded. Furthermore, the majority of the MP3 files were created late in the week.

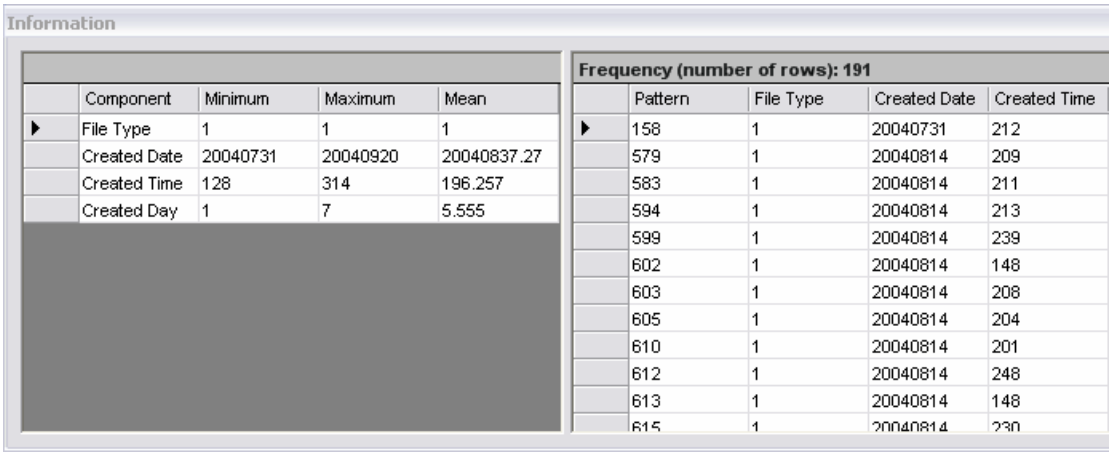


Figure 7.2: Information regarding the selected unit.

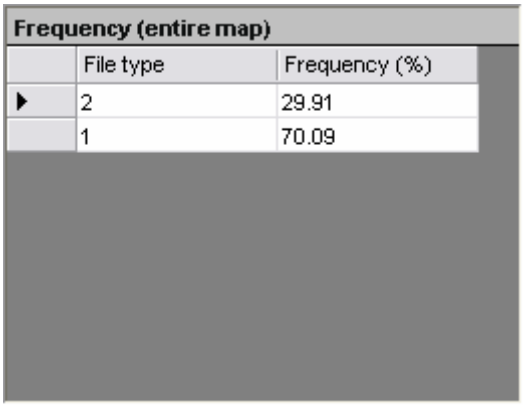


Figure 7.3: Statistics interface indicating the percentages regarding file type.

From a forensic perspective, certain areas were identified which were likely to lead to digital evidence recovery. For example, the graphical images created between 12am to 6am, since the time period does not correspond to normal waking hours. This suggested the investigation should focus on the blue portion in Figure 7.1(c). This could be further refined using the date of the incident. The investigator could focus on graphical images that were created recently, for example, if an incident is believed to have occurred recently. This could be done by analysing the red portions of Figure 7.1(b). The red portions of Figure 7.1(a) also seemed deserving of extra attention because they refer to the downloading of MP3 files and this is deemed an unethical activity.

To conclude, possible correlations and several patterns were discovered in the forensic data. The locating of points of interest was also achieved quickly and easily by viewing the different maps and also through interaction with the maps. Both of which will help digital investigators locate vital information and plan the next step in their search. The SOM was used to explore forensic data and provided unobtrusive insights to the forensic data, which made, in this case, a child pornography investigation more effective and efficient.

7.2 Detecting anomalous behaviour

Employees with access to the Internet via their computer system at work can use the World Wide Web as an important resource. However, as stated earlier, excessive Internet usage for non-job purposes and the deliberate misuse of the Internet, such as accessing Web sites that promote pornography and other unethical activities, has become a serious problem in many organisations.

Since storage media are steadily growing in size, forensic analysis of a single machine is becoming increasingly cumbersome. Moreover, the process of analysing or investigating a large number of machines has become extremely difficult or even impossible. However, what remains of chief importance in this environment is the detection of suspicious behaviour.

During a digital investigation, the forensic analysis of temporary Internet files can be very useful when evidence of excessive or inappropriate Internet usage is being searched for. Most of the temporary Internet files stored on a machine are “image captures” of sites that the user has visited. These files reveal a substantial amount of information about the browsing history of a user and analysing them can be useful in proving a pattern of logon times and durations. The focus of this section is to demonstrate how anomalous browsing behaviours can be detected in a more efficient manner when analysing multiple computer systems.

7.2.1 Experimental setup

The experimental setup was as follows:

- Four computer users or systems were selected within an organisation. All were operating on the Windows platform and were used by individuals who have been given a similar work task.
- FTK was used to create images of the four independent hard drives found in each computer system. Each with a capacity of 70GB.
- Once the images had been created, they were analysed for evidence of excessive Internet usage for non-job purposes. FTK was used to create a text file containing information regarding the files found in the Temporary Internet Files folder within each of the images. These data files (one for each computer system) were subsequently processed by the SOMFA Tool independently.

7.2.2 Data pre-processing

The data files were loaded onto the SOMFA Tool. Each data file included the four fields listed below:

- `File name` (used only for file identification).
- `File type`.
- `File creation date`.
- `Logical file size`.

Similar to Section 7.1.2, data pre-processing was performed on each of the data files. The `File type` fields were converted into numerical values, each numerical value representing a particular file type. The `File creation date` fields were converted and expanded into two separate fields, namely, `File creation time` and `File creation day`. Afterwards, the `File creation time` fields were converted into `hhmm` format. Finally, the `Logical file size` field was converted from bytes to KB.

7.2.3 Pattern discovery

Pattern discovery using the SOM occurs after the data pre-processing phase. The data files were processed independently. The processing time for each one was different since the default number of learning iterations fluctuated due to the differences in size between the data files. For example, in one data file, there are a total of 9,985 entries and in another data file, there are a total of 5,273 entries. The learning process finished when the pre-set limit in the number of learning iterations for each data file was reached.

7.2.4 Pattern analysis

The Internet behaviour of each computer user was ascertained by analysing the different component maps. For each computer system, four component maps were presented (see Figures 7.4, 7.5, 7.6 and 7.7). The first component map represented the file type, for example, documents or graphical images; blue indicating that the majority of fields in that particular unit were documents and red that the majority were graphical images. The second component map represented the time when the temporary Internet files were created, that is, the time of day when Internet activities occurred. In this instance, blue indicated the early hours of the morning (after midnight). As the time of day progressed, the colour would change from blue to green, and eventually red. The third component map represented the days of the week during which the temporary Internet files were created. The colour blue indicated that the majority of the files were created at the beginning of the week, that is, on Monday or Tuesday, while green indicated that the majority of the files were created in the middle of the week. Lastly, red revealed that the majority of the files were created later in the week, that is, between Friday and Sunday. The fourth and final component map represented the logical file size of the files; blue indicating that the logical file size (in terms of KB) was small and red indicating that the logical file size was large.

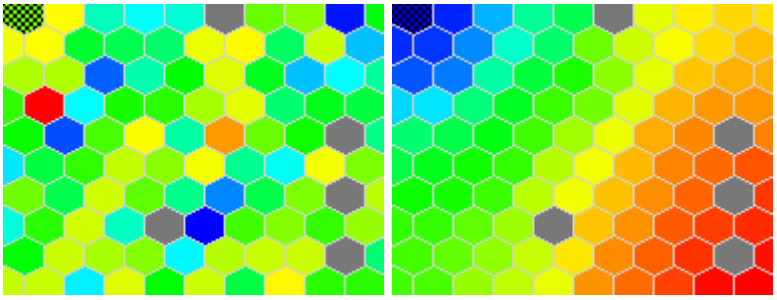
The objective of this investigation was to study the behaviour of each computer user. This suggested that it would be appropriate to analyse the second and third maps in detail because the time and day of the week would be of paramount importance when determining the behaviour of the different computer

users. However, the logical file size was also taken into account since it also gave some indication of the kind of behaviour of the different users.

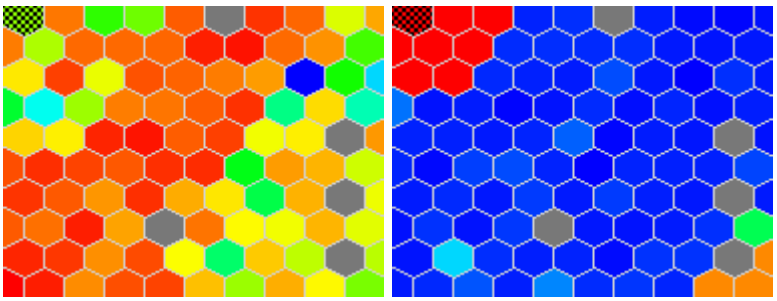
The results of the four computer systems are presented first, followed by a discussion based on the findings of the results presented.

First computer system

For the first computer system, the following maps were generated:



(a) Component map (File type); (b) Component map (File creation time).



(c) Component map (File creation day); (d) Component map (Logical file size).

Figure 7.4: Component maps generated for the first computer system.

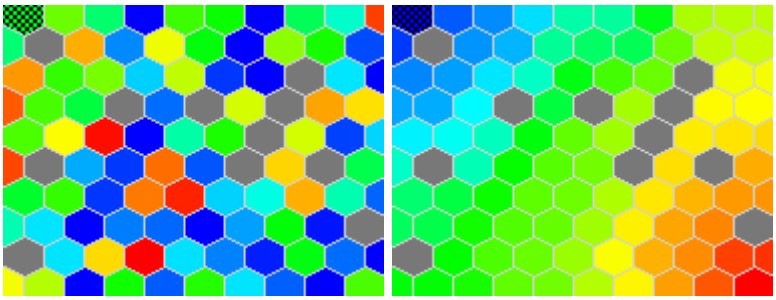
In Figure 7.4(b), the blue portion of the map denotes the period between 12am and 6am; the green portion points to the period between 6am and 7pm, and the red portion represents the period from 7pm to 12am.

In Figure 7.4(c), a significant portion of the map is shown in red, which indicates that the majority of the Internet activities occurred on Fridays, Saturdays and Sundays.

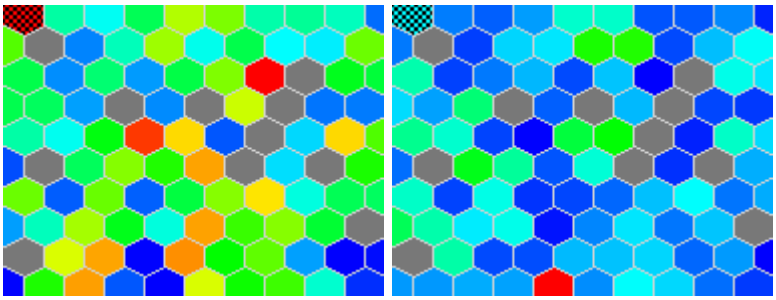
By analysing (or comparing) Figures 7.4(b) and 7.4(c), it appears that the green portion in Figure 7.4(b) correlates with the red portion in Figure 7.4(c). This suggests that the majority of Internet activities took place during the weekends between 6am and 7pm, while most Internet activities occurred during the week took place at night, between 12am and 6am and again from 7pm to 12am.

Second computer system

For the second computer system, the following maps were generated:



(a) Component map (File type); (b) Component map (File creation time).



(c) Component map (File creation day); (d) Component map (Logical file size).

Figure 7.5: Component maps generated for the second computer system.

In Figure 7.5(b), the blue portion of the map represents the period from 7am to 12pm; the green portion refers to the period between 12pm and 4pm, and the red portion denotes the period from 4pm to 8pm. Through the use of colour coding, it becomes immediately apparent that the green portion is significantly

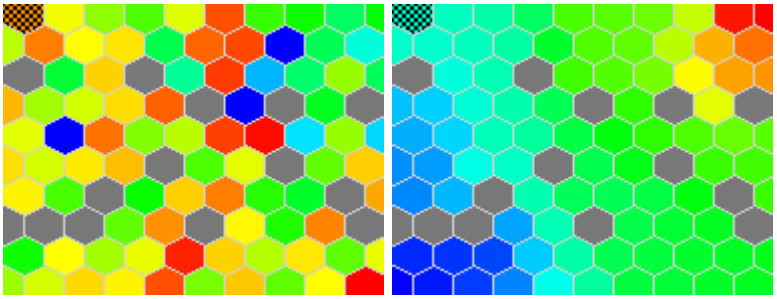
larger than the rest – implying that Internet usage was heaviest in the afternoons, between 12pm and 4pm.

Figure 7.5(c) suggests that the Internet activities of the second computer user were spread fairly evenly across the different weekdays, except that not much activity occurred on Fridays. This is clearly shown on the map since only a few red units can be found.

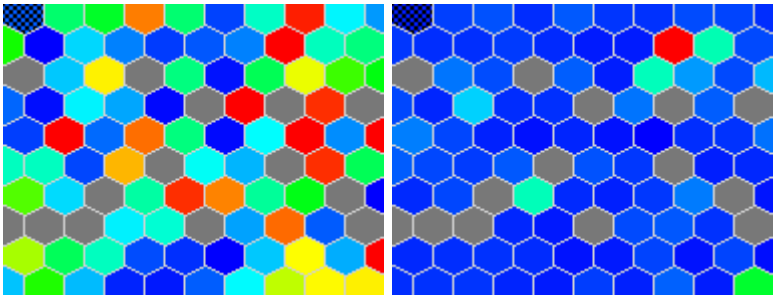
By analysing or comparing Figures 7.5(b) and 7.5(c), several correlations were found. Firstly, it appears that the second computer user is only logged on until late, 8pm, on a Monday. This is shown by the red units in Figure 7.5(b), located at the bottom right hand corner of the map, and the blue units in Figure 7.5(c), located at the bottom right hand corner of the map. Secondly, the red units in Figure 7.5(c) indicate that on Fridays, the user logs on from 7am until 3pm. This gives a clear indication about the times and durations that the user was logged on for.

Third computer system

For the third computer system, the following maps were generated:



(a) Component map (File type); (b) Component map (File creation time).



(c) Component map (File creation day); (d) Component map (Logical file size).

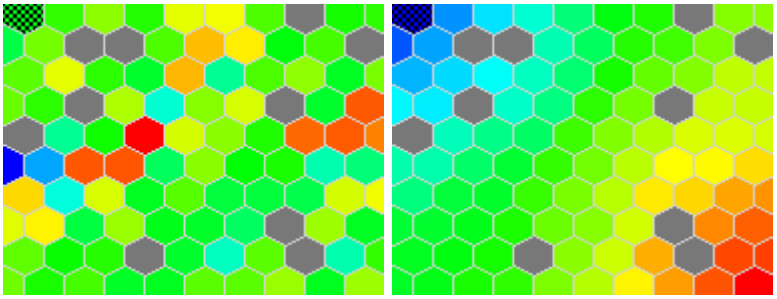
Figure 7.6: Component maps generated for the third computer system.

In Figure 7.6(b), the blue portion of the map represents the period from 9am to 12pm, the green portion stands for the period from 12pm to 3pm and the red portion represents the period between 3pm and 6pm.

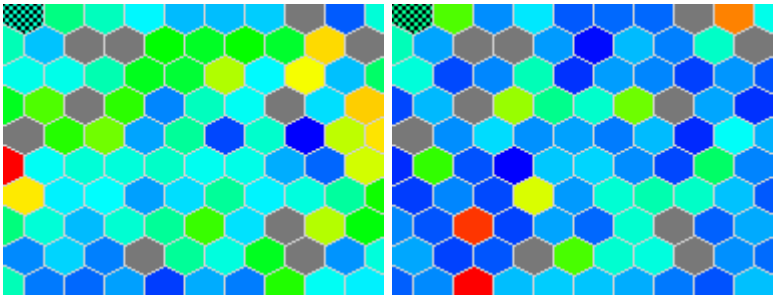
In Figure 7.6(c), more than half of the map is covered in blue (or shades of blue). This means that over 50% of the Internet activities took place on Mondays and Tuesdays. Later on in the week all use of the Internet seemed to dwindle.

Fourth computer system

For the fourth computer system, the following maps were generated:



(a) Component map (File type); (b) Component map (File creation time).



(c) Component map (File creation day); (d) Component map (Logical file size).

Figure 7.7: Component maps generated for the fourth computer system.

In Figure 7.7(b), the blue portion of the map represents the period between 7am and 12pm, the green portion denotes the period from 12pm to 4pm and the red portion indicates the period between 4pm and 9pm.

According to Figure 7.7(c), the Internet activities of the fourth computer user seem to be distributed evenly across the different days of the week.

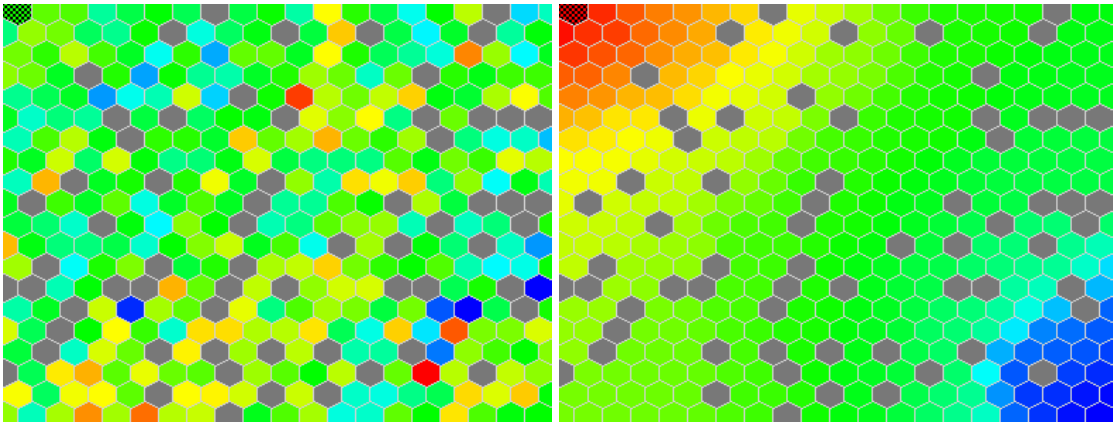
Discussion

After analysing the four independent computer systems with the SOMFA Tool, the behaviour of each computer user was noted. Earlier it was remarked that the four computer systems were used by individuals who had a similar work task. Therefore, it would be expected that the four computer systems would display

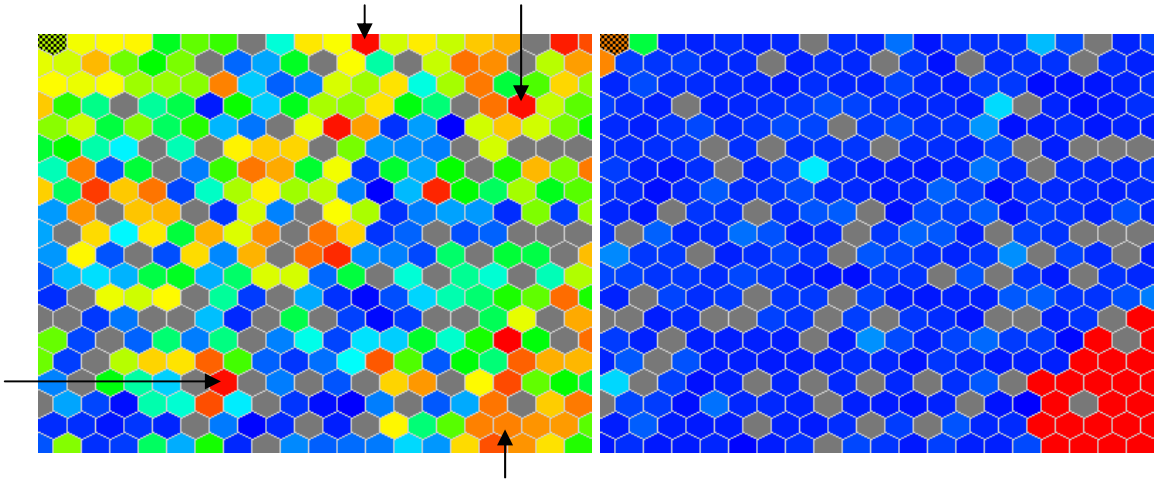
similar behaviours. Based on the above observations, it is clear that anomalous behaviour was found in the first computer system. This is because the behaviour of the user of the first computer system deviates significantly from that of the users of the other three computer systems, who share similar behaviours. This is clearly seen when viewing the maps that represent the days that files were created on the various systems. Furthermore, by comparing the maps that represent the logical file sizes created on the different computer systems, the first computer system stands out again. From Figure 7.4(d), it can be seen that during the period from 12am to 6am and 7pm to 12am, the logical file size is fairly large.

In order to confirm that the behaviour of the first computer system's user was anomalous when compared with the others, the data files of the four computer systems were combined and processed by the SOMFA Tool. After the learning process had completed, maps were generated (see Figure 7.8). Note that an additional map was generated. The reason for this is that an additional dimension was included in the learning process. This additional component map reveals the value variation of the temporary Internet files belonging to a specific computer system that were created at a specific time (see Figure 7.8(e)). In Figure 7.8 (e), blue indicates the first computer system, while red indicates the fourth computer system and the other colours indicate the second and third computer systems.

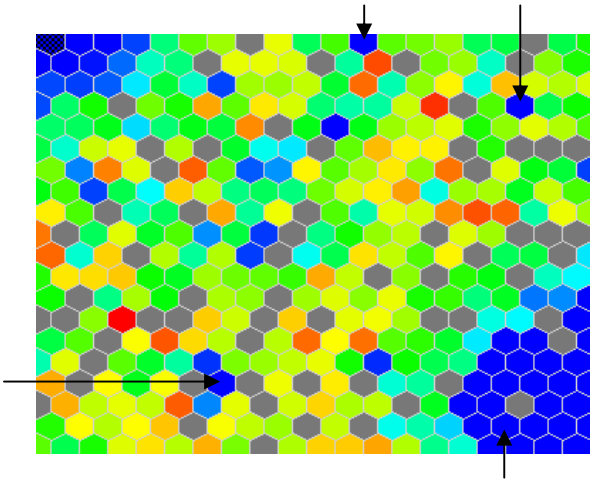
By comparing Figures 7.8(b), 7.8(c), 7.8(d) and 7.8(e), possible correlations can be detected, for example, between the times and days on which Internet activities of a specific computer system took place. By looking at the maps, the anomalous behaviour of the first computer system can be immediately detected. Firstly, the first computer system (represented by blue – see Figure 7.8(e)) is the only one where Internet activities took place between 9pm and 12am (top left of Figure 7.8(b)) and between 12am and 6am (bottom right of Figure 7.8(b)). Secondly, it can be seen that during those two periods, the logical file sizes are significantly larger (see Figure 7.8(d)). Thirdly, by comparing Figures 7.8(c) and 7.8(e), it is found that the majority of the red units in Figure 7.8(c) correlate to the blue units in Figure 7.8(e) (indicated with arrows). This implies that the first computer system has made use of the Internet mostly over weekends (when hardly anyone else was around).



(a) Component map (File type); (b) Component map (File creation time).



(c) Component map (File creation day); (d) Component map (Logical file size).



(e) Component map (Computer system).

Figure 7.8: Component maps generated for all computer systems.

Given that an anomalous behaviour has been discovered, further investigation would now be needed to determine the reasons behind the anomaly. The individual using the first computer system could well be using the Internet for inappropriate or illegal activities, but the specific reasons for these anomalies are beyond the scope of this investigation. Nonetheless, this section has shown the possible contribution of the SOM to large-scale forensic analysis. More specifically, this section has demonstrated that the SOM can be an efficient aid when digital investigators are searching for anomalous behaviours among the Internet browsing behaviour of individuals within an organisation. Once the suspicious computer system has been identified, digital investigators are free to proceed with the next step in their investigation.

7.3 Analysing Web proxy data

A fundamental goal in network forensics is to gather evidence. As mentioned earlier, evidence can be gathered from various sources, for example, at the server level, evidence can be obtained from Web server logs that record the browsing behaviour of site visitors. Evidence can be also gathered from usage data provided by packet sniffers that monitor network traffic to a Web server.

This section deals with network forensics. More specifically, it deals with the forensic analysis of data on a Web proxy. The purpose of a Web proxy is to relay a request in the form of a uniform resource locator (URL) (Berners-Lee et al., 1994) from a client to a server, receive the response from the server and send it back to the client (Maltzahn and Richardson, 1997).

What is significant about forensic analysis of data on a Web proxy – as opposed to on a single computer system or on multiple computer systems – is that, while computer users can delete traces of their requests on their own computer system, Web proxy data pertaining to URL requests made by users is generally accessible only to network administrators and digital investigators.

Another benefit is that digital investigators can focus on a single point in the network topology, which saves time that might be crucial to the investigation. For example, they can focus on employees in an organisation who access Web sites that promote illegal activities. In their investigation they could make use of

data gathered at the Web proxy. Since the seizing of computer systems used by employees is not necessary for evidence recovery, the investigation can be performed without the employees even knowing that they are being investigated. In some cases, the organisation can perform an internal investigation without having to alarm the employees. Thus, employees' daily tasks would not be affected until supporting evidence has been found. The digital investigators will have all the data from which they can draw their conclusions, whereas in the case of computer systems, employees might have been able to delete all traces of their illegal activities.

This section begins with an assessment of related work that deals with the visualisation and statistical analysis of logs. It also assesses related work that deals with the SOM applied to Web data. This is followed by the experimental setup and the three main phases as before: data pre-processing, pattern discovery and pattern analysis.

7.3.1 Related work

Several tools have been proposed that are capable of providing the visualisation and statistical analysis of logs. Examples of such tools are MieLog (Takada and Koike, 2002a), Tudumi (Takada and Koike, 2002b), and NVisionIP (Lakkaraju et al., 2004). The differences between these tools are in terms of their visual representation and functionalities. For example, MieLog includes functionalities like information visualisation and statistical analysis, but Tudumi includes functionalities like log summarisation and reflecting known rules into the visualisation method as well as information visualisation.

Related work on using the self-organising map (SOM) in Web mining focuses on performing the following:

- clustering Web pages according to the computer users' navigation behaviours (Smith and Ng, 2003)
- clustering Web server logs to discover usage patterns (Wang et al., 2005a)
- constructing profiles of computer users based on their search histories (Ding et al., 2005)

- organising Web documents according to the content of the documents (Kohonen et al., 2000)
- organising Web usage data into clusters (Mobasher et al., 2000)

Using the SOM in the way that is purposed in this study for network forensics has not been examined. The SOM is used as a visualisation platform offering a powerful framework for visualising and analysing the large volumes of data often encountered during network forensics.

7.3.2 Experimental setup

The experimental setup was as follows:

- The web proxy logs of twenty computer users in an organisation were taken. These proxy logs, which were generated by a Squid proxy (Wessels, 2005) over a period of one month, contained data pertaining to 374,620 HTTP requests. A sample of the proxy logs is shown in Figure 7.9.
- These logs were subsequently processed by the SOMFA Tool. This process is discussed in the following subsections.

7.3.3 Data pre-processing

In terms of network forensics, it is generally more practical to analyse subsets when involving large quantities of data. Eliminating irrelevant data simplifies the learning process. A Squid proxy log has the following format (the description of each field is depicted in Table 7.1):

```
time:etime:client:log-tag:size:request:url:userid:hierarchy
```

```
1120474306.106 21 137.215.41.128 TCP_MISS/200 4721 GET
http://www.bbc.co.uk/home/images/live8.jpg bfei FIRST_UP_PARENT/cache.up.ac.za
image/jpeg
1120474306.199 92 137.215.41.128 TCP_MISS/200 5996 GET
http://www.bbc.co.uk/home/images/live8_bground.jpg bfei
FIRST_UP_PARENT/cache.up.ac.za image/jpeg
1120474306.228    27 137.215.41.128 TCP_MISS/200 4752 GET
http://www.bbc.co.uk/home/images/wonderland/rhs_africa_02.gif bfei
FIRST_UP_PARENT/cache.up.ac.za image/gif
1120474306.388    241 137.215.41.128 TCP_MISS/200 1012 GET
http://www.bbc.co.uk/home/symbols/sml/10.gif bfei FIRST_UP_PARENT/cache.up.ac.za
image/gif
```

Figure 7.9: A sample of the Squid proxy logs.

Table 7.1: Description of each field of the Squid proxy log format.

Field	Description
time	Timestamp of the request in seconds since 1970
etime	Elapsed time of the request in milliseconds
client	IP client address
log-tag	Log tag and HTTP code
size	Number of bytes written to the client
request	The HTTP request method
url	The requested URL
userid	The user that made the request
hierarchy	The request object that was fetched and the content-type field of HTTP reply

Only certain data fields are required to analyse the browsing behaviour of computer users and the Internet usage of an organisation. For example, the `client` and `userid` fields in the Web proxy logs refer to the same person. Therefore, only one field is required and the other can be classified as irrelevant data. During the data reduction process, the following fields were deleted:

- `etime`
- `client`

- `log-tag`
- `request object`

Next, certain data transformations were performed on the Web proxy logs. The timestamp of each request was converted into a more readable format. For example, `1121182139` was transformed into `2005/07/12 15:28 Thursday`. Furthermore, strings in the Web proxy logs were converted to numerical values. For example, the day of the week and the content type of the HTTP reply were replaced with a numerical value, as depicted in Tables 7.2 and 7.3 respectively. During the data transformation process, the following columns were transformed:

- `time`: the dates were converted into three columns, namely, `date`, `time` and `day`.
- `size`: the size of the request was converted to KB.
- `request`: fields were replaced with numerical values.
- `url`: the URLs were transformed so that each URL was restricted to its domain name and subsequently replaced with numerical values.
- `userid`: fields were replaced with numerical values.
- `content type`: fields were replaced with numerical values.

Table 7.2: Numerical values substituted for the day of the week.

Content type	Numerical value
Monday	1
Tuesday	2
Wednesday	3
Thursday	4
Friday	5
Saturday	6
Sunday	7

Table 7.3: Numerical values substituted for the content type.

Content type	Numerical value
text	1
unknown	2
application	3
image	4
video	5
audio	6

7.3.4 Pattern discovery

Earlier it was remarked that pattern discovery draws on algorithms used in data mining, machine learning and so on. Pattern discovery using the SOM occurs after data pre-processing. During the pattern discovery phase, the default number for the learning iterations was 749,240. Furthermore, the fields to be included in the learning process were `time`, `day`, `request`, `content type` and `size`.

7.3.5 Pattern analysis

Two-dimensional maps displayed in the form of hexagonal grids were generated once the learning process had completed (see Figure 7.10). Using the SOM, the multi-dimensional data residing in the Web proxy logs was successfully transformed into a two-dimensional visualisation.

Figure 7.10(a) reveals variations in the time that computer users made HTTP requests. The map has three portions: blue (bottom left), green (middle) and red (top right). The blue portion denotes the 12am to 9am time period; green denotes 9am to 3pm, and red, 3pm to 12am. Upon viewing the map, it is immediately obvious that the green portion is significantly larger than the others, implying that Internet usage mostly occurred from 9am to 3pm. This fact can be very useful in guiding digital investigators to the next step in their investigation. For example, if the focus of the investigation is on the time period when Internet usage mostly occurs, then the green portion of the map should be the centre of attention.

Figure 7.10(b) presents variations in specific days that HTTP requests were made. HTTP requests occurred mainly in the middle of the week: Tuesday, Wednesday and Thursday (represented by green), as opposed to Friday, Saturday and Sunday (represented by red).

Figure 7.10(c) reveals the variations in HTTP requests. POST and GET were common requests (indicated by blue, green and yellow). Usually, the GET method is a request for a specific URL; Figure 7.10(c) clearly shows that is, in fact, the case since there are more units with the colour blue and green when compared to other colours in the map.

Figure 7.10(d) presents variations in the type of data returned by HTTP requests. The map shows that the content is predominantly graphical images, applications and text (green and yellow), with text being the most common.

Figure 7.10(e) reveals variations in the URLs of the requests. The requests involved 4,263 distinct domain names, which were replaced with numerical values. Most of the requests involved domains that were mapped to lower numerical values (blue and green). The most popular domain was `www.google.co.za` (represented by 1), which was accessed 43,952 times (approximately 10% of requests).

Figure 7.10(f) presents variations with regards to which users made the HTTP requests. In particular, the map provides information about user browsing behaviour and Internet usage.

Figure 7.10(g) reveals variations in the size of data returned by HTTP requests. At least 90% of the map is blue, corresponding to HTTP requests that returned data less than 1,000KB in size. However, there are approximately three units that have a colour other than blue; corresponding HTTP requests were for data ranging from 1,000KB to 7,000KB.

The seven component maps are used in conjunction and together form the backbone of the forensic analysis. By comparing the maps, the investigator can review Internet usage and analyse the browsing behaviour of users. The browsing behaviour includes HTTP request times and URLs, the type of data requested from specific URLs and the size of data requested from specific URLs. Furthermore, to facilitate easier analysis, a frequency map was also generated (shown in Figure 7.11). The frequency map was used to reveal the approximate

data distribution among the units in the map and obtaining an overall view of the Web proxy data as well. A darker red colour indicates a larger frequency and a lighter colour indicates a smaller frequency of HTTP requests for a unit. In Figure 7.11, it is clearly apparent where most of the HTTP requests were made.

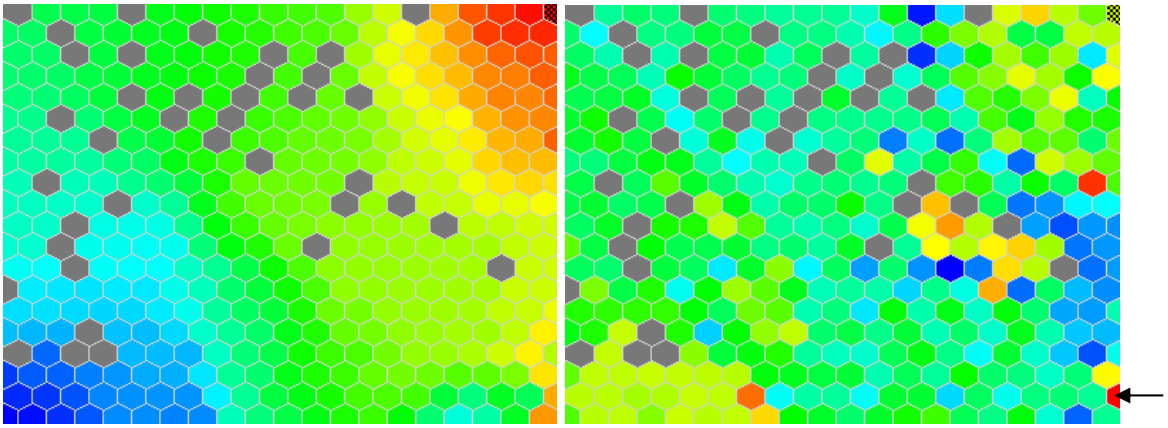
Through the use of colour coding, several patterns or correlations were identified. Firstly, by viewing Figures 7.10(c), 7.10(e), 7.10(f) and 7.11, a definite shape is formed. This shape is clearly seen in Figure 7.10(c) (represented by yellow) and in Figure 7.11 (the portion with the lighter colours). Another correlation is shown at the bottom left of Figures 7.10(a) (represented by blue), 7.10(b) (represented by green-yellow), 7.10(c) (represented by blue) and 7.10(d) (represented by cyan). Furthermore, the red regions (middle right) in Figures 7.10(e) and 7.10(f) indicate that there is a group of users that only visit specific URLs.

Detecting anomalous browsing behaviour is important in many network forensic investigations. Ideally, this is accomplished by examining the irregular portions of component maps. That is the regions within a map where a specific colour occurs less frequently. For example, in Figure 7.10(b), the irregular portions are the two red portions which indicate that the HTTP requests were made on Saturday. The use of this map, in conjunction with the other maps, reveals possible correlations between the various dimensions. It appears that there is a correlation between the red portion, located at the bottom right corner, in Figures 7.10(b) and 7.10(d) (indicated with arrows). Upon investigating the URLs and the type of data requested, it was found that suspicious activities had in fact occurred. First, Figure 7.10(d) indicates that the majority of the requests were for graphical images. Second, Figure 7.10(g) indicates that the size of the requests in that region is fairly large when compared to other regions. On examining the original proxy logs, it was observed that a particular user visited several adult Web sites on a Saturday and the contents retrieved by the user were mainly graphical images.

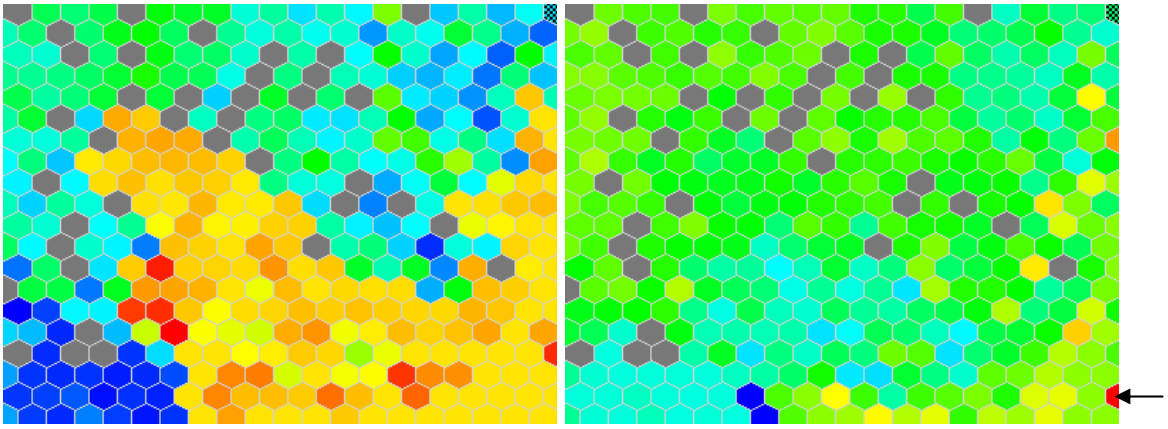
In Figure 7.10(c), the irregular portions are indicated in red, which corresponds to the CONNECT method. The CONNECT method is normally used to tunnel a connection through an HTTP proxy. By investigating the URLs and the type of data requested, it was found that a user was conducting Internet banking, which was not deemed to be an unauthorised activity.

Although certain activities deviate significantly, they may not necessarily be unauthorised or illegal. In the example investigation above, one anomalous incident involved Internet banking while the other involved visits to adult Web sites. Therefore, when anomalous activity is suspected, it is still necessary to conduct a detailed examination of the Web original proxy logs.

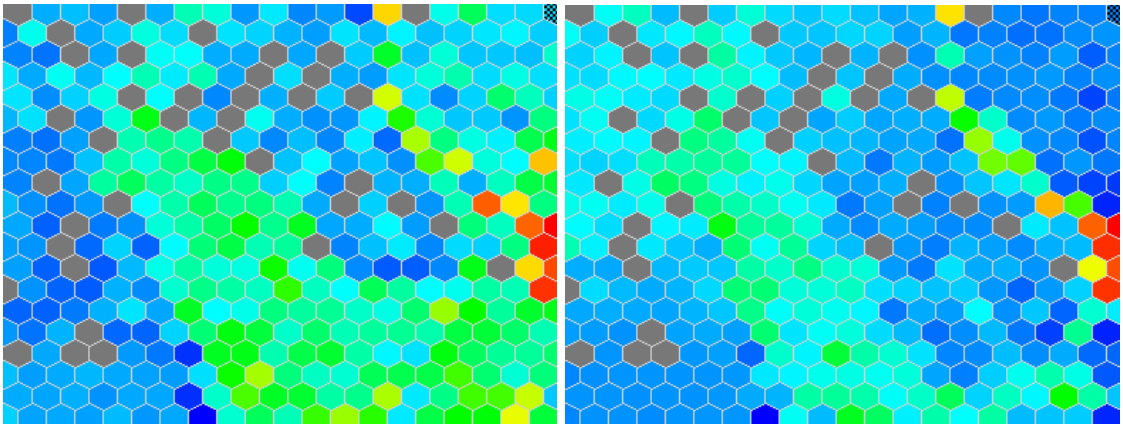
To conclude, the component maps offer a powerful framework for visualising and analysing the large volumes in data contained in Web proxy logs. By comparing the different component maps against each other, and also with the frequency map, a digital investigator can rapidly obtain an overview of Internet usage and an understanding of the browsing patterns of computer users, including anomalous behaviour. Only when anomalous behaviour is indicated, does it become necessary for digital investigators to conduct a detailed analysis of the Web proxy logs. This can contribute to an increase in the quality of digital investigations and a reduction in the amount of effort, especially in network forensics, which involves the collection and analysis of large quantities of data.



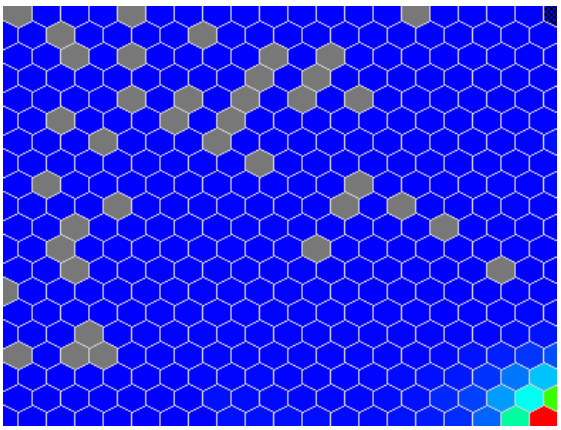
(a) Component map (time); (b) Component map (day).



(c) Component map (request); (d) Component map (content type).



(e) Component map (url); (f) Component map (userid).



(g) Component map (size).

Figure 7.10: Component maps generated from Web proxy data.

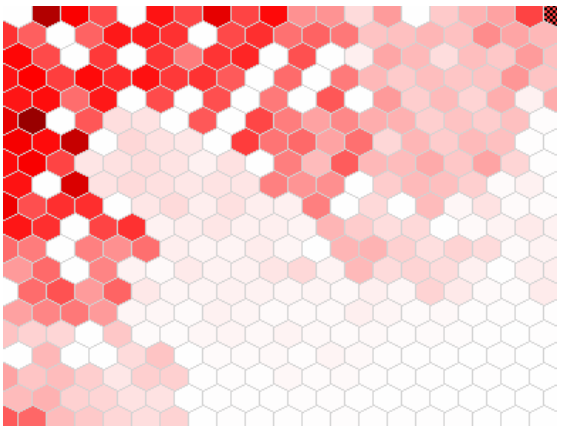


Figure 7.11: Frequency map of HTTP requests.

7.4 Conclusion

Three independent experiments were conducted using the SOMFA Tool. In each of the experiments, it was shown that the SOM offers great potential to the field of digital forensics. The SOM can be seen as a powerful framework for visualising and analysing the large volumes of data often encountered during forensic investigations. The visualisation capabilities of the SOM provide immediate insight into the forensic data. Furthermore, they enable the identification of correlations, patterns and anomalies within the forensic data. Moreover, through interactive analysis, the patterns that are discovered can be

quickly analysed to determine whether they are interesting or imperative. Only once suspicious events have been correctly identified is it necessary to conduct a detailed analysis of the evidence. This can help digital investigators decide upon the next step of their search and ensure that digital evidence recovery is carried out in a more efficient and effective manner.

University of Pretoria etd – Fei, B K L (2007)

Chapter 8

Conclusion

The SOM has several applications in digital forensics. These include identifying correlations in forensic data (association), discovering and sorting forensic data into groups based on similarity (classification), locating groups of latent facts (clustering), and discovering patterns in data that may lead to useful predictions (forecasting). While the SOM is ideal for association, classification, clustering and forecasting, it is also particularly useful for visualisation. Visualisation enables digital investigators to locate vital information that is of interest rapidly and efficiently. In addition, it guides digital investigators towards the best next step in their search so that digital evidence recovery is carried out in a more efficient and effective manner.

In this study, the SOM is used as a visualisation platform and was shown to offer a powerful framework for visualising and analysing forensic data. It was also demonstrated that, by providing new perspectives for visualising and analysing forensic data, it can facilitate the forensic analysis of the large volumes of data frequently encountered in digital investigations. It should be noted that visualisation is not advocated as a replacement the capabilities provided by existing computer forensic tools, but rather to function as a supporting tool for forensic analysis.

In concluding the investigation undertaken by this study, the research conducted needs to be summarised and related back to the problem statement

defined at the start in Chapter 1. By relating the investigation back to the problem statement, it will be possible to establish whether it has been successfully addressed or not. Once the research has been summarised, possible future work will be suggested.

8.1 Summary

In the opening chapter, Chapter 1, it was stated that digital forensics as a discipline faces several problems and that, among the more acute and limiting are the following:

- Digital investigations are becoming more time consuming and complex as the volumes of data requiring analysis continue to grow;
- Digital investigators are finding it increasingly difficult to use current tools to locate vital evidence within the massive volumes of data;
- In terms of network forensics, log files are often large in size and multi-dimensional, which makes the digital investigation and search for supporting evidence more complex.

Based on the above, several research questions were posited and addressed throughout this study. They are as follows:

- Are there ways to improve the efficiency and quality of forensic analysis?
- What are the current capabilities of computer forensic tools?
- Can the complexity that exists within log files be reduced?
- Have there been developments in other disciplines that could benefit digital forensics?
- What is the proposed solution and if there have been developments pertinent to the proposed solution, what are they?

The investigation began with an assessment of the current state of digital forensics by surveying the available literature. Key terms and concepts relating to the field of digital forensics that emerged during this survey were presented in Chapter 2. Once the key terms and concepts has been established and defined, the current capabilities of computer forensic tools was investigated in Chapter 3. It

was established that the forensic data requiring analysis in digital investigations is increasingly too large and complex for current computer forensic tools and that this can make conducting the forensic analysis in an effective and efficient manner impossible. The analysis of contemporary computer forensic tools was also used to identify and categorise the most important features of the tools. This resulted in a classification which facilitated the identification of the tools' common limitations and enable several improvements to be recommended. Based on the findings, it was clearly seen that current computer forensic tools focus primarily on digital evidence recovery. For example, the tools have the ability to acquire storage media and perform analysis on the acquired image. While excellent at these tasks, the tools were found to offer limited capabilities with regards forensic analysis. These limitations were discussed in Chapter 3.

The field of data mining was introduced in Chapter 4 and it was established that data mining techniques could offer potential benefits in the digital forensics arena. The various data mining functionalities available were described, with particular reference to those that have been applied in the digital forensics arena. This led to a discussion of Web mining since there is a correlation between Web mining and network forensics. Following from that, the SOM was proposed as a means of improving forensic analysis through pattern discovery. The SOM was proposed, specifically, as an expedient way of clustering, visualising and analysing forensic data.

In Chapter 5, the SOM was discussed in detail. The discussion started with a presentation of a typical SOM architecture and concluded with an examination of the learning process. Furthermore, it focused on the various visualisations the SOM can produce and how these make it ideal for supporting forensic analysis. The SOM has the ability to map high-dimensional data onto a low-dimensional space, typically, a two-dimensional space. As a result, multi-dimensional data can be transformed into a two-dimensional visualisation. This directly addresses the problem posed by the complexity that often exists within forensic data.

A prototype implementation, the SOM Forensic Analysis Tool (SOMFA), was presented in Chapter 6. Firstly, the motivation for the prototype implementation was discussed. Thereafter, the architecture of the SOMFA Tool was presented. This was followed by the system requirements of the SOMFA Tool and a comprehensive functional overview of the SOMFA Tool.

The theoretical benefits offered by the SOM were then tested in Chapter 7 using the prototype implementation. Three independent experiments were conducted in order to demonstrate the application of data visualisation in digital forensics using the SOM. The experiments focused on both computer and network forensics. For each experiment, the experimental setup was described and the subsequent findings analysed. Based on the experiments conducted, it was demonstrated that the SOM has several pertinent applications in digital forensics. Furthermore, it was conclusively demonstrated that the SOM can improve the efficiency and quality of forensic analysis.

8.2 Future work

Much work still remains to be done in the digital forensics arena. As this study has shown, there are many techniques being developed in other fields, like data mining techniques, which can support and improve digital investigations in myriad ways. There is also a great deal of work still needed to improve the efficiency of digital investigations and the quality of the decisions made.

With regards future studies into the application of the SOM in digital forensics, one possibility is to further examine the colour representation used by the SOM. For example, regions containing potential evidence could be assigned a specific colour to allow for a better visualisation and interpretation. Another possibility is to investigate the possibility of improving the SOM by introducing three-dimensional visualisations. This type of visualisation may allow for a clearer illustration of the underlying data structures. However, three-dimensional visualisations are likely to place strenuous demands upon performance.

Another avenue for future research is to develop and incorporate automated evidence recovery techniques within the SOMFA Tool and specialise it for all the major digital forensic processes.

The work done in this study has shown that the SOM has potential in the field of digital forensics. A drawback, however, is that the learning process of the SOM on a very large data set is computationally intensive and requires a fast processor and sufficient memory. Therefore, another potential area of research is to optimise the learning process when dealing specifically with forensic data.

This dissertation concludes with the following observation about the increasingly importance role digital forensics is set to play within society:

"Practically every crime now involves some aspect of digital evidence; digital forensics provides the techniques and tools to articulate this evidence."

Advances in digital forensics. Edited by Mark Pollitt and Sujeet Shenoi.

It is also noted that no work is the product of an entirely solitary effort and hoped that the work presented in this dissertation will stimulate further research that will contribute to the digital forensics community.

University of Pretoria etd – Fei, B K L (2007)

Glossary

Cookie file

A cookie file is a small piece of information that an HTTP server transmits to the browser upon the initial connection.

Cyclic Redundancy Check

Cyclic redundancy check is a form of checksum used to detect certain kinds of error (Sammes and Jenkinson, 2000).

Discriminant Analysis

Discriminant analysis is a classification technique used to predict a categorical response variable (Han and Kamber, 2005).

DoS Attack

A denial of service (DoS) attack is an attack in which a server is targeted to prevent legitimate users from using a service.

File Signature

A file signature is a unique hex header signature associated with a file type.

Flow

A flow can be considered as a related set of packets.

HTTP

A hypertext transfer protocol (HTTP) is a protocol used to transfer information on the World Wide Web (Berners-Lee et al., 1996).

IDS

An intrusion detection system (IDS) is a system that attempts to identify intrusions by analysing information from sources such as audit records, system tables and network traffic summaries (Puketza et al., 1996).

Logical File Size

Logical file size is the exact size of a file in bytes.

Metadata

Metadata is data in the file system that describes the layout and attributes of the regular files and directories (Buchholz and Spafford, 2004).

Neural Network

A neural network is composed of a layered network of neurons capable of learning and storing data (Engelbrecht, 2003).

R2L Attack

A remote-to-local (R2L) attack is an attack in which an attacker without user level access gains the ability to execute commands locally (Mahoney and Chan, 2002).

Router

A router is a device that forward data packets to its destination (Keshav and Sharma, 1998). It is connected to at least two networks and decides which direction to forward the data packet.

Packet Sniffer

A packet sniffer is a program designed to intercept traffic on wired or wireless networks and capture packets.

Slack Space

Slack space is the space from the end of a file to the end of the last cluster containing the file (Sammes and Jenkinson, 2000).

Swap File

A swap file is a hidden system file that is used for virtual memory (Prosise et al., 2003).

Temporary Internet files

Temporary Internet files are “image captures” of sites that the user has visited.

Thumbnail

A thumbnail is a smaller size version of a graphical image.

Unallocated Space

Unallocated space is the area not currently allocated to a file (Prosise et al., 2003).

U2R Attack

A user-to-root (U2R) attack is an attack in which an attacker with user level access gains the privileges of another user.

URL

Uniform resource locator (URL) is an identifier used to specify a particular page of the World Wide Web (Berners-Lee et al., 1994).

Appendix A

Papers Published

During the course of this study the following papers were prepared and published:

- Fei, B.K.L., Eloff, J.H.P., Venter, H.S. and Olivier, M.S. 2004. Classifying computer forensic tools with the aim of extending the functionality of EnCase. *Proceedings of the Annual Post Graduate Symposium of the South African Institute of Computer Scientists and Information Technologists Conference*.
- Fei, B., Eloff, J., Venter, H. and Olivier, M. 2005. Exploring forensic data with self-organising maps. *Advances in digital forensics*, pp. 113-123. Springer.
- Fei, B.K.L., Eloff, J.H.P., Olivier, M.S., Tillwick, H.M. and Venter, H.S. 2005. Using self-organising maps for anomalous behaviour detection in a computer forensic investigation. *Proceedings of the Fifth Annual Information Security South Africa Conference*.
- Fei, B.K.L., Eloff, J.H.P., Olivier, M.S. and Venter, H.S. 2006. Analysis of Web proxy logs. Accepted for presentation at the *Second Annual IFIP WG 11.9 International Conference on Digital Forensics*.
- Fei, B.K.L., Eloff, J.H.P., Olivier, M.S. and Venter, H.S. 2006. The use of self-organising maps for anomalous behaviour detection in a digital investigation. To be published in *Forensic Science International*.

University of Pretoria etd – Fei, B K L (2007)

Bibliography

Abraham, T. and de Vel, O. 2002. Investigative profiling with computer forensic log data and association rules. *Proceedings of the IEEE International Conference on Data Mining*, pp. 11-18.

Abraham, T., Kling, R. and de Vel, O. 2002. Investigative profile analysis with computer forensic log data using attribute generalisation. *Proceedings of the Australasian Data Mining Conference*.

Abraham, A. and Ramos, V. 2003. Web usage mining using artificial ant colony clustering and linear genetic programming. *Proceedings of the Congress on Evolutionary Computation*, pp. 1384-1391.

AccessData Corporation. 2005. (http://www.accessdata.com).

Agrawal, R., Imielinski, T. and Swami, A. 1993a. Mining association rules between sets of items in large databases. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 207-216.

Agrawal, R., Imielinski, T. and Swami, A. 1993b. Database mining: a performance perspective. *IEEE Transactions on Knowledge and Data Engineering*, vol. 5, no. 6, pp. 914-925.

Alsabti, K., Ranka, S. and Singh, V. 1998. An efficient k-means clustering algorithm. *Proceedings of the Eleventh International Parallel Processing Symposium*.

Altheide, C. 2004. Forensic analysis of Windows hosts using UNIX-based tools. *Digital Investigation*, vol. 1, no. 3, pp. 197-212.

Apple Computer, Inc. 2006. (http://www.apple.com).

Armor Forensics. 2006. (http://www.forensics-intl.com).

ASR Data Acquisition and Analysis, LLC. 2005. (http://www.asrdata.com).

Barbara, J.J. 2005. Digital evidence accreditation in the corporate and business environment. *Digital Investigation*, vol. 2, no. 2, pp. 137-146.

Baryamureeba, V. and Tushabe, F. 2004. The enhanced digital investigation process model. *Proceedings of the fourth Digital Forensic Research Workshop*.

Beebe, N.L. and Clark, J.G. 2004. A hierarchical, objectives-based framework for the digital investigations process. *Proceedings of the fourth Digital Forensic Research Workshop*.

Beebe, N. and Clark, J. 2005. Dealing with terabyte data sets in digital investigations. *Advances in Digital Forensics*, pp. 3-16. Springer.

Berendt, B. 2000. Web usage mining, site semantics, and the support of navigation. *Proceedings of the Second International Workshop on Visualizing Software for Understanding and Analysis*.

Berners-Lee, T., Masinter, L. and McCahill, M. 1994. Uniform resource locators (URL). RFC 1738, Internet Engineering Task Force.

Berners-Lee, T., Fielding, R. and Frystyk, H. 1996. Hypertext transfer protocol -- HTTP/1.0. RFC 1945, Internet Engineering Task Force.

Brittle, J. and Boldyreff, C. 2003. Self-organizing maps applied in visualising large software collections. *Proceedings of the Second International Workshop on Visualizing Software for Understanding and Analysis*.

Brown, R. and Pham, B. 2005. Image mining and retrieval using hierarchical support vector machines. *Proceedings of the Eleventh International Multimedia Modelling Conference*, pp. 446-451.

Brown, R., Pham, B. and de Vel, O. 2003. A grammar for the specification of forensic image mining searches. *Proceedings of the Eighth Australian and New Zealand Intelligent Information Systems Conference*, pp. 139-144.

Brown, R., Pham, B. and de Vel, O. 2005. Design of a digital forensics image mining system. *Proceedings of the International Workshop on Intelligent Information Hiding and Multimedia Signal Processing*.

Buchholz, F. and Falk, C. 2005. Design and implementation of Zeitline: a forensic timeline editor. *Proceedings of the fifth Digital Forensic Research Workshop*.

Buchholz, F. and Spafford, E. 2004. On the role of file system metadata in digital forensics. *Digital Investigation*, vol. 1, no. 4, pp. 298-309.

Caloyannides, M.A. 2001. *Computer forensics and privacy*. Artech House.

Caloyannides, M.A. 2004. *Privacy protection and computer forensics*. Artech House.

Carney, M. and Rogers, M. 2004. The Trojan made me do it: a first step in statistical based computer forensics event reconstruction. *International Journal of Digital Evidence*, vol. 2, no. 4.

Carrier, B. and Spafford, E.H. 2003. Getting physical with the digital investigation process. *International Journal of Digital Evidence*, vol. 2, no. 2.

Carrier, B.D. and Spafford, E.H. 2005. Automated digital evidence target definition using outlier analysis and existing evidence. *Proceedings of the fifth Digital Forensic Research Workshop*.

Casey, E. 2002. *Handbook of computer crime investigation: forensic tools and technology*. Academic Press.

Casey, E. 2004a. *Digital evidence and computer crime: forensic science, computers and the Internet*. Academic Press.

Casey, E. 2004b. Network traffic as a source of evidence: tool strengths, weaknesses, and future needs. *Digital Investigation*, vol. 1, no. 1, pp. 28-43.

CERT Coordination Center (CERT/CC). CERT/CC Statistics 1988-2005. 2006. (http://www.cert.org/stats).

Chen, M., Han, J. and Yu, P.S. 1996. Data mining: an overview from a database perspective. *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 6, pp. 866-883.

Chen, H., Chung, W., Xu, J.J., Wang, G., Qin, Y. and Chau, M. 2004. Crime data mining: a general framework and some examples. *IEEE Computer*, vol. 37, no. 4, pp. 50-56.

Cooley, R., Mobasher, B. and Srivastava, J. 1999. Data preparation for mining World Wide Web browsing patterns. *Knowledge and Information Systems*, vol. 1, no. 1, pp. 5-32.

Corey, V., Peterman, C., Shearin, S., Greenberg, M.S. and Van Bokkelen, J. 2002. Network forensics analysis. *IEEE Internet Computing*, vol. 6, no. 6.

Corney, M., de Vel, O., Anderson, A. and Mohay, G. 2002. Gender-preferential text mining of e-mail discourse. *Proceedings of the Eighteenth Annual Computer Security Applications Conference*, pp. 282-289.

Creative Technology Ltd. 2006. (http://www.creative.com).

Davis, L. 1991. *Handbook of genetic algorithms*. Van Nostrand Reinhold.

Davis, M., Manes, G. and Shenoi, S. 2005. A network-based architecture for storing digital evidence. *Advances in Digital Forensics*, pp. 33-42. Springer.

Deboeck, G.J. 1998. Financial applications of self-organizing maps. *Neural Network World*, vol. 8, no. 2, pp. 213-241.

Deutsch, P. 1996. GZIP file format specification version 4.3. RFC 1952, Internet Engineering Task Force.

de Vel, O., Anderson, A., Corney, M. and Mohay, G. 2001. Mining e-mail content for author identification forensics. *ACM SIGMOD Record*, vol. 30, no. 4, pp. 55-64.

Ding, C., Patra, J.C. and Peng, F.C. 2005. Personalized Web search with self-organizing map. *Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service*, pp. 144-147.

Dix, A.J., Finlay, J.E., Abowd, G.D. and Beale, R. 1998. *Human-computer interaction, second edition*. Prentice Hall.

Drobics, M., Winiwarter, W. and Bodenofer, U. 2000. Interpretation of self-organizing maps with fuzzy rules. *Proceedings of the Twelfth IEEE International Conference on Tools with Artificial Intelligence*, pp. 304-311.

dtSearch Corporation. 2006. (http://www.dtsearch.com).

Duval, T., Jouga, B. and Roger, L. 2005. The Mitnick case: how Bayes could have helped. *Advances in Digital Forensics*, pp. 91-104. Springer.

Eastlake III, D. and Jones, P. 2001. US secure hash algorithm 1 (SHA1). RFC 3174, Internet Engineering Task Force.

Eirinaki, M. and Vazirgiannis, M. 2003. Web mining for Web Personalization. *ACM Transactions on Internet Technology*, vol. 3, no. 1, pp. 1-27.

Engelbrecht, A.P. 2003. *Computational intelligence: an introduction*. Wiley.

Farid, H. and Lyu, S. 2003. Higher-order wavelet statistics and their application to digital forensics. *Proceedings of the IEEE Workshop on Statistical Analysis in Computer Vision*.

Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. 1996. *Advances in knowledge discovery and data mining*. The MIT Press.

Fei, B., Eloff, J., Venter, H. and Olivier, M. 2005a. Exploring forensic data with self-organizing maps. *Advances in Digital Forensics*, pp. 113-123. Springer.

Fei, B.K.L., Eloff, J.H.P., Olivier, M.S., Tillwick, H.M. and Venter, H.S. 2005b. Using self-organising maps for anomalous behaviour detection in a computer forensic investigation. *Proceedings of the Fifth Annual Information Security South Africa Conference*.

Fei, B.K.L., Eloff, J.H.P., Olivier, M.S. and Venter, H.S. 2006a. The use of self-organising maps for anomalous behaviour detection in a digital investigation. To be published in *Forensic Science International*.

Fei, B.K.L., Eloff, J.H.P., Olivier, M.S. and Venter, H.S. 2006b. Analysis of Web proxy logs. Accepted for presentation at the *Second Annual IFIP WG 11.9 International Conference on Digital Forensics*.

Gery, M. and Haddad, H. 2003. Evaluation of Web usage mining approaches for user's next request prediction. *Proceedings of the Workshop on Web Information and Data Management*, pp. 74-81.

Gollmann, D. 1999. *Computer security*. Wiley.

Guidance Software, Inc. 2005. (http://www.guidancesoftware.com).

Han, J. and Kamber, M. 2005. *Data mining: concepts and techniques, second edition*. Morgan Kaufmann.

Himberg, J. 1998. Enhancing SOM-based data visualization by linking different data projections. *Intelligent Data Engineering and Learning*, pp. 427-434. Springer.

Inman, K. and Rudin, N. 2001. *Principles and practice of criminalistics: the profession of forensic science*. CRC Press.

Jacobson, V., Leres, C. and McCanne, S. 2006. tcpdump/libpcap public repository. (http://www.tcpdump.org).

Joachims, T. 2002. *Learning to classify text using support vector machines: methods, theory and algorithms*. Kluwer Academic Publishers.

Karypis, G., Han, E. and Kumar, V. 1999. Chameleon: hierarchical clustering using dynamic modelling. *IEEE Computer*, vol. 32, no. 8, pp. 68-75.

Kaski, S. 1997. Data exploration using self-organizing maps. PhD thesis, Helsinki University of Technology, Finland.

Kaski, S., Kangas, J. and Kohonen, T. 1998. Bibliography of self-organizing map (SOM) papers: 1981-1997. *Neural Computing Surveys*, vol. 1, pp. 102-305.

Kenneally, E.E. and Brown, C.L.T. 2005. Risk sensitive digital evidence collection. *Digital Investigation*, vol. 2, no. 2, pp. 101-119.

Keshav, S. and Sharma, R. 1998. Issues and trends in router design. *IEEE Communications Magazine*, vol. 36, no. 5, pp. 144-151.

Kohonen, T. 1981. Automatic formation of topological maps of patterns in a self-organizing system. *Proceedings of the Second Scandinavian Conference on Image Analysis*, pp. 214-220.

Kohonen, T. 1982. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, vol. 43, no. 1, pp. 59-69. Springer.

Kohonen, T. 1990. The self-organizing map. *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464-1480.

Kohonen, T., Kaski, S., Lagus, K., Salojarvi, J., Honkela, J., Paatero, V. and Saarela, A. 2000. Self organization of a massive document collection. *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 574-585.

Kohonen, T. 2001. *Self-organizing maps*. Springer.

Kolari, P. and Joshi, A. 2004. Web mining: research and practice. *IEEE Computing in Science and Engineering*, vol. 6, no. 4, pp. 49-53.

Kosala, R. and Blockeel, H. 2000. Web mining research: a survey. *SIGKDD Explorations*, vol. 2, no. 1, pp. 1-15.

Kruse II, W.G. and Heiser, J.G. 2002. *Computer forensics: incident response essentials*. Addison-Wesley.

Lakkaraju, K., Yurcik, W. and Lee, A.J. 2004. NVisionIP: netflow visualizations of system state for security situational awareness. *Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security*, pp. 65-72.

Li, Y., Chen, X. and Yang, B. 2002. Research on Web mining-based intelligent search engine. *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 1, pp. 386-390.

Liu, W., Duan, H., Ren, P., Li, X. and Wu, J. 2003. Wavelet based data mining and querying in network security databases. *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 1, pp. 178-182.

Lu, H., Setiono, R. and Liu, H. 1996. Effective data mining using neural networks. *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 6, pp. 957-961.

Lyle, J.R. 2003. NIST CFTT: testing disk imaging tools. *International Journal of Digital Evidence*, vol. 1, no. 4.

Maltzahn, C. and Richardson, K.J. 1997. Performance issues of enterprise level Web proxies. *Proceedings of the ACM Sigmetrics International Conference on Measurement and Modeling of Computer Systems.*

Marcella, A.J. and Greenfield, R.S. 2002. *Cyber forensics: a field manual for collecting, examining and preserving evidence of computer crimes.* Auerbach.

Marsico, C.V. 2005. Digital music device forensics. Technical Report 2005-27, Center for Education and Research in Information Assurance and Security, Purdue University, Indiana.

Marsico, C.V. and Rogers, M.K. 2005. iPod forensics. *International Journal of Digital Evidence*, vol. 4, no. 2.

Mahoney, M.V. and Chan, P.K. 2002. Learning nonstationary models of normal network traffic for detecting novel attacks. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 376-385.

Mellars, B. 2004. Forensic examination of mobile phones. *Digital Investigation*, vol. 1, no. 4, pp. 266-272.

Mena, J. 2003. *Investigative data mining for security and criminal detection*. Butterworth Heinemann.

Merkl, D. and Rauber, A. 1997. Alternative ways for cluster visualization in self-organizing maps. Proceedings of the Workshop on Self-Organizing Maps, pp. 106-111.

Meyers, M. and Rogers, M. 2004. Computer forensics: the need for standardization and certification. *International Journal of Digital Evidence*, vol. 3, no. 2.

Microsoft Corporation. 2006. (http://www.microsoft.com).

Middleton, B. 2001. *Cyber crime investigator's field guide*. CRC Press.

Mobasher, B., Jain, N., Han, E. and Srivastava, J. 1996. Web mining: pattern discovery from World Wide Web transactions. Technical Report 96-050, University of Minnesota, Minnesota.

Mobasher, B., Cooley, R. and Srivastava, J. 2000. Automatic personalization based on Web usage mining. *Communications of the ACM*, vol. 43, no. 8, pp. 142-151.

Mohay, G. 2005. Technical challenges and directions for digital forensics. *Proceedings of the First International Workshop on Systematic Approaches to Digital Forensic Engineering*, pp. 155-161.

Morris, R. 2002. Options in computer forensic tools. *Computer Fraud and Security*, vol. 2002, no. 11, pp. 8-11.

Motion, P. 2005. Hidden evidence. *The Journal of the Law Society of Scotland*, vol. 50, no. 2, pp. 32-34.

Mukkamala, S. and Sung, A.H. 2002. Feature ranking and selection for intrusion detection systems using support vector machines. *Proceedings of the Second Digital Forensic Research Workshop*.

Mukkamala, S. and Sung, A.H. 2003. Identifying significant features for network forensic analysis using artificial techniques. *International Journal of Digital Evidence*, vol. 1, no. 4.

National Institute of Justice. 2004. Forensic examination of digital evidence: a guide for law enforcement. (http://www.ojp.usdoj.gov/nij).

National Institute of Standards and Technology (NIST). 2002. Secure hash standard. Federal Information Processing Standards Publication 180-2.

National Institute of Standards and Technology (NIST). 2005. Computer Forensics Tool Testing (CFTT) project. (http://www.cftt.nist.gov).

Noblett, M.G., Pollitt, M.M. and Presley, L.A. 2000. Recovering and examining computer forensic evidence. *Forensic Science Communications*, vol. 2, no. 4.

Oatley, G.C. and Ewart, B.W. 2003. Crimes analysis software: 'pins in maps', clustering and Bayes net prediction. *Expert Systems with Applications*, vol. 25, no. 4, pp. 569-588.

Oja, M., Kaski, S. and Kohonen, T. 2002. Bibliography of self-organizing map (SOM) papers: 1998-2001 addendum. *Neural Computing Surveys*, vol. 3, pp. 1-156.

Osei-Bryson, K. 2004. Evaluation of decision trees: a multi-criteria approach. *Computers and Operations Research*, vol. 31, no. 11, pp. 1933-1945.

Palmerini, P. 2004. On performance of data mining: from algorithms to management systems for data exploration. PhD thesis, University of Venice, Italy.

Payer, U., Teufl, P. and Lamberger, M. 2005. Traffic classification using self-organizing maps. *Proceedings of the Fifth International Network Conference*, pp. 11-18.

Perner, P. 2006. Recent advances in data mining. *Engineering Applications of Artificial Intelligence*, vol. 19, no. 4, pp. 361-362.

Pernkopf, F. 2005. Bayesian network classifiers versus selective k-NN classifier. *Pattern Recognition*, vol. 38, no. 1, pp. 1-10.

Petersen, J.P. 2005. Forensic examination of log files. MSc thesis, Informatics and Mathematical Modelling, Technical University of Denmark, Denmark.

Piatetsky-Shapiro, G. 1999. The data mining industry coming of age. *IEEE Intelligent Systems*, vol. 14, no. 6, pp. 32-34.

Pollitt, M.M. 2001. Report on digital evidence. *Proceedings of the Thirteenth International Forensic Science Symposium*.

Prosise, C., Mandia, K. and Pepe, M. 2003. *Incident response and computer forensics, second edition*. McGraw-Hill.

Puketza, N.J., Zhang, K., Chung, M., Mukherjee, B. and Olsson, R.A. 1996. A methodology for testing intrusion detection systems. *IEEE Transactions on Software Engineering*, vol. 22, no. 10, pp. 719-729.

Reith, M., Carr, C. and Gunsch, G. 2002. An examination of digital forensic models. *International Journal of Digital Evidence*, vol. 1, no. 3.

Rivest, R. 1992. The MD5 message-digest algorithm. RFC 1321, Internet Engineering Task Force.

Roussev, V. and Richard III, G.G. 2004. Breaking the performance wall: the case for distributed digital forensics. *Proceedings of the fourth Digital Forensic Research Workshop*.

Rude, T. 2002. Independent validation and verification of Storage Media Archival Recovery Toolkit (SMART). Red Hat, Inc.

Sammes, T. and Jenkinson, B. 2000. *Forensic computing: a practitioner's guide*. Springer.

Schweitzer, D. 2003. *Incident response: computer forensics toolkit*. Wiley.

Scott, M. 2003. Independent review of common forensic imaging tools. Memphis Technology Group, LLC.

Slay, J. and Jorgensen, K. 2005. Applying filter clusters to reduce search state space. *Advances in Digital Forensics*, pp. 295-301. Springer.

Smith, K.A. and Ng, A. 2003. Web page clustering using a self-organizing map of user navigation patterns. *Decision Support Systems*, vol. 35, no. 2, pp. 245-256.

Sommer, P. 1999. Intrusion detection systems as evidence. *Computer Networks*, vol. 31, no. 23-24, pp. 2477-2487.

Srivastava, J., Cooley, R., Deshpande, M. and Tan, P. 2000. Web usage mining: discovery and applications of usage patterns from Web data. *SIGKDD Explorations*, vol. 1, no. 2, pp. 12-23.

Stephenson, P. 2003. A comprehensive approach to digital incident investigation. *Information Security Technical Report*, vol. 8, no. 2, pp. 42-54.

Takada, T. and Koike, H. 2002a. MieLog: a highly interactive visual log browser using information visualization and statistical analysis. *Proceedings of the Sixteenth USENIX Large Installation System Administration Conference*, pp. 133-144.

Takada, T. and Koike, H. 2002b. Tudumi: information visualization system for monitoring and auditing computer logs. *Proceedings of the Sixth International Conference on Information Visualization*, pp. 570-576.

Tangsripairoj, S. and Samadzadeh, M.H. 2004. Application of self-organizing maps to software repositories in reuse-based software development. *Proceedings of the International Conference on Software Engineering Research and Practice*, vol. 2, pp. 741-747.

Thompson, E. 2005. MD5 collisions and the impact on computer forensics. *Digital Investigation*, vol. 2, no. 1, pp. 36-40.

Ultsh, A. 1993. Self-organizing neural networks for visualization and classification. *Information and classification: concepts, methods and applications*. Springer.

Vacca, J.R. 2002. *Computer forensic: computer crime scene investigation*. Charles River Media.

Vaughan, C. 2004. Xbox security issues and forensic recovery methodology (utilising Linux). *Digital Investigation*, vol. 1, no. 3, pp. 165-172.

Vesanto, J. 1997. Data mining techniques based on the self-organizing map. MSc thesis, Helsinki University of Technology, Finland.

Vesanto, J. 2000. Using SOM in data mining. Licentiate thesis, Helsinki University of Technology, Finland.

Vesanto, J., Himberg, J., Alhoniemi, E. and Parhankangas, J. 2000. SOM Toolbox for Matlab 5. Report A57, Helsinki University of Technology, Finland.

Vesanto, J. 2002. Data exploration process based on the self-organizing map. PhD thesis, Helsinki University of Technology, Finland.

Wang, W. and Daniels, T.E. 2005. Network forensics analysis with evidence graphs: demo proposal. *Proceedings of the Fifth Digital Forensic Research Workshop*.

Wang, X., Abraham, A. and Smith, K.A. 2005a. Intelligent Web traffic mining and analysis. *Journal of Network and Computer Applications*, vol. 28, no. 2, pp. 147-165.

Wang, Y., Cannady, J. and Rosenbluth, J. 2005b. Foundations of computer forensics: a technology for the fight against computer crime. *Computer Law and Security Report*, vol. 21, no. 2, pp. 119-127.

Wessels, D. 2005. Squid Web proxy cache. (http://www.squid-cache.org).

Willassen, S.Y. 2003. Forensics and the GSM mobile telephone system. *International Journal of Digital Evidence*, vol. 2, no. 1.

Willassen, S. 2005. Forensic analysis of mobile phone internal memory. *Advances in Digital Forensics*, pp. 191-204. Springer.

Witten, I.H. and Frank, E. 2005. *Data mining: practical machine learning tools and techniques, second edition*. Morgan Kaufmann.

Xue, Y. and Brown, D.E. 2006. Spatial analysis with preference specification of latent decision makers for criminal event prediction. *Decision Support Systems*, vol. 41, no. 3, pp. 560-573.

Yang, C.C., Chen, H. and Hong, K. 2003. Visualization of large category map for Internet browsing. *Decision Support Systems*, vol. 35, no. 1, pp. 89-102.

Zaiane, O.R. 1999. Resource and knowledge discovery from the Internet and multimedia repositories. PhD thesis, Simon Fraser University, Canada.