Safer Live Forensic Acquisition

Ryan Jones
Computer Science Laboratory
University of Kent at Canterbury
United Kingdom
ryan@ryanjones.eu

Abstract

Computer forensic practitioners must gather, analyse and present evidence at court. A method of data gathering called live forensic acquisition is often used. This method involves gathering data from a system without shutting it down and allows forensic practitioners to access a large quantity of valuable information. However, as data on the computer is constantly changing the copy of data taken is slurred. Slurred data is difficult to analyse as different pieces of the data show the system at different times and these times are unknown. This project identifies the weaknesses with the current live forensic methodology and provides an improved method which produces perfect, unslurred, images. This greatly improves the analytical value and evidential integrity of the data.

1. Introduction

This project will explore the current methods of computer forensic acquisition, identifying and explaining their shortcomings. Possible improvements to the methodology will be discussed and one of these will be developed upon and specified in detail. Proof of Concept (PoC) code to demonstrate the improved method will be produced. The project will analyse possible uses and benefits of the new method and consider the potential impact on computer forensics. Finally, we will look at future research that could build upon the findings of this project.

1.1. Appeal and Importance

Computer forensics is an exciting and challenging field. Like most of computer science its landscape is constantly changing. Computer forensics is constrained by technical issues but additionally, is bound by the laws within which practitioners must operate. Computer forensics is also a practical field; used across the globe to enable prosecutions to be mounted on a wide range of criminals. This topic impacts upon many fields outside of computer science, detailed later in this chapter.

Computer forensics must evolve rapidly to counter the techniques used by criminals to impede forensic practitioners. There are very few areas of computer science that involve this interesting and challenging ‘arms race’. Criminals are constantly pushing the boundaries of technology both in terms of what they can do and how they hide their tracks. Consequently techniques can rapidly become ineffective against criminals using certain methods. The increasing computing power available exacerbates this problem as techniques, previously impractical, become available to criminals.

Difficulties are introduced due to the unique practical and legal constraints. Legal constraints restrict the methods in which data can be obtained. For instance data acquired in certain ways can only be used for intelligence and not as evidence. A possible practical constraint could be: A computer in an internet café has been used by a criminal and needs to be imaged. The owners of the café are known to be hostile to the police so the forensic gathering of data must be covert. How will this be done?

Computer crime in the UK is on the increase and along with it the number of computer forensics professionals. Due to this growth, specific government agencies have been set up to tackle hi-tech crime. As computer use increases and society relies more upon computers, the importance of computer forensics will increase. It is important to note that computer forensics is not just useful against cyber-criminals but also against any criminal who leaves digital traces of crime. Criminals ranging from burglars to kidnappers leave digital footprints in the wake of their criminal dealings.

1.2. Disciplines Impacted by Computer Forensics

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Law</td>
<td>Computer forensic practitioners are consulted to determine changes in the law which could aid investigations. An ongoing example of this is Part III of the Regulation of Investigative Powers Act (RIPA). This makes the act of not proving a known cryptographic key to the police when requested a criminal offence. RIPA Part III was formulated in direct response to criminals’ use of encrypted files to hamper forensic</td>
</tr>
</tbody>
</table>
A large proportion of the work of a forensic practitioner is the analysis of traces left behind by software. Software developed by computer scientists may have an impact on the analytical value and evidential integrity of that data.

The goals and restrictions placed upon evidence are the same for computer forensics as forensics in general. When computer forensic practitioners work with forensic scientists they must be aware of the effects of the other's actions. For example the techniques used to lift fingerprints from surfaces can damage computer hardware.

Any improvements in computer forensics give investigators greater flexibility to conduct enquiries.

Computer networks and systems can be configured with consideration for forensic readiness. Effective preparation, such as logging maximises the chances of a successful prosecution.

First responders to security incidents are often system administrators. If they act in a forensically safe manner they are able to collect evidence for possible future prosecutions.

Advances in computer forensics which reduce the disruption caused by an investigation are advantageous.

1.3. Project Objectives

This project investigates the current state of computer forensic acquisition. It analyses the shortfalls of the current live forensic acquisition methodology followed by an investigation of possible improvements. The project then provides PoC code implementing one of the improved methods. The improved method is evaluated and suggestions for further research are provided.

1.4. Research areas and methodology

This paper provides a general overview of computer forensic methods and practices today. The research will focus on the current acquisition methodology and its shortfalls. The sources used include: journal articles, research papers, eminent scholars in the field, books and online articles.

1.5. Summary

The introduction has justified the interest and relevance of computer forensics. It has detailed what has been researched along with how and where the research data was obtained. The ultimate goals for the dissertation have been stated – in the project objectives – allowing these to be compared to the research outcomes.

2. Background

This chapter provides the context of the research and defines the terminology used throughout. The four stages of the traditional, dead forensic acquisition method will be detailed. The weaknesses of the dead methodology will be identified, demonstrating the need for an additional method to be developed. The additional methodology, developed in response to the limitations of the dead method, is live forensic acquisition. This methodology will be explained and contrasted against the dead method.

2.1. Definitions

For the purposes of this project any reference to non-physical hard disks also refers to partitions, USB/Firewire mass storage devices, RAM disks, mounted encrypted volumes or any other device which presents a hard disk like interface.

The Oxford English Dictionary defines forensic as: “A. adj. Pertaining to, connected with, or used in courts of law; suitable or analogous to pleadings in court.” For the purposes of this paper forensic will be defined as follows: “for use in courts of law”.

I have defined computer forensics as “the collection and analysis of data from computer systems, networks, communication streams (wireless) and storage media in a manner that is admissible in a court of law. It is a merger of the disciplines of computer science and the law.” This is the definition that is appropriate here.

Computer forensic acquisition will be defined as “the process of copying data from a computer in a forensic manner.” This does not involve any interpretation of data, only the act of copying data to a separate device where it can be safely analysed.

Imagers are hardware or software which implement computer forensic acquisition.

2.2. Computer Forensic Investigation Process

The computer forensic process is made up of four distinct stages: Collection, Examination, Analysis and Reporting. The collection stage involves the location, seizure and acquisition of data in a forensic manner. The examination utilises both manual and computerised techniques to identify and extract data relevant to the case. Analysis is the process of using the relevant data to prove which actions have been
performed utilising the computer's resources. Finally reporting presents the information gathered by the forensic examiner, this will usually take the form of a written report. The remainder of this dissertation investigates the acquisition portion of the collection stage.

2.3. Dead forensic acquisition

The most reliable and predictable method of acquisition is performed on computers that have been powered off. It is commonly known as 'traditional' or 'dead' acquisition. Dead acquisition is a very simple process:

2.4. Simplicity, Reliability and Thoroughness

Dead acquisition has clear merits principally due to its simplicity; its main strength is the clearly defined and straightforward stages of the acquisition, which can be verified at any time. The logic required for the acquisition software is extremely straightforward making it easily understood. There is no risk of altering and therefore contaminating the evidence as the data on the hard disk is not modified. It is also very reliable due to its clear stages and the predictability of hard disks.

This method is not simply copying files; file copying does not collect unallocated sectors, slack space† and complete file metadata. Dead acquisition is extremely thorough; every byte on the hard disk is acquired including unallocated, slack space and metadata.

2.5. Weaknesses of Dead Acquisition

As we have seen, the dead methodology acquires a complete copy of all data on a hard disk. While this is useful, there is much information present on a computer that is not on the hard disk.† Slack space: The spare, unused space in a cluster. This space may contain valuable remnants of past, deleted files. Criminals rapidly responded to the successes of computer forensic practitioners. For example, many criminals began to use encryption extensively.† Clearly, having an exact copy of an encrypted file is no use to a forensic examiner as analysis of seemingly random data is impossible.

Encrypted volumes, files which contain an encrypted file system, are widely used by criminals. These files are opened with a program, such as BestCrypt, which can decrypt the data and mount the file system. Once a key has been provided to the program the file system can be mounted and accessed like any other file system; encryption and decryption are transparent. Encrypted volumes are useless if a forensic practitioner can assess a computer while the volume is still mounted.

Also, as networks became more widespread, the importance of network data - such as currently open ports - grew dramatically. As this data is volatile, turning off the computer automatically causes the data to be lost. Other important data, such as decryption keys for encrypted files, can also be stored in volatile memory.

2.6. The Response: Live Forensics

To combat the problems of how dead acquisition is ineffective against encryption and loss of volatile data, the live forensic methodology was developed. This allowed computer forensic practitioners to run programs on suspect's computers to acquire RAM, unencrypted files and any other data they saw fit. There are guidelines concerning what can be done but ultimately the forensic practitioner can perform any action that can be justified in court. The forensic practitioner must stay within the limits of search warrants and keep modified data to a minimum. Live forensics clearly solves the problems detailed above as now we are able to acquire unencrypted data, if present, and also contents of RAM. Performing a full dead acquisition of the computer is recommended, if it is possible to do so after live acquisition.

When it is deemed appropriate to perform live imaging of hard disks the method used is similar to dead acquisition. A program is run in user mode on the target system which reads all sectors sequentially, storing the data from each sector on forensically sound media. The data is commonly sent over a network to a forensically sound machine and stored to ensure minimum modification of the target computer. There are other options for data storage such as forensically sound external USB hard drives.

As every computer installation is different, this process is extremely difficult to describe. The guidelines published by the Association of Chief Police Officers (ACPO) offer the following advice to forensic practitioners:

“In exceptional circumstances, where a person finds
it necessary to access original data held on a computer or storage media, that person must be competent to do so and be able to give evidence explaining the relevance and implications of their actions.”

As live forensics can be used in a wide variety of situations, training courses cannot attempt to cover all eventualities. Forensic examiners are taught the principles of live forensics and the implications of certain actions. However in the field, forensic examiners must rely on their knowledge and experience to make decisions.

Forensic examiners enter a scene without knowing the exact computer equipment that they will find. Even experienced computer forensic professionals can never know the best way to deal with every eventuality. This inevitably means sometimes with hindsight poor decisions are made about the way to progress with a live acquisition, leading to the possibility of rendering data inadmissible as evidence. The availability of a tool to reduce the risk in retrieving corrupt or inadmissible data would prove very valuable.

2.7. Summary

This chapter has detailed the two methods of forensic acquisition in present usage. The shortfalls of the dead acquisition method have been highlighted. These show the necessity of the development of live acquisition. The next chapter will explain in detail the current problems with live acquisition methodology.

3. Problems with Live Forensics

This chapter will explain the two main technical problems with live acquisition and the difficulties they cause.

3.1. Data modification during acquisition process.

Data on a computer may be modified by any process during acquisition. These can range from user applications to server applications or the operating system itself. Forensic examiners have no control over the types and numbers of processes running when they begin to investigate a computer. Every running process is liable to change data on the drive during acquisition.

3.1.1. Slurred images. ‘Slurred’ images are produced when the file system being acquired is modified during acquisition. Any modification causes a problem as the meta-data section of the hard disk - the record of where all files on the hard disk are currently situated - (e.g. The MFT or FAT) is read first. If sectors, stated by metadata to hold parts of files, are changed before they have been acquired the analysis becomes problematic as the metadata and sectors don’t correspond.

For example: If metadata indicates MyPasswords.doc is held in sectors 1567 – 1626 but when the imaging software reaches these sectors the file has moved we have potentially lost valuable evidence. The data may still be on the drive; however it becomes extremely difficult to find it.

This process is analogous to hand-copying a book, one page at a time, whilst someone else is still editing it. The contents page has been acquired but by the time the rest of the book has been copied chapters have been moved around and edited. The contents page can no longer be relied upon as an accurate guide to the book.

3.1.2. Potential for hard disk modification by forensic practitioners. To perform live acquisition forensic practitioners must execute code which will run on the CPU of the suspect system. The code will change data in the CPU registers and RAM. It may also change data on the hard disk. This is the case even if there are no explicit write commands. This is because, as the acquisition programs run in a virtual machine, the operating system may decide to swap the program to hard disk. This complicates the displaying of evidence in a case and gives the defence the opportunity to argue that evidence should be ruled inadmissible. This often happens in practice as the evidence against someone is so overwhelming the only possible line of defence is to argue the evidence inadmissible.

3.1.3. We may ruin all evidence when inappropriate action is taken by forensic examiners. Errors in a forensic examination can cause an unnecessary amount of data to be changed. This may be due to something as simple as running an application on the suspect hard drive. Running a program commonly causes it to overwrite much data, such as last opened time and other lists of recent actions. If the criminal use of this application is pivotal to the case it will cause many issues in court. If there is a large amount of anomalous data in images it may call in to question the competence of the forensic examiner and the reliability of the evidence in general. In this case the data gained by live forensic acquisition might be less useful than if data had been gained by dead acquisition.

3.1.4. Anti-forensic programs. Criminals who are forensically aware are liable to take steps to reduce the effectiveness of a potential investigation. Writing a program to detect forensic acquisition programs and subsequently to destroy evidence is not complicated. EnCase FIM publishes the registry keys it changes on installation. This makes it extremely easy to identify the program and allow evidence destroying to commence.

3.2. Acquisition dependant on operating system

Live forensic acquisitions are performed by programs running in user space. In other words they run in their
own virtual machine on top of the operating system. Consequently the only way acquisitions can be performed is by requesting information from the operating system via its APIs. The acquisition software must trust the operating system completely to gather its information.

3.2.1. Problems with this dependency. Criminals who suspect they may have forensic techniques used against them may modify the operating system to lie for them. A method to provide programs in user space with deliberately sanitised data is a new idea, however proof of concept code has been produced that enables this. In this case, live imaging has been likened to "turning up to a homicide at the docks and asking the mafia to collect your evidence". This is because we examine the computer (scene of crime) and ask the potentially 'bad' operating system to give us the data on the hard disk.

3.3. Summary

This chapter has identified and explained the main problems with live forensics today. The next chapter will discuss some of the possible methods to tackle these issues along with their merits and shortfalls.

4. Solutions to live forensic problem

Four solutions have been identified to tackle the problem of data changing during acquisition. The advantages and disadvantages of each solution shall be explained and the evaluation criteria shall include the practicality and flexibility of use. One of these solutions will be selected and implemented within the project time-scale.

4.1. Freezing current state of computer

A solution to allow us to image the ram in memory would be to freeze (perhaps crash) the computer while still maintaining power to the RAM. It may then be possible to image the ram by connecting directly to the hardware. This method affords us a complete image of the RAM and hard disk with the knowledge that no modifications can be made during the acquisition. It guarantees us the image produced will not be slurred. To perform this, direct access to the internals of a computer must be gained. It would be a delicate process in which different imaging hardware would be required for each type of RAM. Another major disadvantage with this system is the fact that when we freeze the computer we will probably be unable to resume it without a reboot.

4.2. Killing unnecessary programs

There may be many programs running on the system during acquisition any one of which may alter data on the hard disk or in RAM. One method to deal with this is to kill all the processes which are not vital to the system. This would minimise the possibility of programs causing slurred images during acquisition. It would also minimise the possibility of anti-forensics programs, running in user mode, destroying evidence. There are however some large issues with this method. Identifying the programs vital to the operation of the system is difficult. Killing the wrong processes could cause a system crash. A forensic examiner accidentally crashing a suspect computer system would be looked upon very poorly in court; the jury has an expectation of competency from a computer forensic expert. There would be difficulties implementing this system, using a black-list may leave many problematic processes running that are not on the list, while a white-list would be likely to kill many programs that are vital to the system.

4.3. Swapping hard disk for forensic hardware

It may be possible to place forensic write blocking hardware between the motherboard and the hard disk while it is running, without crashing the system. A solution which builds upon the principles of the shadow drive may be feasible. When new forensic hardware is installed all write commands would be written to the new hard disk attached to the shadow drive, instead of the original suspect hard disk. The suspect hard disk would remain unchanged. This would result in obtaining an exact image of the hard disk at the time the forensic hardware was added to the system. There would be possible issues with the system crashing during the install of the hardware. Direct access to the internals of the system would be needed for this method to be installed.

4.4. Imager with write command policing

The final option would be to use a forensic acquisition program placed between the hard disk driver and all processes which use it. This gives it one large advantage; it may inspect every command that is issued to the hard disk before any data is changed. This would allow the acquisition program to intercept any writes to the hard disk and image sectors before they are allowed to be overwritten. Therefore as the acquisition process occurred in the times that read and write commands were not being serviced we gain a complete image of the hard drive at the exact time that the acquisition software was loaded. This method guarantees a complete, non slurred, image of the hard disk. This solution can also be run remotely. This solution also has the potential to be modified to perform similar actions on RAM. However acquisition programs must delve deep into the kernel which greatly increases the possibility of poor programming causing a system crash. Also as the acquisition program must wait for the data to be sent over the network it may cause hard disk access times to increase. This may be a problem if the acquisition is being performed covertly.
as it may alert the user of the system to the presence of
the imager.

4.5. Implementation choice

This project will implement For this project we will implemen solution 4.4. Solutions 4.1 and 4.3 would require a deep understanding of electronics and direct access to the internals of computers. Internal access to live computers has two drawbacks; the solutions are complex to use and there is no opportunity for covert use. Solution 4.2 would require much research into 'good' and 'bad' processes. It would not be possible to keep a complete and accurate process database. It also has serious potential to crash computers by killing essential processes. Solution 4.4 is clearly the most elegant and flexible solution. It is possible to implement PoC code and its operation is simple and easy to understand. There are opportunities to extend the functionality described to provide further advantages.

4.6. Summary

This chapter has outlined four potential solutions to the current problems in live forensics. It has introduced the pros and cons of each solution, allowing an implementable solution to be selected. The next chapter describes in detail the operations of the improved method.

5. Analogy

This chapter offers an analogy of a system which holds data, potentially constantly modifying, which we wish to gain an image of. Two methods to image the data will be described which map to the behaviour of the current and proposed improved live forensic imager. This chapter explains the proposed technical improvements and their legal relevance.

5.1. Analogous components

The following components are represented in the analogy:

<table>
<thead>
<tr>
<th>Component</th>
<th>Represented by</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer System</td>
<td>Company</td>
</tr>
<tr>
<td>Hard Disk</td>
<td>Safe Boxes</td>
</tr>
<tr>
<td>Hard Disk Sector</td>
<td>One Safe Box</td>
</tr>
<tr>
<td>Hard Disk Interface</td>
<td>Secretary</td>
</tr>
<tr>
<td>User Mode Programs</td>
<td>Employee</td>
</tr>
<tr>
<td>Current Forensic Imager</td>
<td>Police Officer</td>
</tr>
<tr>
<td>Improved Forensic Imager</td>
<td>Undercover Police Officer</td>
</tr>
</tbody>
</table>

5.2. System functioning normally

A company has many safe boxes, each capable of holding a piece of paper. Only one secretary can access boxes directly. The secretary can perform two actions; either retrieve a copy of the paper in a box or, replace the piece of paper in a box and have the previous piece destroyed. The secretary mans a desk at which any person can queue at, to request an action to be performed on a box. When a request is made the secretary goes into the room with the safe boxes and carries out the request.

5.3. Scenario

A copy of the contents of all the boxes at precisely 12:30 are needed. This data is vital to an important case.

5.4. Solution with current imager

At 12:30 a police officer requests a copy of the paper in box 0. When the secretary returns he files the copy and asks for a copy of the paper in box 1. The officer repeats this until he has copies of all the boxes.

This works perfectly unless data is modified during the acquisition process. For example, the officer finishes a request and files the copy. During the filing time, unbeknownst to the officer, an employee has made a 'write request' to a box not yet acquired. When the officer reaches this box the data retrieved is not the data that was present at 12:30. The officer will have no knowledge that the copy he has been given has been modified since 12:30.

Clearly this presents the officer with a problem. He can only conclude that the data he has copied was present at some point between 12:30 and the completion of the acquisition, not that the data was present at 12:30. This weakens the evidence collected, a fact which the defence is likely to exploit.
5.5. Solution with Improved Imager

This method replaces the secretary with an undercover police officer who can now regulate all requests to the boxes. The officer replaces the secretary at 12:30 and, while there is no queue, takes copies of the boxes starting at 0 and finishing when all boxes have been copied.

When employees need to access boxes, the officer pauses his copying and services requests. ‘Read requests’ are serviced as the secretary did but ‘write requests’ are treated differently. The officer replaces contents as normal but takes a copy of the data, already in the box, before it is destroyed.

The operation of the system from the employee’s point of view does not change; this allows the system to be deployed both covertly and without disruption to business. With this technique the police officer knows the data gathered was in the boxes at exactly 12:30, making the results easier to analyse and evidentially stronger in court.

5.6. Summary

This chapter has used an analogy to explain the current and improved live imaging methods, clearly demonstrating the benefits of the improved method. The rest of this research will implement and evaluate the improved methodology.

6. Implementation Plan

Section 4 identified a suitable solution for implementation, along with the analogy presented in section 5 the concepts of the solution have been discussed in depth. This section states the goals of the implementation of this solution and explains the main details of the implementation. It will provide the rational for the programming language selection and other design decisions. A diagrammatic representation of the program's structure will be shown along with the reasons for this choice.

6.1. Goal

1. To show it is possible to create a live acquisition program, capable of capturing an exact image of a hard disk, while allowing the computer to run as normal.
2. To demonstrate how this newly developed method differs from the current method in both implementation and results.

6.2. Improved imager operations

The improved imager is designed to police all access to the hard drive therefore it must be placed between the hard disk and all other programs. When executed, the imager will begin to copy the hard disk data, sector by sector until another program requests access to the hard drive. When this occurs the imaging process will pause and determine whether a read or write command has been issued. If a write command for an unimaged sector is received, the request will be blocked to allow the sector to be acquired. Once the sector has been imaged the write operation will be performed as normal. If a read command, or write command for a previously imaged sector, is received it will be passed through to the Hard Disk driver and serviced normally. This ensures we gain an exact image of the hard disk at the time the acquisition started on the forensic workstation.

6.3. Other essential components

For comparison purposes a forensic imager using the current live forensic methodology is required. To allow the acquisition programs to be run and compared we will produce an environment called the Forensic Comparison Platform (FCP). The FCP is described in the next chapter.

6.4. Programming language decision

Before programming can begin a choice of programming language must be made. Four main considerations influenced the decision.

Concurrency – The components of a computer system operate in parallel. The purpose of PoC code is to show that ideas are implementable; therefore its components must also be programmed concurrently.

File System – A file system is required for structured data. For a programming language to be suitable for this project there must be a file system implemented in the language or the ability for a file system to be created relatively easily.

Graphical Feedback – It must be possible to visualise the effects of the imagers in real-time; mature graphics libraries are important.
Time Constraints – The time available for CO620 project is limited making it unfeasible for a new language to be learnt for this PoC.

For these reasons Java was selected for the development. The Java language supports parallelism through its threads and locks system, along with other methods via libraries. Java has mature graphics capabilities, various implementations of file systems and a new file system could be developed if required. The developer is confident in his ability to successfully create a program of this complexity in Java.

6.6. Parallelism options

The parallelism options considered for this project were either a method based on CSP or a threads and locks approach. Threads and locks are the usual method for implementing parallelism in Java. We shall investigate these first.

6.6.1. Threads and Locks. Threads and locks are the tools offered by Java to implement parallelism. Complex parallel systems can be created with threads and locks, however they prove extremely difficult to use safely. Race hazards and deadlock can all occur easily and are difficult debug. Even Sun Microsystems, the developers of Java, warn against the use of threads:

“extreme caution is warranted when designing and building multi-threaded applications... in almost all cases they also make debugging, testing, and maintenance vastly more difficult and sometimes impossible... we urge you to think twice about using threads in cases where they are not absolutely necessary.”

6.6.2. CSP for JAVA. While multiple Java implementations of CSP are available, we will simply consider JCSP. The JCSP model requires creating processes to be run in parallel, communicating through clearly defined channels. The CSP paradigm allows the safety of parallel programs to be easily determined while still enabling the programmer to use the powerful sequential features of Java.

6.6.3. JCSP chosen. The code we are looking to produce models well to the CSP paradigm of processes communicating over channels. The implementer of the code had previous experience with both methods of parallelism and was more confident with JCSP. It was ultimately chosen due to these issues and the frank warnings from Sun's website.

6.7. Implementation Differences.

The current and improved forensic imagers operate in different parts of a computer system. Neither requires modification of other system components to function. In this implementation the current forensic imager accesses the hard disk, sends information to the forensic server and reports to the visualisation.

The improved imager is placed between the file system and the hard disk so it can also manage modifications requested through the user interface.

Notice, the only difference between these diagrams is the placement of the forensic imagers – the components themselves are unmodified. The blue components are part of the FCP, while the yellow processes are of specific interest for this research.

6.6. Summary

This chapter has detailed how the code will be produced. The major components of the system have been identified and explained. Decisions have been made with respect to the programming language and code structure. The next two chapters discuss implementation issues.

7. Forensic Comparison Platform

The code for the forensic imagers themselves is relatively short; however they must be run in an appropriate environment. This chapter discusses briefly the design and construction of that environment; the Forensic Comparison Platform. Further details of the implementation can be found in Appendix 1

7.1. Main Components

The main components of the FCP are

- Hard Disk: Data is stored here and can be accessed sector by sector.
- File System: Allows structured data to be stored.
- User Interface: Allows user modification of files
- Visualisation: Provides display of the systems operations.
- Forensic server: Collects and stores information sent to it by forensic imagers
7.2. Hard Disk

Clearly, the hard disk is vital to the FCP as without it there is no data to image. The virtual hard disk is implemented by providing a hard disk style interface to a 32MB file accessed by the RandomAccessFile class. The hard disk services commands to read and write sectors at any place on the drive. It does not provide any performance enhancing features such as caching.

7.3. File System

The file system allows the FCP to handle structured data; essential to demonstrate imager caused data corruption. It communicates with both the UI and processes with a Hard Disk style interface. The file system is very limited but features include:

- Reading and Writing of files
- Fragmented file support
- Modified, Accessed and Created times

7.4. User Interface

A basic textual user interface (UI) has been implemented to allow interactive data modification and retrieval. This allows the user to modify the hard disk and, with the aid of the visualisation see the affects modifications have with different forensic acquisition programs.

7.5. Visualisation

The visualisation is required to display the affects of hard disk modifications upon different imaging methods. This component of the FCP is not needed for the correct operation of the forensic imagers, only as an aid to understanding. Visualisations would not be present in any real world solution and as it is implemented as a component of the imagers, it is not included on conceptual diagrams. The visualisation shows a view of every sector of the hard disk; white areas are unimaged sectors, green – correctly imaged sectors and black – sectors modified before imaging.

7.6. Forensic Server

The forensic server must receive data from forensic imagers and store the information in an image file. The messages it will receive from the imagers are of the form “Sector X currently holds the data Y”

7.6. Summary

This chapter has briefly discussed the four main components of the FCP, allowing the forensic imagers to be placed into an environment which both correctly mimics a real system and also provides visualisation capabilities.

8. Forensic imagers

In previous chapters we have discussed the FCP and the methods of current and improved live acquisition. We now move on to the implementation of the current and improved imager processes. The programming of both imagers will be discussed along with details of where the imagers are placed in the FCP.

8.1. Conceptual process networks

The current forensic imager operated in user space. It read the hard disk directly but not exclusively – other programs can also access and modify the hard disk. It can effectively image the hard disk iff no unimaged data is modified during the acquisition.
In contrast the improved imager is placed between the file system and the hard disk. No programs in user space can access the hard disk without going through the imager process, allowing it to police all reads and writes of the disk.

8.2. Current forensic imager

The current forensic imager runs in user mode, alongside the UI. It is connected to the hard disk and the forensic server. Its operation is as described in section 5: Read all sectors sequentially, sending sector data across the network to the forensic server.

8.3. Improved forensic imager

As with the current forensic imager, this imager connects directly to the hard disk. However the improved imager does not sit in user mode, it is placed in the kernel, between the hard disk and the rest of the system. Unlike the current acquisition method this imager must not just communicate with the hard disk and forensic server but also with the file system and other processes. The operation of this imager is as described in section 5: Image as normal until a request for hard disk access received. If a write command for an unimaged sector is received image the sector before overwriting the data, otherwise carry out the request as normal.

8.4. Conclusion

The code described in this section, with the assistance of the FCP, successfully implements the current and improved live forensic methods. This has been confirmed by imaging virtual hard disks and using a cryptographic hash function to compare the original file to the results. The current and improved forensic imagers operate without the need to modify any other processes.

8.5. Summary

This chapter has described the successful implementation of the current and improved forensic imagers. It should be noted that none of the FCP components were modified.

9. Conclusions

This chapter will investigate the PoC code and evaluate its success. The various benefits of the improved live acquisition method will be discussed and looking to the future, areas of further research have been identified. Finally a summary of the research is provided.

9.1. Success of code

The PoC code produced for this project successfully implements both the current and improved forensic imagers. This has been confirmed with both comparisons of entire images and extensive use of industry standard cryptographic hashing. The improved method has retained the ability to be used covertly. The visualisation clearly shows the deficiencies of the current imager and effectiveness of the improved method. The improved method provides a range of benefits which will be discussed later but importantly, it undeniably produces clearer, unslurred, easier to analyse images.

9.2. Time Constraints

With more time it would have been beneficial to improve the PoC code to enable greater user interaction with the imaging process. Functionality to control imaging speed, allow pausing and to ‘rewind’ the visualisation would allow easier demonstration. Documentation and code readability could also be improved.

All the areas of further research, detailed later, tackle useful and interesting topics; it is therefore disappointing that time does not allow these to be investigated.

9.3. Benefits

9.3.1. Improves evidential clarity. Images produced with the new method are never slurred. There are two important benefits which come from this. Firstly a hard disk acquired with the improved method produces a snapshot of the system at an exact time. This is a far clearer concept for a jury to understand than one of a slurred image, aiding understanding and limiting the
ability of the defence to bamboozle the jury.
An unslurred image is easier for a forensic practitioner to analyse, reducing investigation time and improving the clarity of evidence produced from the image.

9.3.2. User mode anti-forensic programs ineffective. Anti forensics programs, running in user mode, may attempt to wipe data from the drive by overwriting sectors. This is clearly a threat to current live forensic imagers. The effectiveness of the improved imager is not affected in any way by these programs; malicious write commands are not sent to the hard disk. Write commands are first passed to the improved imager which is able to image the sector before they are overwritten.
A further benefit of this method is that the anti-forensic program does not know its attempts to destroy evidence are useless. Therefore malicious wiping of data, after the forensic imager has been loaded, is not only in vain but appears to be effective.

9.3.3. Strengthens foundations of forensic evidence. As discussed in section 2.6 the ACPO good practice guidelines13 state the examiner must “be able to give evidence explaining the relevance and implications of their actions”. To do this the forensic examiner must understand the operations and interactions of every program running on the computer. This is an extremely complicated task, impossible to reason about formally, which gets exponentially more difficult as the number of programs increase.
The improved imager, however, aids greatly in this task. From the time the imager is loaded it is the only process which can modify the hard drive allowing the forensic practitioner to explain the actions of one process to cover all hard disk modifications.

9.3.4. Secures integrity of hard disk. Once the improved imager has been loaded it is impossible for modifications of the disk to occur without the sectors being imaged first. No actions performed by a forensic practitioner, after the imager is loaded, can affect the evidential integrity and reliability of the acquired hard disk data.
Practitioners, as long as they are trained in the use of the improved imager, are able to perform any operation on the computer without risking the hard disk data. This reduces the risk of live acquisition in general which aids the decision of whether to perform live or dead acquisition.

9.4. Further Research

9.4.1. Implementation on Windows/Unix/Linux. One obvious area of further research is to investigate the difficulties of implementation on a real operating system. This will enable essential tests to be carried out determining reliability, performance and practicality.

9.4.2. Imaging over time. The improved method is able to obtain a snapshot of the hard disk as it has the ability to detect, intercept and alter all hard disk commands. The current method doesn’t use all the information available to it and research into how best to use this information would be useful. One way of building upon the improved imager would be to allow the collection of, not just a snapshot, but also all subsequent modifications. This would enable investigators to see how the data on a hard disk develops over a period of time.

9.4.3. Improved RAM acquisition. This research does not deal with the imaging of RAM as acquiring RAM, although similar in principle, has a major complication: The acquisition process itself is stored in RAM. This causes great problems when the acquisition program attempts to acquire the RAM in which it is itself situated. Research is vital in this area to ensure and guarantee the program will not deadlock and crash the computer.

9.5. Research summary
This paper has analysed the current method of live forensic acquisition and identified shortfalls. Four solutions to the shortfalls are described. The ideas behind one solution have been developed and PoC code has been produced to display its operation. The benefits of this method have been detailed and it clearly has potential importance for the computer forensic community. The topics of further research demonstrate there is work to be done before this method becomes an implementable solution.

Nevertheless, this research paves the way for a new, safer, generation of live forensic imagers which can only improve the integrity of evidence available to courts.

_________________________
[1] Postnote, October 2006, Number 271, Parliamentary Office of Science and Technology
Response, Karen Kent, Suzanne Chevalier, Tim Grance, Hung Dang, Special Publication 800-86, August 2006, Recommendations of the National Institute of Standards and Technology


