Bridging the Divide: Rising Awareness of Forensic Issues amongst Systems Administrators

Vlasti Broucek, Paul Turner
School of Information Systems, University of Tasmania, Australia

Introduction

The traditional divide between technical and legal areas of expertise is increasingly problematic in the age of malware, hactivism and cyber-warfare. Technical advances in the ability of systems to detect intrusions, denial of services attacks and also to enhance network monitoring and maintenance are well documented and subject to constant research and development. These advances provide systems administrators with tools to treat the symptoms of such illegal and/or inappropriate behaviours. However, these technical responses do little to treat the causes of these or future problematic behaviours. Significantly, these technical solutions are not currently designed to collect forensic data hence making evidence acquisition difficult. Additionally, systems administrators are often unfamiliar with the processes for evidence collection, collation and presentation suitable for the legal sphere. At the same time, legal regimes globally are struggling to address the challenges posed by illegal and inappropriate behaviours conducted in cyberspace. These challenges include both conceptual problems of definition and practical problems of legal jurisdiction. Most legal professionals are suffering from a limited understanding of technical advances and continue to lack confidence in the ability of technical specialists to produce evidence that will be admissible in a court of law.

In this context, this paper considers the emergence of forensic computing as an area of academic expertise and the importance of forensic awareness for network and system administrators, computer security community and legal professionals in handling criminal, illegal and other inappropriate on-line behaviours.

The paper is divided into two main parts. Part one discusses the significance of the relationship between computer security and forensic computing and how this has affected the approaches of systems administrators to cyber-attacks. This part defines the relationship between the two areas, explores their commonalities and identifies the key fundamental differences between them (Broucek & Turner, 2001a, 2001b; Patel & Ciardhuáin, 2000). Part two explores the issues aligned to evidence acquisition and the suitability of Intrusion Detection Systems (IDS) for preparing legally admissible evidence. This part reveals strong disagreement amongst technical and legal experts over the suitability of IDS as a tool for collecting, collating and presenting forensic evidence (Sommer, 1998, 1999; Stephenson, 2000a).

Before examining the key differences between computer security and forensic computing, it is useful to briefly consider the major areas contributing to the emergence of forensic computing as an academic discipline. In previous work by Broucek and Turner (2001a; 2001b) the complexity of the field has been highlighted and a preliminary taxonomy developed. Currently this taxonomy identifies five main building blocks of forensic computing. These are as follows:

- **Computer science** and in particular:
  - Computer Security,
  - Operating Systems and Application Software,
  - Systems Programming and Languages.
- **Law** and in particular:
  - Specialist Computer, Telecommunications and Media Law,
  - Criminal Law,
  - Civil Law,
  - Soft Law.
- **Information Systems** and in particular:
  - Systems Management,
  - Organisational IT/IS policies,
  - User Education.
- **Social Sciences** and in particular:
  - Socio-political issues related to privacy and encryption,
  - Surveillance, activism, hacktivism,
  - Cyberterrorism and cyber-warfare.
- **Non-specific** and in particular:
  - Documentation skills,
  - Development of FC skill sets,
  - Evidential authentication

The taxonomy illustrates the breadth of the discipline, the range of technical, legal, organisational and social factors at play and the need for a coherent approach to combining the different skill sets required to undertake forensic investigations.

Part One

At the same time as the term forensic computing (FC) has become more widely used a common misconception has emerged that it is simply another ‘fancy’ name for computer security. This part of the paper dispels this misconception and examines a number of key differences between the two fields. At the broadest level, computer security is focused on the prevention and detection of systems attacks (e.g. denial of services, viruses, hacking) while forensic computing examines a broader range of illegal and inappropriate on-line behaviours and is focused primarily on collection and presentation of evidence.

Differences between computer security and forensic computing

One very tangible example illustrating difference between the two fields has been identified by Patel and Ciardhuáin (2000) in their consideration of the impact of forensic computing on telecommunications. To illustrate this difference Patel and Ciardhuáin (2000) examine the distribution of child pornography on the Internet. This activity is illegal in the majority of legal jurisdictions and is increasingly a focus for forensic computing investigations conducted by law enforcement agencies. However, this activity is carried out without breaking any computer security systems and is not prevented or detected by the majority of computer security tools. It is possible for any Internet user to set-up a web site on their home computer and to place child pornography on it. Similarly, it is possible for Internet users to place child pornography on any publicly available web services without breaking computer security.

Another example illustrating a difference between the two fields has been discussed by Broucek and Turner (2001a; 2001b). Using their own experience in network administration at the University of Tasmania they examine an instance of e-mail communications being used to send life threatening messages to a female student. Again these repugnant activities were the focus of a forensic investigation but did not involve the breaching of any computer security. Following Patel and Ciardhuáin (2000) the most important differences between computer security and forensic computing can be illustrated through answers to four questions about the foci of the two fields. These questions are: why, when, who and for whom?

- **Why** - Computer security is in place to protect against and to detect cyber-attacks. Forensic computing does not protect against the attack. However, it is worth noting that as the field of forensic computing evolves, more proactive forensic tools may be developed that will blur this distinction. Currently, there are numerous computer security resources available to network and systems administrators to prevent and detect cyber-attacks. But these tools are not designed to provide data sets that are suitable for the generation of forensic evidence. This results in either insufficient data being collected or the data being potentially unreliable. Both raise problems for collection and presentation of forensic evidence.
- **When** - Computer security is ideally conducted in real time while forensic computing is primarily conducted 'post mortem' i.e. after the occurrence of inappropriate, criminal or other illegal behaviour. However, as noted in previous point this distinction will be again narrowed with the evolution of more proactive forensic computing tools.
- **Who** - Computer security is primarily conducted by computer specialists. Forensic computing can be conducted by a wide range of professionals, many of whom do not have specialised computer security skills. Of course, ideally forensic computing specialists should be trained in all the disciplines that were identified in the introduction. Indeed, there has already been at least one senior US government official proposing the establishment of training facilities and funding to develop these types of professionals (Reno, 1996). The same issues were identified during Police Commissioners' Conference on Electronic Crime Strategy held in March 2001 in Australia (Australasian Centre For Policing Research, 2001).

- **For whom** - Computer security is usually subject to minimal presentation requirements. Also if it is presented this is usually to a highly technically literate audience. However, the results of forensic investigation always are presented to non-IT/IS audiences and frequently in the context of legal proceedings.

Aside from these overt differences, there is a major problem with existing computer security approaches that is only revealed by adopting a forensic computing perspective. This problem emerges from the fact that computer security measures can be, and often are bypassed by 'trusted users'. This points to a broader and age-old question of 'who polices the police?' With root access, it is possible to do almost anything on the systems including modifying or deleting log files and disabling system and security processes. Following Farmer (2001) deleted files can be relatively easy to restore however the processes used problematise the legal validity of the subsequent data set[1]. From a forensic computing perspective the lack of existing checks and balances on users with root access raises a fundamental problem concerning the integrity of computer data to be used as evidence.

Despite these differences computer security expertise is a central plank for developing the appropriate skill sets for the field of forensic computing. The ability to identify, track, trace and analyse log files is central to forensic investigations where digital evidence is main source of data. However, the forensic computing perspective moves beyond these technical skills to develop sensitivity towards questions over the admissibility of evidence and legal validity of particular data sets. This is particularly the case during the analysis of log files where 'dirtying of the data' or 'acontextual' presentation may significantly alter the meaning of the evidence.

Part Two

Following on from the previous discussion, the reliance on digital data for forensic computing investigations raises basic questions over the ability of existing systems to generate comprehensive and reliable data sets. This part of the paper begins with an example of problems arising in using log files as a source of forensic data. The paper then proceeds to examine intrusion detection systems (IDS) to illustrate further technical and legal problems related to the conduct of forensic investigations. Firstly these IDS are classified and examined in terms of their technical capability to generate forensic data sets. Secondly, deploying the Rome Labs case the legal dimensions of digital evidence acquisition and legal presentation are examined.

---

[1] *"The persistence of data, however, is remarkable. Contrary to the popular belief that it's hard to recover information, it's actually starting to appear that it's very hard to remove something even if you want to. The unrm/lazarus combination is a fine, if a bit unsettling, trash can analyzer. And while the results can be spotty for simple single file 'undeletion,' robbing graves for fun and profit can be a lucrative venture for an aspiring forensic analyst. Indeed, when testing this software on a disk that had been used for some time on a Windows 95 machine, then reinstalled to be a firewall using Solaris, and finally converted to be a Linux system, files and data from the prior two installations were clearly visible. Now that's data persistence!"* (Farmer, 2001) This statement is technically remarkable, however legally questionable.

Log Files as a source of Forensic Data.

Following Farmer and Venema (2000) the login session in Figure 1 reveals information recorded by three different login facilities of UNIX system. These are examined in turn to reveal problems in using this information in a forensic context.

```
May 25 10:12:46 spike telnetd[13626]: connect from hades.porcupine.org

       wietse   ttyp1   hades   Thu May 25 10:12-10:13 (00:00)

              hostname -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              sed      -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              stty     -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              mesg     -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              who      -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              w        -   wietse   ttyp1   0.00 secs Thu May 25 10:12
              ps       -   wietse   ttyp1   0.00 secs Thu May 25 10:13
              ls       -   wietse   ttyp1   0.00 secs Thu May 25 10:13
              w        -   wietse   ttyp1   0.00 secs Thu May 25 10:13
              csh      -S  wietse   ttyp1   0.03 secs Thu May 25 10:12
              telnetd  -S  root     ----    0.00 secs Thu May 25 10:12
```

Figure 1 - Adapted from Farmer and Venema (2000)

First, the entry from TCP wrappers log file (shown in the first line) shows that on May 25, at 10:12:46 local time, the machine spike received a telnet connection from the machine hades.porcupine.org. From a forensic perspective this is problematic because the TCP wrappers log files contain only information about the initial connection event. They do not provide a corresponding record for the end of the telnet connection. Therefore, for the forensic investigation, alternative sources of information have to be collected to substantiate the length of the connection.

Second, the output from Unix *last* command (shown in the line starting with wietse) shows that the user wietse was logged in on port ttyp1 from host hades. This log apparently reveals that the login session lasted from 10:12 until 10:13 but does not provide details of how many seconds the session lasted, hence the log file shows 00:00 for the length of the session. From a forensic perspective this is problematic because there is a lack of detail as to the length of connection time. The system output (00:00) is clearly in conflict with the connection time start-end 10:12-10:13. It is also worth noting that this output uses only short name for the machine from which user wietse connected, while TCP wrappers used full name.

Third, output from the Unix *lastcomm* command (lines starting hostname through to telnetd) shows the commands executed by the user wietse. Additionally these lines show how much CPU time each command consumed in seconds, and at what time each command started (the last column of these lines). Significantly, *lastcomm* provides a record of the order in which each process is **terminated** (the column hostname through to telnetd). In this case the command interpreter (csh) and the telnet daemon (telnetd) appear at the end even though they were actually the first two processes started. From a

forensic computing perspective this is problematic because there is no reliable and detailed record of the order in which the commands were actually run and for how long.

These technical weaknesses demonstrate the inadequacy of available login systems for forensic computing because the data provided is insufficiently complete, accurate or continuous. Additionally, as previously mentioned, any user with root access could easily modify these log files used by commands *last* and *lastcomm* to generate their respective outputs. Combined these problems create major difficulties for the collection and presentation of digital evidence suitable for a court of law.

In further examining the technical and legal problems related to the conduct of forensic investigations, the next section classifies IDS, identifies their technical problems and reviews the 'Rome Labs case' to illustrate the legal challenges that could be made to the evidence they produce.

Classifying IDS

"*Intrusion detection systems are software or hardware systems that automate the process of monitoring the events occurring in a computer system or network, analysing them for signs of security problems*" (Bace & Mell, 2001)

Following Bace and Mell (2001) IDS can be classified according to their three main functional attributes. These are as follows:

- Information source,
- Analysis,
- Response.

Information Source

Bace and Mell (2001) identify three different sources of information used by IDS: network; host; and application. Laing (2000) introduces a fourth source of information, TCP/IP stack. Based on these four sources of information, four major types of IDS are now recognised and briefly discussed below.

- **Network based IDS** (NIDS) collect raw network packets as a data source, usually using the network interface in 'promiscuous' mode. These interfaces may also be run in 'stealth' mode to hide them from attackers of the network. NIDS form the majority of commercial IDS. An example of this type of IDS is widely used and freely available *snort* (Roesch, 1999, 2001a, 2001b).
- **Host based IDS** (HIDS) usually monitor and analyse system log files and operating system audit trails. These systems can either run in real time or periodically. Some HIDS can also monitor network ports on hosts for activity on these ports. Modern commercial system of this type is described in Crosbie and Kuperman (2001).
- **Application based IDS** (AIDS) are a special subset of host based IDS. The source of information for these systems is usually applications audit log.

- **Stack based IDS** (SIDS) represent the latest IDS technology. These systems work directly on the TCP/IP stack monitoring packets during their transport through OSI layers. More significantly they monitor not only incoming traffic, but also outgoing traffic.

Each of these systems has significant advantages and disadvantages that have been previously analysed in-depth by Bace and Mell (2001) and Laing (2000) and many other publications.

Analysis

Based on the approach to analysis of data, IDS can further be classified into two distinctive groups: misuse detecting; and anomaly detecting systems.

- **Misuse detecting** systems work on the pattern matching principle i.e. looking for patterns known to be associated with particular events. These systems are very often called signature or fingerprint based IDS. Commercial products frequently use separate signatures for each particular attack. This leads to the generation of huge signature files and reduces the IDS pattern matching speed. These systems are only as good as their signature database and can detect only attacks that are already in the database.
- **Anomaly detecting** systems attempt to identify abnormal or unusual behaviour on the network or host. These systems use several different methods (threshold detection, statistical measures, rule based, neural networks, expert systems,) to analyse current system activities in comparison to historical profiles. According to Bace and Mell (2001) only the first two methods are used in currently available commercial intrusion detection systems. These systems produce large numbers of false alarms, however, with careful design they can be used to meaningfully detect new attacks. They provide a significant advantage over signature based IDS.

More recently anomaly based IDS have attracted considerable attention from the social sciences because of their potential implications for the privacy of users. These systems can for example easily be used to identify anomalies or changes in working time patterns that raise concerns about user surveillance and interference with personal privacy (Lundin, 2000; Lundin & Jonsson, 1999b).

Response

IDS can further be classified according to the response options deployed. Two main types can be identified: active; and passive.

- **Active** responses can trigger further collection of data by starting another application, can change the environment, or even take direct action against intruders. The last form of active response is mainly considered in information warfare circles and is not widely recommended due to significant legal issues.

- **Passive** responses provide information to system users and administrators. These responses are in the form of alarms and notifications, SNMP traps etc.

IDS Technical Issues for Forensic Computing

From the above classification it is evident that intrusion detection is a rapidly expanding field in which numerous different approaches are being developed to automate protection and prevention of security breaches in computer systems. Before exploring the legal issues surrounding evidence acquisition and the suitability of IDS for the requirements of legal admissibility it is useful to identify some technical issues that problematise IDS as forensic tools.

First, current intrusion detection systems may be unable to collect all the data they are supposed to collect. Many IDS currently available cannot adequately cope with 100 Megabit per second networks even though most backbone services operate on Gigabit speeds. However, research is currently being conducted to overcome this problem and to increase both the speed of collection of the data as well as its analysis (Desai, 2002; Handley, Paxson, & Kreibich, 2001).

Second, the data collected by IDS may be insufficient. For example, where an IDS monitoring a network detects a cyber-attack from a particular machine it will identify it by its IP address. However, existing problems with IP addressing raise a number of technical concerns. While the IP address can be traced easily, it must be questioned as to how reliable the trace is, particularly for the 'last hop' (the connection between the last two computer hosts involved in the trace). Recent research (Clayton, 2000) has reported several major limitations to traceability in legal proceedings.

For example, if an IP address is traced back to a University network in the USA from an IDS in Australia, it may appear to the Australian network administrator that they have traced a particular computer. However, this technical assumption would never stand up to legal scrutiny. This is because many US colleges provide network access to dormitories and make maintenance easier by using dynamic DHCP (each computer can have a different IP address each time it is used). Another reason is a flaw in network card design. The IP assignment is based on MAC address, but unfortunately many network cards enable the changing of the MAC address, which prevents it from being a unique identifier. Even without this flaw in hardware design, it is easy to spoof MAC address and then use tools freely available from the Internet to pretend to be a different computer on the network. Finally, even most correct trace of IP addresses doesn't provide investigator with any information about the person associated with it. Particularly in environments where the computers are shared and poor access control is implemented such information (IP) is rendered nearly useless.

Third, the intrusion detection systems are themselves susceptible to a variety of attacks and some authors argue (Handley et al., 2001; Mell, Marks, & McLarnon, 2000; Ptacek & Newsham, 1998) that the majority of these systems are fundamentally flawed. In

other words, the data collected by these systems may itself have been tampered with before the attack was discovered and/or investigated.

In turning to examine the legal issues surrounding the use of IDS as a tool for collecting, collating and presenting forensic evidence, the 'Rome Labs' case provides a good example of the strong differences of opinion that are evident amongst experts in the forensic computing field. The 'Rome Labs' Attack refers to a hacker attack against the Rome Air Development Center, Griffiss Air Force Base, New York, on March 28, 1994. This case is interesting not only because the proceedings against one of the attackers were initiated in the United Kingdom.

Jim Christy (Christy, 1998) gave a report of this incident to the Senate Governmental Affairs Committee on May 22, 1996 that highlighted a number of key aspects of the case:

- The attack was only discovered five days after it occurred;
- The responsible commander at the Air Force Base allowed several systems to be kept open thereby allowing the forensic investigating team to trace the attackers.
- Despite a thorough investigation by the Air Force Office of Special Investigations (OSI), numerous questions remained unanswered These can be summarised as follows:
    - The identity and motivation of the second attacker, nicknamed Kuji;
    - The extent of the attack;
    - The extent of the damage.

Peter Sommer from the Computer Research Security Centre at the London School of Economics was subsequently hired by defence lawyers in the UK to assess the quality of the evidence prepared by OSI. Sommer's assessment was significant in calling into question the quality of the evidence for presentation in a court of law. Subsequently published, Sommer (1998; 1999) states that "*Almost every individual stream of evidence could be challenged*".

Sommer goes on to discuss in detail several streams of evidence provided by OSI and as a result of his analysis provides a list of 11 points that should be followed in development of any new intrusion detection systems. Sommer's work appears to be supported by the findings of the NSTAC Network Group Intrusion Detection Subgroup that in its December 1997 reports that:

- *"Current intrusion Detection Systems are not designed to collect and protect the integrity of the type of information required to conduct law enforcement investigations"*.
- *"There is a lack of guidance to employees as to how to respond to intrusions and capture the information required to conduct a law enforcement investigation…"*(Sommer, 1998, 1999)

Unfortunately, Sommer's findings were never put to the test in a court, because the Rome Labs case ended in a form of a 'plea bargaining' deal and defendant was not tried.

In contrast to Sommer's views, other authors have argued that IDS are indeed suitable for evidence acquisition (Stephenson, 2000a, 2000b) while others choose to ignore the legal aspects of evidence acquisition (Yuill, Wu, Gong, & Huang, 1999). In reality the opinions of these different authors are probably not as polarised as they might first appear. However, the disagreements do highlight that while IDS are among the best tools available to date for 'trying' to collect data that could be used in a prosecution, serious problems remain.

Conclusion

Where to from here? – Insights for network administrators

This paper has highlighted three inter-related aspects of digital evidence. These are as follows:

- **Legal admissibility** - In terms of legal principles, digital evidence per se is now commonplace in most legal jurisdictions. More specifically it is now widely accepted that digital evidence in the form of computer log files is also admissible 'in principle'. For example, in the USA - code title 28, section 1732 states that *"logs files are admissible as evidence if they are collected in the regular course of the business"*. However, this principle of admissibility does not provide any guarantee that in any particular case log files will be deemed legally valid.
- **Legal validity/weight –** following on from the previous point, there is an implication that the question of admissibility must be established in each and every instance according to the rules of evidence that apply in the legal jurisdiction concerned. This illustrates the point that 'scientific proof' and 'legal proof' are often quite distinct. Again following Sommer (1998; 1999) *"while scientific proof depends on the application of generally recognised methods of scientific investigation, legal proof depends more on the rule of admissibility of evidence and what is convincingly presented in court"*. These issues can be categorised as 'admissibility and weight' i.e. the legal validity of evidence for a submission in a particular jurisdiction and the ability of the court to be convinced by its presentation. As is readily apparent from these points there is a marked distinction between digital evidence from a technical point of view and from a legal perspective.
- **The legal dimensions of the conduct of forensic analysis** (i.e. the manner in which the methods of forensic investigation may problematise digital evidence). At the broadest level, any analysis must not contaminate the 'crime scene' and must ensure that throughout the investigation that this scene is not changed (Anderson, 1998; Bates, 1997, 1998, 2001; McKemmish, 1999). Although the principles of 'chain of custody' or continuity of evidence and 'auditability' are well known in forensic circles, there remains a general lack of awareness of these principles within the computer security community. As a consequence the dangers of 'dirtying the data' remain prevalent. An additional issue that emerges during analysis concerns 'acontextual' presentation of individual entries in log files. This can lead to a misrepresentation of the significance or insignificance of individual entries and of the log file as a whole.

From the above points a number of key principles for systems administrators to follow in considering data gathering for legal proceedings can be identified. These have been discussed in several publications (Anderson, 1998; Bates, 1997, 1998, 2001; McKemmish, 1999) and can be summed up as:

- **Minimise Handling of the Original** - always use binary-based backup methods and at least MD5 checksums and use only copies of original data for the analysis.

- **Account for any Change** - keep detailed logbooks of the steps made during the investigation. Keep in mind that investigator has to be able to repeat these steps.

- **Comply with the Rules of Evidence -** the presentation of finding has to be made in such a manner that it does not alter the meaning of the evidence.

- **Do not Exceed Your Knowledge** - it is imperative that forensic investigators do not undertake investigations in areas beyond their existing skill sets. By 'learning on the job' the danger of 'dirtying the data' or even damaging the entire evidence set dramatically increases. A guide to forensic analysis of Unix system is provided at http://staff.washington.edu/dittrich/misc/forensics/

Further research

The research in forensic computing is still relatively immature and numerous new avenues of research have already begun to emerge. For example, the development of a 'Black Box' similar to the flight recorders used on aircraft as a means of capturing a complete systems record (Patel & Ciardhuáin, 2000). Some work has already been undertaken in this area, although addressing the protection of users privacy within such systems remains problematic. The easiest solution to the forensic need for evidence may appear to be to collect all data on a continuous basis. Unfortunately, recent work on anomaly based IDS and privacy suggests that the issue of privacy is even more complicated than it might have been imagined (Biskup & Flegel, 2000; Lundin, 2000; Lundin & Jonsson, 1999a).

Finally, several researchers are currently arguing that all logging systems and intrusion detection systems are suitable only for data collection up to the point when the cyber-attack occurs. This perspective highlights the fact that after an attack it is highly problematic to deal with data collected from 'untrustworthy' machines. Several research groups are working on creating so call trusted log files on untrusted machines using cryptography and distributed environments (Arona, Bruschi, & Rosti, 1999; Biskup & Flegel, 2000; Schneier & Kelsey, 1998, 1999; Sommer, 1997). However, the notion of having trusted log files on 'untrustworthy' machines remains to be tested by legal professionals.

References

Anderson, M. R. (1998). *Computer Evidence Processing: Good Documentation Is Essential*. Retrieved December 20, 2000, from http://www.forensics-intl.com/art10.html

Arona, A., Bruschi, D., & Rosti, E. (1999, 6-10 December 1999). *Adding availability to log services of untrusted machines.* Paper presented at the 15th Annual Computer Security Applications Conference (ACSAC'99), Phoenix, AZ, USA.

Australasian Centre For Policing Research. (2001). *Electronic Crime Strategy of the Police Commissioners' Conference, Electronic Crime Steering Committee, 2001 - 2003*: Australasian Centre For Policing Research.

Bace, R., & Mell, P. (2001, November 2001). *Intrusion Detection Systems*. Retrieved March 23, 2002, from http://csrc.nist.gov/publications/nistpubs/800-31/sp800-31.pdf

Bates, J. (1997). Fundamentals of Computer Forensics. *International Journal of Forensic Computing*(January/February 1997).

Bates, J. (1998). Forensic lessons - case study. *International Journal of Forensic Computing, 1998*(No 20), 16-19.

Bates, J. (2001). *DIVA Computer Evidence (Digital Integrity Verification and Authentication)*. Retrieved 26 March 2001, 2001, from http://www.forensic-computing.com/archives/diva.html

Biskup, J., & Flegel, U. (2000, July 2000). *On Pseudonymization of Audit Data for Intrusion Detection.* Paper presented at the Workshop on Design Issues in Anonymity and Unobservability, Berkeley, California.

Broucek, V., & Turner, P. (2001a, 11 July 2001). *Forensic Computing: Developing a Conceptual Approach for an Emerging Academic Discipline.* Paper presented at the 5th Australian Security Research Symposium, Perth, Australia.

Broucek, V., & Turner, P. (2001b). Forensic Computing: Developing a Conceptual Approach in the Era of Information Warfare. *Journal of Information Warfare, 1*(2), 95-108.

Christy, J. (1998). Rome Laboratory Attacks: Prepared Testimony of Jim Christy, Air Force Investigator, before the Senate Governmental Affairs Committee, Permanent Investigation Subcommittee, May 22, 1996. In D. E. Denning & P. J. Denning (Eds.), *Internet Besieged: Countering Cyberspace Scofflaws* (pp. 57-65): ACM Press.

Clayton, R. (2000). *The Limits of Traceability*, from http://www.cl.cam.ac.uk/~rnc1/The_Limits_of_Traceability.pdf

Crosbie, M. J., & Kuperman, B. A. (2001). *A Building Block Approach to Intrusion Detection.* Paper presented at the RAID 2001.

Desai, N. (2002). *Increasing Performance in High Speed NIDS: A loot at Snort's Internals*. Retrieved March 13, 2002, from http://www.snort.org/docs/Increasing_Performance_in_High_Speed_NIDS.pdf

Farmer, D. (2001). Bring Out Your Dead. The Ins and Outs of Data Recovery. *Dr Dobb's Journal, 30*(1).

Farmer, D., & Venema, W. (2000). Forensic Computer Analysis: an Introduction. Reconstructing Past Events. *Dr Dobb's Journal, 29*(9), 70-75.

Handley, M., Paxson, V., & Kreibich, C. (2001). *Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics.* Paper presented at the 10th USENIX Security Symposium, Washington, DC, USA.

Laing, B. (2000). *How To Guide: Impletmenting a Network Based Intrusion Detection System*. Retrieved November 21, 2001, from http://www.snort.org/docs/iss-placement.pdf

Lundin, E. (2000). Anomaly-based intrusion detection: privacy concerns and other problems. *Computer Networks, 34*(4), 623-640.

Lundin, E., & Jonsson, E. (1999a, 7-9 September 1999). *Privacy vs Intrusion Detection Analysis.* Paper presented at the The 2nd International Workshop on Recent Advances in Intrusion Detection (RAID'99), Lafayette, Indiana, USA.

Lundin, E., & Jonsson, E. (1999b, 1-2 November 1999). *Some Practical and Fundamental Problems with Anomaly Detection.* Paper presented at the The fourth Nordic Workshop on Secure IT systems (NORDSEC'99), Kista, Sweden.

McKemmish, R. (1999). What is Forensic Computing. *Trends and Issues in Crime and Criminal Justice*(118).

Mell, P., Marks, D., & McLarnon, M. (2000). A denial-of-service resistant intrusion detection architecture. *Computer Networks, 34*, 641-658.

Patel, A., & Ciardhuáin, S. Ó. (2000, November 2000). The Impact of Forensic Computing on Telecommunications. *IEEE Communications Magazine,* 64-67.

Ptacek, T. H., & Newsham, T. N. (1998). *Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection*. Retrieved November 11, 2001, from http://www.snort.org/docs/idspaper/

Reno, J. (1996). Law Enforcement in Cyberspace Address. In D. E. Denning & P. J. Denning (Eds.), *Internet Besieged: Countering Cyberspace Scofflaws* (pp. 439-447): ACM Press.

Roesch, M. (1999, November 7-12). *Snort - Lightweight Intrusion Detection for Networks.* Paper presented at the 13th Systems Administration Conference - LISA '99, Seattle, WA.

Roesch, M. (2001a). Snort 1.8.3 [man pages].

Roesch, M. (2001b, November 6, 2001). *Snort Users Manual - Snort Release: 1.8.3*. Retrieved November 12, 2001, from http://www.snort.org

Schneier, B., & Kelsey, J. (1998, January 26-29). *Cryptographic Support for Secure Logs on Untrusted Machines.* Paper presented at the 7th USENIX Security Symposium, San Antonio, Texas.

Schneier, B., & Kelsey, J. (1999). Secure Audit Logs to Support Computer Forensics. *ACM Transactions on Information and System Security, 2*(2), 159-176.

Sommer, P. (1997). Downloads, Logs and Captures: Evidence from Cyberspace. *Journal of Financial Crime*, 138-152.

Sommer, P. (1998, 14-16 September 1998). *Intrusion Detection Systems as Evidence.* Paper presented at the Recent Advances in Intrusion Detection - RAID'98, Louvain-la-Neuve, Belgium.

Sommer, P. (1999). Intrusion Detection Systems as Evidence. *Computer Networks, 31*(23-24), 2477-2487.

Stephenson, P. R. (2000a, 2-4 October 2000). *The Application of Intrusion Detection Systems in a Forensic Environment.* Paper presented at the Recent Advances in Intrusion Detection - RAID 2000, Toulouse, France.

Stephenson, P. R. (2000b). *Intrusion Management: A Top Level Model for Securing Information Assets in an Enterprise Environment.* Paper presented at the EICAR 2000.

Yuill, J., Wu, S. F., Gong, F., & Huang, M.-Y. (1999, 7-9 September 1999). *Intrusion Detection for an On-Going Attack.* Paper presented at the Recent Advances in Intrusion Detection - RAID'99, Purdue, IN, USA.