FORENSIC SYSTEMS ENGINEERING

Dr. B. B. Jayasingh
CVR College of Engineering, Department of IT, Ibrahimpatan, R.R.District, A.P., India.
Email: bbjayasingh9@rediffmail.com

***Abstract*-Forensic Engineering, to the average engineer, would mean the activity of the Expert Witness who investigates engineering matters involved in legal proceedings. Legal proceedings, although served by forensic engineering, can make only a limited contribution to engineering safety. It is therefore crucial that forensic engineers promote engineering safety by finding ways to share lessons learnt from failures with the engineering community whilst at the same time serving the purposes of the courts. The paper presents the meaning to encompass the investigation of all computer engineering failures; not just restricted to those ending up in Court. Educating for the future is the title of the paper, where advice is offered as to how the engineering professions should promulgate the knowledge gained from the investigation of structural failures.**

***Index Terms*—Forensics, Forensic Computing, Forensic Systems Engineering, computer Forensics, Cyber Forensics, Software Forensics.**

I. INRODUCTION

Twenty-first century engineers, driven by sustainability and technology, are pushing sciences to new limits by creating leaner structures with modern materials using state-of-the-art design and novel construction techniques. They operate in a changing global climate of increased intensity, natural hazards and manmade disasters. Managing and mitigating higher risks may be considered a challenge in engineering new structures and also a threat in assessing the vulnerability of existing infrastructure, especially in the underdeveloped world; a timely launch of Forensic Engineering.

Legal proceedings, although served by forensic engineering, can make only a limited contribution to engineering safety. It is therefore crucial that forensic engineers promote engineering safety by finding ways to share lessons learnt from failures with the engineering community whilst at the same time serving the purposes of the courts. In systems engineering the technical approach is known as a 'hard' system (e.g. structures), and the managerial approach is a 'soft' system (e.g. people). Although much of the focus of forensic engineering is on hard system failure, a forensic 'systems' engineer should facilitate integration of both approaches by seeking to understand and tackle any sources of complexity. For example, a forensic engineer needs to appreciate that the hard system failure is embedded in a soft system failure and cooperate in the investigation of the soft system failure.

Forensic systems engineering [1] is the discipline investigating the history of Information Technology failures. It therefore focuses on the post-mortem analysis and study of project disasters. The work involves a detailed investigation of the project, the environment, decisions taken, politics, human errors and the relationship between subsystems. The work draws upon a multidisciplinary body of knowledge and assesses the project from several directions and viewpoints. The concept of systems is a central tool for understanding the delicate relationships and their implications in the overall project environment.

Forensics highlights the central role of risk management and decision making, leading to **a** new perception of their importance in the development of sound and reliable software systems. In the long-run the field of forensics will help in understanding what it is that software engineers do, before one can start learning (or teaching) how to do it better.

Forensic computing (FC) starts with the fact of abuse having occurred and attempts to gather the evidence needed by investigators to identify the culprits. Moreover, FC must be able to deal with the use of the infrastructure by authorized users for unauthorized or illegal activities. It is already well known that most computer-related crime is carried out not by highly skilled external attackers, but by insiders who have easy access to IT systems.

The primary value of the study of failures, is in feeding knowledge back into the engineers. Detailed analysis of failures pinpoints areas for future research, which are essential rather than accidental, as they result from observed shortcomings of contemporary approaches. We observed the newly developed technology of forensic science has given rise to digital forensic. Digital forensics is categorized into three parts such as computer forensics, cyber forensics and software forensics. Computer forensics is the collection, analysis, examination and presentation of information held in or retrieved from computer hard disks in such a way that it can be used as potential legal evidence. It

 DOI: 10.32377/cvrjst0115

deals with the stand alone computer related crimes. To deal with network/internet related crimes there is a different forensics method referred as cyber forensics [2]. Finding the author of a program code is called software forensics. Thus forensic science encompasses the principles and techniques that help identify evidence at a crime scene.

II. COMPUTER FORENSIC

Computer forensics deals with extracting evidence from the computer itself, or the field of extracting hidden or deleted information from the computer disk is called computer forensics. Computer forensics is that branch of forensic science, which is harnessed to identify, locate, preserve, and extract digital information from a computer system to produce clinching evidence of a crime in the court of law, in an effective manner.

Computer forensics is the study of computer security breaches and their consequences. Computer forensics involves the "preservation, identification, extraction, documentation, and interpretation of computer media for evidentiary and/or root cause analysis [3]. Detection of computer crime and thereafter examination of computer evidence is an emerging field in forensic science today, for which skills need to be developed. Such evidence is required in economic offences, espionage, sabotage, data communication network, terrorism, murder, drugs trafficking, cellular frauds, child pornography etc.

The basic methods of recovering unrecoverable data are described in [4]. The forensic analysis tools are used for recovering hard-disk information. Forensic tools analyze hard disks or hard-disk images from a variety of different operating systems and provide an Explorer-style interface so that one can read the files. The international important forensic tools are here in this paper [5].

The investigator needs to know the rudimentary basics about the computer's hardware and software, operating system, and underlying file system. The professional investigator needs to be comfortable with both Windows and Unix/Linux, including the command line interfaces of both; how each operating system moves, manipulates, and "deletes" files; and how to examine areas of the storage media beyond the file structure, such as unallocated space, file slack, and a host of other areas [3], "The operating system sees all, but it may not tell you about it." The analyst even needs to know how to properly power down and power up a computer, as well as how to disconnect peripherals and network connections, without destroying any of the information on the computer.

Computer evidence is very fragile. Evidence present in a hard disk of a computer can be deleted overwritten or altered in some other manner, unrecoverable or contaminated. Thus, it is essential to isolate a computer involved in a crime as quickly as possible. However, a trained forensic specialist to avoid damage must perform the act of isolating the computer correctly. Unlike the other branches of forensic science, computer forensics did not have time to establish itself as the related technology is changing at a very high speed. But certain procedures and tools have been developed that enable the investigator to analyze the digital evidence.

Consider that many operating systems, such as Linux and Windows 2000, maintain a number of timestamps associated with every file, including the last access date. Using ordinary operating system tools to examine the contents of files will probably cause the last-access date to be changed while specialized analysis tools can examine files without modifying this date. It is important to maintain the integrity of the original data so that you can be sure that the results of the analysis are legally and technically valid.

The heart of the actual forensic analysis, of course, is examining the computer(s) and/or network, recovering all possible information, and reconstructing the activity related to the incident being investigated. One of the most well-known computer forensics tools is the Windows-based analysis software package EnCase, used to perform a thorough analysis of the contents of a system's hard drive. For example, it provides a detailed of the use of EnCase, covering the entire process from media acquisition to analysis to reporting.

There are also tools that end users might employ for defense of their own system, including anti-virus software, IDS, and firewalls. But end users might also deploy tools that make forensics difficult, such as file scrubbers that really do delete files and purge the browser cache, and encryption and steganography software that make the examination of file contents next to impossible without a crypto key. Corporate policies may or may not prohibit use of these tools on a corporate computing resource, but these anti-forensics tools are well known to criminals as well as benign users.

III. CYBER FORENSICS

Cyber Forensics is the branch of digital forensics, which refers to the scientific compilation, examination, exploration and presentation of information held on or retrieved from computer networks in such a way that it can be used as potential legal evidence**.** In other words**,** Cyber Forensics deals with forensic analysis of evidence in computer networks. Computer and Network Forensics (CNF) techniques [6] are used to find out evidence from a variety of computer/network crimes. The ultimate goal of computer and network forensics is to provide sufficient evidence to the law enforcement

agencies where the criminal perpetrator can be prosecuted.

With the rapid expansion of Internet infrastructure, government agencies as well as business organizations of all dimensions are enthusiastic about getting connected to the World Wide Web. The worldwide connectivity has made it possible to provide different kinds of services electronically. This has raised a vital issue of confidentiality and security while transacting over the network. For example, a customer may not like his/her personal details (name, credit card number, job etc.) are disclosed to others, even accidentally when visiting a web server. An effective way of handling such issues is to adopt a mechanism that guarantees secured transaction and at the same time provides all the flexibility. Agent technology is being advocated as a suitable means to fulfill requirements of flexibility, adaptability, autonomy, pro-activeness in such problem areas.

The forensic investigation of cyber crimes involves the identification of the source of communication. First, the person who initiated the communication is to be identified i.e. to trace the communication trail from the victim to the originator. Technically this is a complicated task, as the rapidly changing communication technologies would help the criminal to hide his identity. Such hurdles in the path of the investigating agency encourage the cyber criminals to continue their nefarious designs.

A . Challenges of Cyber Forensics

In a networked environment, the evidence capture and preservation generally occurs after an intrusion or abnormal behavior is detected, so that the abnormal or suspicious activity can be preserved for later analysis [7]. Since relevant information is available in packet headers, try to capture message packets for analysis. In the network environment, there are three kinds of challenges against cyber crimes defined by the law enforcement agencies of the World. Firstly, the technical challenges, which has the ability of law enforcement agencies to find and prosecute criminals operating online environment. Secondly, the legal challenges are due to absence of appropriate laws to combat cyber crimes. Hence the legal laws and legal tools are needed to investigate cyber crimes. Thirdly, The operational challenges, which are the challenges at the ground level. Unlike traditional crimes, the operational canvas of cyber criminals is large both geographically and logically. To successfully detect and gather evidence, a network of well trained, well equipped investigators and prosecutors should work in tandem with great swiftness.

B. Goals of Cyber Forensics

The goal of the cyber forensic includes:

1. The primary goal is to find the evidence against a criminal system in the inter-networked environment, also by assisting the law enforcement agencies in their investigations.
2. The secondary goal of these systems is to reduce investigation time and complexity.

The author [8] provides a broadest coverage of computer and network technology, with special reference to of Windows and Unix systems. In another work, wireless network technologies. [3] have been dealt that provides deeper coverage of computer technology, including the basics of storage media, encryption and steganography, hiding data, and hostile code. The authors also cover Windows and Unix forensics in detail. This provides an excellent introduction to these operating systems for the forensics investigator while assuming no prior knowledge of the operating systems and file systems.[9] also provides detailed coverage of technology that will be of interest to the forensics analyst, covering a long set of tips on how make Windows more secure and private — such as disabling the built-in microphone and not using virtual memory — but doesn't fully explain the underlying rationale for the steps that are recommended. Despite the absence of Unix, it provides detailed and broad coverage of a variety of network and computer technologies. Consistently providing detailed coverage of how data is stored in the memory, registry, and hard drive of computers; modes of data insertion and self-protection, including keystroke logging software, telephone taps, spyware, and even Van Eck radiation; the application and detection of encryption and steganography software; achieving and protecting on-line privacy covering the browser, e-mail, secure protocols, firewalls, and encryption. [10] provide the weakest coverage of technology on Windows discusses how files are stored on the computer with particular emphasis on the Internet Explorer history buffer, cache, and temporary files, the registry, and Event Viewer, while the chapter on Internet abuse primarily describes browsers' cookies, bookmarks, and swap files. A chapter on the tools of the trade covers vulnerability detection tools such as nmap and nessus, protection tools such as BlackICE and swatch, and analysis tools such as The Coroners Toolkit (TCT) and Encase.

C. Policies of Cyber Forensics

It [6] proposes the policies to recover the evidence from computer and network related attacks. In response to any successful attacks the techniques are developed by the try and fix methods of computer and network forensics (CNF). The goal of CNF is to provide sufficient evidence in favor of a criminal to be successfully prosecuted. This paper [6] presents six types of policies in the following categories i.e.

Retaining Information, Planning the Response, Training, Accelerating the Investigation, Preventing Anonymous Activities and protecting the Evidence.

1. The first policy states that to Copy and Retain Application and Local User Files. The illegal copy of user files should not violet the users' privacy that should take care by the company otherwise the evidence may not be admissible to the court. The company has to employ the policy that systematically stores and retains the contents of application and user files as potential legal evidence. As the system logs are vital source of potential evidence the enterprise has to copy and retain computer and network activity logs. The companies that use the network devices like servers and routers have to keep logs of the data packets that flow through them. The packets are of more interest for forensic investigation that is why enterprises should retain network traffic logs.
2. The second policy is to planning the response to an attack has to establish a forensic team includes members from upper management, Human Resources, the technical staff, and outside members. Also to establish an intrusion response procedure of step-by-step guide that employees can follow if an attack is suspected and to formalize the investigative procedure to followed by the computer forensic experts during a forensic investigation.
3. The third policy is to give special training to the response team, investigative team and to all the persons of an enterprise who uses computers. The training is to know the CNF procedures to follow and to use. During a preliminary investigation, the investigative team will use these skills to determine whether an attack actually occurred, and if possible to identify the crime by determining how it was committed and who did it, and find the evidence left behind. In order to do this, the investigative team needs to understand the steps followed by the attacker so that they can be retraced. The team must also know where to find possible evidence. It is essential that forensics investigators be expert in computer and network administration so that they know the technical in's and out's of the target systems.
4. The fourth policy states to accelerate the investigation as quickly as possible through prohibiting personal file encryption as it may not be possible to ever recover the original contents, prohibiting disk scrubbing tools and file shredding software as they wipe out or destroy the information, utilizing data indexes for every packets in a log to minimize the search, utilizing information fusion techniques of IDS for a large volumes of data to store. Additionally, the more time the investigation takes the more the chance that potential evidence will be destroyed or compromised.
5. The fifth policy states to prevent anonymous activities and to protect personal privacy on the Internet. It is difficult to do the investigation if anonymity is allowed. To prevent the anonymity, the Onion routing research project is building an Internet based system that strongly resists traffic analysis, eavesdropping, and other attacks both by outsiders (e.g. Internet routers) and insiders (Onion Routers themselves). It requires date, time, user stamps in file to know what date a file was created, or modified, or deleted, and who did it. The user has to follow the strong authentication policy to access the system but the passwords are vulnerable so the encryption-based authentication is effective. Also to use strong access control mechanisms for limiting use of resources to authorized users.
6. The sixth policy state to protect the evidence from an attacker who tries to destroy the evidence of a crime or an employee tries to erase incriminating data from log files. A cornerstone of effective CNF is to have strong authentication and integrity services that control administrative access to network devices. It is preferable to use the encrypt evidence files and connections to guarantee the security and integrity of the data. Also to apply strong integrity checking technology to show the evidence has not been corrupted.

IV. SOFTWARE FORENSIC

Research in the field of software forensics has been carried out to identify the author of a computer program [11]. Various objective metrics, such as the proportion of blank lines, the proportion of comments, the average length of identifiers and statistics based on those metrics have been proposed to characterize the author of a program with some success. Kilgour et al. [12] proposed the use of other variable measures such as the presence or absence of a feature in an attempt to identify better the authorship of a program. Certain structural features can be used to classify e-mail message authorship. Email messages have macro-structural features, such as the presence or absence of greetings, farewells, signatures and attachments that can be analyzed along with the micro-features of the text contained within them although these are readily falsified. The last one is very new and very little work has been done in this area, though serious efforts are being made to evolve procedures and tools that would stand the test of legal scrutiny. Source code is the textual form of a computer program that is written by a computer programmer in a computer programming language. These programming languages can in some respects be treated as a form of

language from a linguistic perspective, or more precisely as a series of languages of particular types, but within some common family. In the same manner as written text can be analyzed for evidence of authorship, as in [13], computer programs can also be examined from a forensics or linguistics viewpoint [14] for information regarding the program's authorship. The goals of computer program authorship are also often similar to, or even identical to, those encountered in forensic linguistics and computational linguistics.

Once the classification is made that program source code is in fact a type of language that is suitable for authorship analysis, a number of applications and techniques emerge. Similarly, techniques used in forensics for handwriting and linguistic analysis can also, in some cases at least, are transferred in some respect to what is referred to here as *software forensics*. It is assumed that the term software forensics refers to the use of measurements from software source code, or object code, for some *legal* or *official* purpose [15]. This is similar to, but in some respects also distinct from, the use of the term in some literature where the focus tends to be very much on malicious code analysis. The legal or official nature of software forensics requires a high level of objectivity, as well as methods for calculating the degrees of evidence provided and combining that evidence with other sources.

A. Applications

There are four broad areas of application emerge in software forensics that are discussed.

Author identification: The goal here is to determine the likelihood of a particular author having written some piece(s) of code, usually based on other code samples from that programmer. This can also involve having samples of code for several programmers and determining the likelihood of a new piece of code having been written by each programmer. An example of this applied to source code would be ascribing authorship of a new piece of code, such as a computer virus, to an author where the code matches the profile of other pieces of code written by this author.

Authorship discrimination: This is the task of deciding whether some pieces of code were written by a single author or by (some number of) different authors. This can possibly also include an estimate of the number of distinct authors involved in writing a single piece or all pieces of code. It is obviously necessary to distinguish between identifying multiple authors for a series of programs and co-authorship on a single program. This task involves the calculation of similarity between the two or more pieces of code and possibly some estimate of between- and within-subject variability.

Author characterization: This is based on determining some characteristics of the programmer of a code fragment, such as personality and educational background, based on their programming style. An example of this would be determining that a piece of code was most likely to have been written by someone with a particular educational background due to the programming style and techniques used.

Author intent determination: It may be possible to determine, in some cases, whether code that has had an undesired effect was written with deliberate malice, or was the result of an accidental error. Since the software development process is never error free and some errors can have catastrophic consequences, such questions can arise reasonably frequently. This can also be extended to check for negligence, where erroneous code is perhaps suspected to be much less rigorous than a programmer's usual code. This is a much-neglected aspect of source code authorship analysis [15] with no other literature found that mentions its use.

B. Metrics for software forensics

Source code metrics: Expert opinion can, potentially, be given on the degrees of similarity and difference between code fragments. Psychological analysis of code can also be performed, even as a simple matter of opinion. However, a more scientific approach may also be taken (and should be taken) since both quantitative and qualitative measurements can be made on computer program source code and object code. These measurements can be either automatically extracted by analysis tools, calculated by an expert, or arrived at by using some combination of these two methods. Some metrics can obviously only be calculated by an expert, such as the degree to which the comments in code match the actual behaviour of that code.

Object code metrics: While not part of source code analysis itself, some environmental measurements can sometimes also be extracted from executable code such as the hardware platform and the compiler employed for its production. Executable code can also be decompiled; a process where a source program that could then be compiled into the executable is created by reversing the compiling process. Since many source programs can be written to create the same executable there is considerable information loss, but some of the source code metrics can still be applicable.

Metric models of authorship: Once these metrics have been extracted, a number of different modelling techniques, such as cluster analysis, logistic regression, and discriminant analysis, can be used to derive models. The form of the model, the technique used, and the metrics of use all depend greatly on the purpose of the analysis and on the information available. In most respects the particular technique used for the modelling process is less important than the variables selected and their coding.

CONCLUSIONS

The cause of the failure is due to the third party involvement in the system. This paper present various forensic methods that are available in the society by the researchers would be of immense benefit to the engineering profession. It advises all engineers, whether young or old, experienced or just commencing on their careers, to gain an understanding of why failures occur and how they can be forensically analysed.

REFERENCES

[1] Forensic ECBS: The Way Forward, Darren Dalcher, School of Computing Science, Middlesex University, Trent Park, Bramley Road, London NI4 4YZ, UK. d.dalcher @mdx.ac .uk

[2] Séamus Ó Ciardhuáin An Extended Model of Cybercrime Investigations, *International journal of Digital Evidence*, Volume 3, Issue 1, Summer 2004.

[3] Kruse, Warren G. II and Jay G. Heiser (2001), "Computer forensics: *Incident Response Essentials*", Addison-Wesley Pub Co., 2001.

[4] McMillan & Jim "Federal Guidelines for searching and seizing computers "http://www.usdoj.gov/criminal/cybercrime/search_docs/toc.hm,2000.

[5] Lavoie, Regean, "forensic Acquiring and Analysis", SANS Institute, 2003.

[6] Alec Yasinsac, Yanet Manzano, Policies to Enhance Computer and Network Forensics, *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security United States Military Academy*, West Point, NY, 5-6 June, 2001.

[7] McKemmish, Rodney (1999) "What is forensic computing?" *Australian Institute of Criminology*, trends & issues in crime and criminal justice No. 118, 1999.

[8] Eoghan Casey, Handbook of Computer Crime Investigation: Forensics, Tools and Technology, *Academic Press*, published 2002.

[9] Michael Caloyannides, Computer Forensics & Privacy, *Artech House,* published 2001

[10] Albert J. Marcella Jr., Robert S. Greenfield, Cyber Forensics: A Field Manual for Collecting, Examining, and Preserving Evidence of Computer Crimes, *Auerbach publication*, 2002.

[11] I. Krsul and E. H. Spafford. Authorship analysis: Identifying the author of a program. *Computers and Security*, 16(3): pp. 233–57, 1997.

[12] R. I. Kilgour, A. R. Gray, P. J. Sallis, and S. G. MacDonell, A fuzzy logic approach to computer software source code authorship analysis. *In International Conference on Neural Information Processing and Intelligent Information Systems,* Springer-Verlag, Singapore, pp. 865–868, 1997.

[13] Sallis P., Aakjaer, A., and MacDonell, S. (1996). Software Forensics: Old Methods for a New Science. *Proceedings of SE:E&P'96 (Software Engineering: Education and Practice).* Dunedin, New Zealand, IEEE Computer Society Press, 367-371.

[14] Dunsmore, H.E. (1984). Software Metrics: An Overview of an Evolving Methodology. *Information Processing & Management* 20:183-192.

[15] Gray, A.R., Sallis, P.J., and MacDonell, S.G. (1997) Software Forensics: Extending Authorship Analysis Techniques to Computer Programs. Presented at *The Third Biannual Conference of the International Association of Forensic Linguists*, 4-7 September 1997, at Duke University, Durham, North Carolina, USA.