Case-Relevance Information Investigation: Binding Computer Intelligence to the Current Computer Forensic Framework

Gong Ruibin
Chan Kai Yun, Tony
School of Computer Engineering
Nanyang Technological University, Singapore

Mathias Gaertner
INI-Service Center
Fraunhofer Institut fuer Graphische Datenverarbeitung, Germany

Abstract

Computer Forensics has grown rapidly in recent years. The current computer forensic investigation paradigm is laborious and requires significant expertise on the part of the investigators. This paper proposes a highly automatic and efficient framework to provide the *Case-Relevance* information, by binding computer intelligence technology to the current computer forensic framework. Computer intelligence is expected to offer more assistance in the investigation procedures and better knowledge reuse and sharing in computer forensics.

Background

Cybercrime is a mirror of the dark side of human society in the cyberworld. Its countermeasure, *Computer Forensics*, also referred as *Digital Forensic Science*, has been explicitly defined as,

> The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations. [14]

The process of "identifying, preserving, analyzing, and presenting digital evidence in a manner that is legally acceptable via the application of computer technology to the investigation of computer based crime" is called *Forensic Computing* [11] or *Digital Evidence Investigation*.

As almost every piece of digital evidence could be challenged, computer forensic investigators are required to follow a rigorous process path. The work of the First Digital Forensics Research Workshop (DFRWS) [14] established a solid ground and allowed the experts and researchers from governments, law enforcement, and other third parties to work together under a unified schema. Some continued work can be found in [1,2,5,12,15,16].

A basic single tier framework of digital evidence investigation process (Figure 1) was depicted in [2], which consists of six phases, and each phase may include several sub-phases. Phases and sub-phases are "distinct, discrete steps in the process that are usually a function of time and suggest a necessarily sequential approach" [2]. Although the phases are sequential and non-iterative in the framework, the actual investigation process is highly iterative. That is, the findings in a certain phase may be fed back to the previous phases for a further refinement.

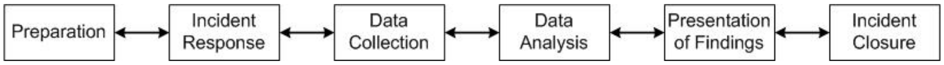

Figure 1: Single Tier Framework of Digital Evidence Investigation Process

Arguably, the Data Analysis Phase is the most complex phase in the whole investigation process and has drawn intensive attention from the researchers. In [14,16], it was divided into two separate examination and analysis phases. Also, two phases, "the live system processing and data collection" and "the analysis of secured data," were used in [12] to describe the operation flow of the overall data analysis part in a network forensic environment. In [5], the data analysis part was described as three first-tier phases, including *Survey Phase*, *Search and Collection Phase,* and *Reconstruction Phase*. In [2], a three sub-phases model including *Survey Sub-Phase*, *Extract Sub-Phase,* and *Examine Sub-Phase* was proposed as shown in Figure 2.

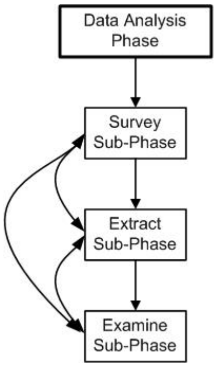

Figure 2: Three Sub-Phases Model of Data Analysis Phase

Although the exact definition and boundary of the sub-phases differ here and there in the above works, the objectives of the overall Data Analysis Phase is clear, that is, to exam, search and extract relevant data collected in the Data Collection Phase and to supply sufficient information for the crime scenario reconstruction and suspected activity confirmation.

Motivations

Today a lot of computer software systems [6] are available to help to carry out the tasks more effectively and more efficiently. Among them, some multi-functional systems, such as Encase [17], AccessData Forensic Toolkit [7] and Vogon [18] Forensic Software, offer an integrated environment for data capturing, imaging, searching, filtering, and analyzing. For example, eScript, a simple script language used in Encase, allows the investigators to write their own programs for customized data searching and filtering or perform a sequence of operations. Users can download, share and exchange the script codes, which bring some certain flexibility, reusability, automation and efficiency to the investigation process.

By taking an in-depth look into the data analysis phase of these systems, it is found that the major part of the information searching, extraction, and analysis work is still left to human. A typical evidence searching procedure often consists of a first round search starting with some initial clues. These clues could be some keywords in a text-based incident, some questionable log records in a computer intrusion matter, or some pornography images in a case involving child abuse. We call all these clues *Seed Information* because they are the start point of the investigation. The first round search is quite loose and usually returns dozens to hundreds of hits. An investigator checks the return, gets rid of the irrelevant items, and refines the query terms, if necessary. Once some evidence piece is found, the electronic file or the stream containing the evidence will be analyzed immediately to bring other clues to light. These new findings are used to build a new search list for the next round search. The whole bootstrapping procedure is highly iterative, trying to capture as much evidence as possible for the later scenario reconstruction, as shown in Figure 3. The data analysis phase is tedious, time-consuming, and requires significant expertise on the part of the investigator.

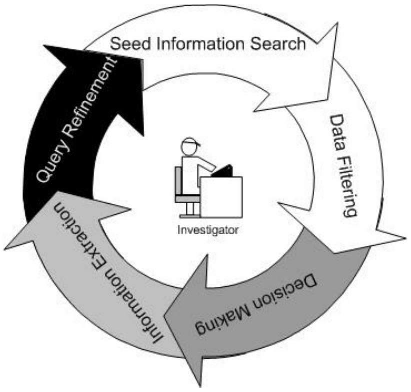

Figure 3: Searching/Extraction Loop

Another problem is *knowledge reuse*. For example the selection of the *Seed Information* needs very strong background knowledge. An experienced investigator usually maintains a collection of search lists from his previous cases. In a new case, he could build the search list based on this collection or even re-use one from a similar case directly. Some software allows the user to import and export the search list for a quick startup. The search lists may be shared and distributed within a restricted community. This. is some kind of primary information reuse, but far from enough. Problems still exist for, how does a freshman learn to search not only the word "bomb" but also "primer" and other jargons? What we need is a systematic mechanism for knowledge collection, management, sharing and reuse, offering decision support for the investigators.

To solve the problems mentioned above, we believe that computer intelligence *should* and *could* play a more active role in the data analysis phase of computer forensics. In the remainder of this paper, we will discuss how to bind computer intelligence into the current framework effectively and how it could benefit the current investigation procedure with higher automation, effectiveness and better knowledge reuse.

Case-Relevance Information Investigation

Despite the fact that computer forensics and computer security share a lot of tools and knowledge, there are significant differences between them. These have been discussed intensively in the literature. Parts of them are summarized by [3] as shown in Table1.

Security	Forensic Computing
Protects the system against attack	Does not protect the system against attack
Usually in real time	Post mortem
Conducted by computer specialists	Can be conducted by computer specialists, but often this is not the case
Restricted environments for presentation of developments, issues	Evidence is nearly always presented to non-IT/IS personnel
Can be bypassed by trusted individuals/users	Integrity of the evidence is most important

Table 1: Key Distinctions between Computer Security and Forensic Computing

In this paper, we would like to define a new concept, *Case-Relevance,* as:

> ***Case-Relevance:*** *the property of any piece of information, which is used to measure its ability to answer the investigative "who, what, where, when, why and how" questions in a criminal investigation.*

The degrees of *Case-Relevance* are assigned as shown in Figure 4. Absolutely Irrelevant refers to definitely no sign of the crime. Provably Case-Relevant means the information is undoubtedly critical to the criminal investigation. Actually, the degree of Case-Relevance covers a continuous spectrum from Absolutely Irrelevant to Provably

Case-Relevant, rather than simply relevant or irrelevant. But for the convenience of general discussions, it could be defined as a discreet set of degrees. We use *possible* and *probable* to describe the increasing levels of *Case-Relevance* or irrelevance. The degree of *Case-Relevance* provides the possibility to establish an effective framework for analyzing cost versus completeness.

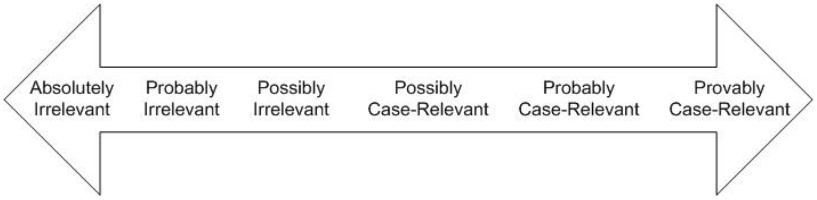

Figure 4: Degrees of *Case-Relevance*

We would like to use the concept of *Case-Relevance* to distinguish computer forensics from computer security for the following reason.

First, computer security is generalized, while computer forensics is strictly case restricted. While computer security is looking for any possible harmful behaviors, computer forensic investigators are interested in a very narrow scope - the information that could be used to reconstruct the criminal scenario. The ultimate target of a computer forensic system is to provide directly case relevant information. On this point, most results (e.g., an intrusion happened) from computer security systems are only intermediate data and need to be passed to a computer forensic system to check its relationship with the case. The concept of *Case-Relevance* could become an effective criterion to search, filter and organize all these data effectively.

Secondly, while the hackers are "willing to invest a lot more time looking for weaknesses to exploit than most of us are willing to spend implementing good security" [13], the computer security experts have to keep alert for twenty four hours a day for every possible exploit because they do not know when the hackers will break through. The computer forensic investigators are more fortunate on this point - more or less, "Every investigation starts with a preliminary analysis of the crime notification (*notitia criminis*) which leads to the formulation of some initial hypotheses that drive the evidence discovery process" [4]. The initial information can be used to build the *Case-Relevance* judgment and give an explicit direction for the later steps.

Thirdly, computer forensics has much more time restriction than computer security, although it is an after-event procedure. The restriction comes from the law and other pragmatic issues. For example., in the legal system in Germany, it often happens that a judge cancels an investigation only because it costs too much on either time or budget. The degree of *Case-Relevance* offers a great opportunity to rank the potential information according to their importance to the criminal investigation and allows the investigators to handle the most important parts within the limited time.

Case-Relevance is a high level concept, not a specific method that could be deployed directly. The ultimate target of a computer forensic system is to provide the information directly relevant to the case. Towards this target, all kinds of methods can be tried, guided by the concept of *Case-Relevance*. Binding computer intelligence into the computer forensic framework based on *Case-Relevance* will turn the current system into a target-oriented one without any redesigning work on the framework itself.

Case-Relevance Concept in Investigation Scenarios

The following discussions are restricted to a text-based environment; that is, all the data involved are in text format. The processing of the evidence in other formats could follow the same logic and working flow.

Case-Relevance Information Extraction

This is a common scenario in digital evidence investigation -- an exhaustive searching procedure to maximize the evidence availability and quality. A flowchart is shown in Figure 5. Similar to the Data Analysis Phase framework in [2], the procedure is also divided into three Sub-Phases.

1. <u>Survey Sub-Phase</u>*.* First, an experienced investigator studies the initial case information carefully to work out a Case Profile. Some intuitive computer-human interfaces may be used to offer some kinds of assistance. The Case Profile will be sent to an Expert System, behind which is a case database that keeps all the previous case records, to recommend the keywords for the first round search in the Extraction Sub-Phase.

2. <u>Extraction Sub-Phase</u>*.* The Automatic Evidence Extraction Module is a fully automatic bootstrapping procedure. It starts from few keywords and will retrieve more and more case relevant information in the iterative procedures until all evidences have been extracted. These few keywords are the *Seed Keywords*. The detailed block functions will be explained later in this subsection.

3. <u>Examination Sub-Phase</u>*.* The extracted information is examined by an experienced investigator. It is a basic principle that the critical decisions should be left to the human, no matter how smart a computer would evolve finally. The investigator can confirm or deny the findings, or return to the previous step for a refinement. The result will also be added into the Case Database and the Expert System. That is how the knowledge is accumulated and developed.

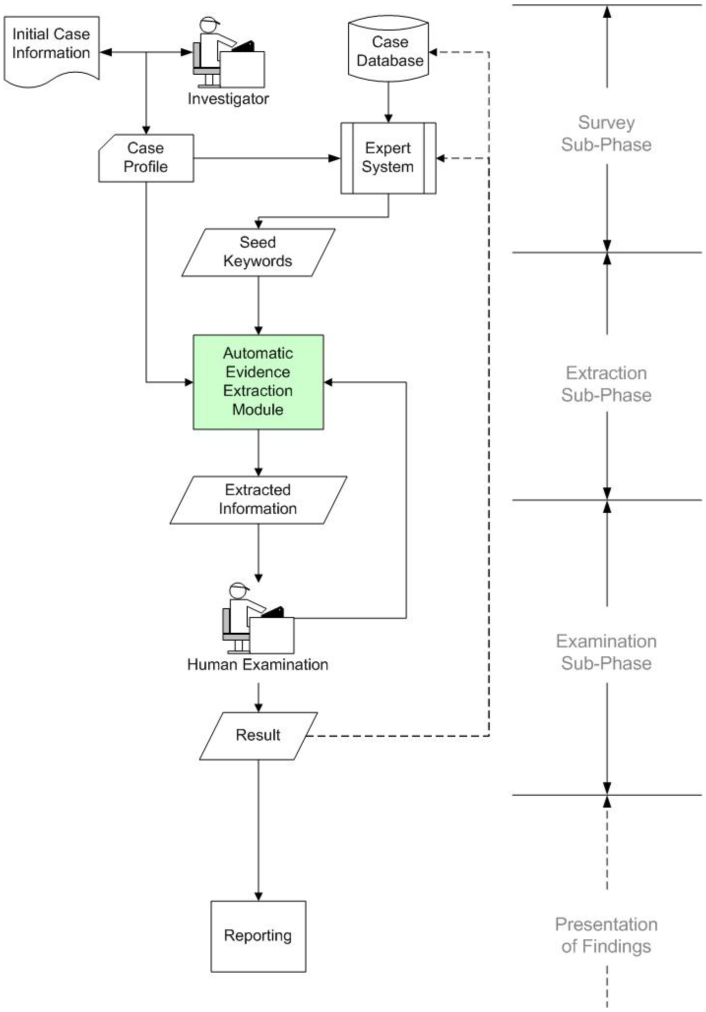

Figure 5: Case-Relevance Information Extraction

Automatic Evidence Extraction Module.

The Automatic Evidence Extraction Module is the core of the Case-Relevance Information Extraction scenario. Here we propose a hybrid architecture consisting of Information Retrieval (IR), Information Extraction (IE) and computer intelligence function blocks as shown in Figure 6.

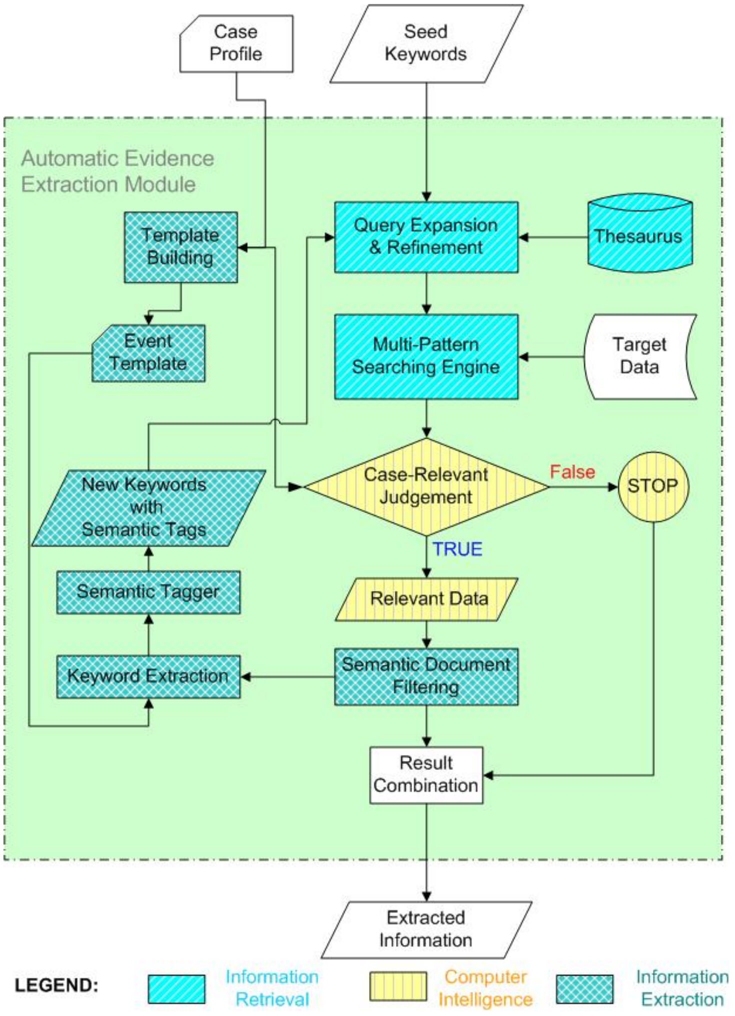

Figure 6: Automatic Evidence Extraction Module

1. Information Retrieval Block. The Seed Keywords are sent to the Query Expansion & Refinement Module. A predetermined concept-based thesaurus adds synonyms to the original query automatically. The thesaurus categorizes the patterns by their semantic concept and is very effective to control the number of query terms and improve the precision. The thesaurus is built and maintained by authorized experts from the previous case documents and other sources such as WordNet® [19]. Then the query terms are sent to a Multi-Pattern Searching Engine. The previous expansion step may increase the number of keywords dramatically and thus add the payload of the searching engine. A fast multiple patterns searching algorithm is preferred, and hardware-based architecture attracts enough attention by its intrinsic parallelism and speed advantage.

2. Computer Intelligence Block. A Case Relevant Judgment module scans the target data, makes a decision on the given Case Profile, and returns data ranked by their

degrees of *Case-Relevance*.

3. Information Extraction Block. The IE block can be divided into two parts: Template Building and Keyword Extraction. In the Template Building part, the event templates are automatically created based on the given Case Profile, and will be used in the Keyword Extraction part. This Keyword Extraction part has three functional modules. The first one is a semantic-level document filter, which permits the identification of relationships rather than the purely Boolean. Hence, topics and documents can be matched not only by whether the specified keywords occur in both, but by whether they occur in the same (or similar) relationship in both topic and document. The non-relevant documents are abandoned even if they have the same keywords. The second one is a keywords extraction module, which fills the Event Templates with the key information that may be used as new query terms, e.g. personal name, time and locations. The third module is a semantic tagger. It is based on the semantic-level analysis to disambiguate the concepts of the words or phrases in the given context and add corresponding tags to the selected keywords. The new keywords list will be sent to the IR block to start the next round of search. The added tags can be processed by the Concept-based thesaurus to produce accurate query terms.

Case-Relevance Information Confirmation Scenario

We will only discuss the scenario briefly in this paper. In a typical real situation, the investigator is asked to confirm or deny one activity. Instead of searching and bringing everything involved to the court, the investigator is expected to do a comparable narrow and highly target-oriented examination on the captured data. A three sub-phases working flowchart is shown in Figure 7, in which an Activity-Relevant Judgment module is used to scan the target data and return highly selective information for the activity confirmation only.

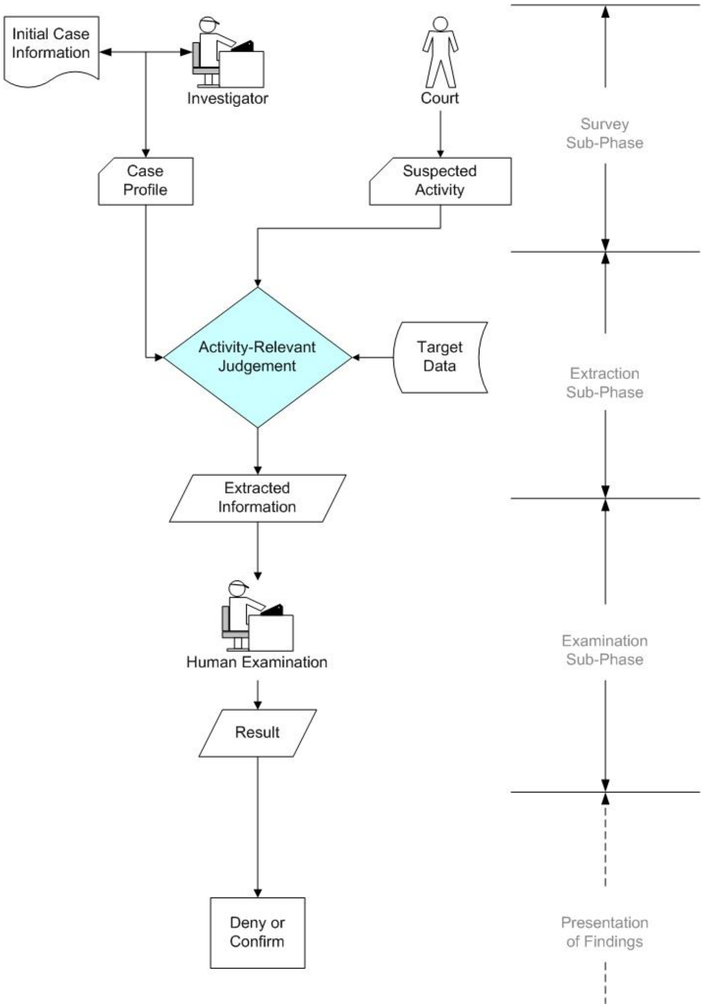

Figure 7: Case-Relevance Information Confirmation

Challenges

The method we propose requires a lot more work to reach its full potential. The challenges include but are not limited to the fields of Profiling, Case Matching, Expert System, Template Building and Text Mining.

We believe a major obstacle for the deployment of computer intelligence in computer forensics is the lack of Standard Test Datasets and Evaluation Criteria. Some notice [8] has been given to the formalization of the test and evaluation activities of different products. It is very urgent to establish a formal and repeatable test dataset and evaluation environment for the Data Analysis Phase. Computer intelligence is extremely computational intensive and needs large volume of data for training and testing.

Obtaining or simulating real case data for a stand-alone computer is not difficult, but fetching data from the network of a large organization or a large volume of legal cases is very complex and not applicable in many situations. Law enforcement agencies have the best-maintained document systems. But for reasons of security and privacy protection, these documents are not accessible to the research community. In Intrusion Detection, the researchers have the same problem of protecting the sensitive data, but they have already worked out some standard datasets and test beds and are still working to improve them, such as the work in [9,10]. We suggest adopting their methods to build and publish a standard dataset for computer forensics. Raw data from selected cases should be re-organized and filtered, adding some manually created events if needed; all the events should be carefully examined, located, categorized and listed, and the private or sensitive information should be removed.

Some open competitions such as the blind-test form in the Message Understanding Conference supported by DARPA are suggested to attract the interest from experts and researchers from governments, law enforcement, business companies and other third parties. These will boost the research and implementation in both computer forensics and computer intelligence.

Conclusion

Computer forensics will play an increasingly important role in criminal investigation. When examining the current progress of computer intelligence in computer forensics, we find it is lagging far behind many other research and implementation fields.
This paper proposes a method to bind computer intelligence to the current computer forensic framework, particularly to the data analysis phase. A high level concept, *Case-Relevance*, is defined to measure the importance of any information to a given case. The proposed framework demonstrates the benefits of computer intelligence technologies: automatic evidence extraction and knowledge reusability, resulting in great savings on human resources.

About the Authors

Gong Ruibin is currently a Ph.D candidature in Centre for Advanced Information Systems, School of Computer Engineering, Nanyang Technological University. His research interest includes computer forensics, computer security and information retrieval. Contact: gongrb@pmail.ntu.edu.sg.

Chan Kai Yun, Tony is concurrently an Associate Professor in the Division of Computing Systems, School of Computer Engineering, Nanyang Technological University and the Director of gameLAB. His teaching and research are in the areas of computer graphics and visualization, multimedia, microprocessor applications, parallel and distributed storage systems. Contact: askychan@ntu.edu.sg.

Mathias Gaertner is the project group leader of Service-Center, deputy CIO of

Fraunhofer-IGD and head of the Competence Center Lan-Management for the Fraunhofer-Society. Since March 2000, he started to work as an Expert witness (von der IHK Darmstadt öffentlich bestellter und vereidigter Sachverständiger für Systeme und Anwendungen der Informationstechnologie für den Bereich Netzwerktechnik) for IT and networking. Concurrently, he is also a lecturer at the University of Applied Science in Wuerzburg and the University of Applied Science in Darmstadt. Contact: mathias.gaertner@igd.fraunhofer.de.

References

[1]. Baryamureeba, Venansius and Florence Tushabe. The enhanced digital investigation process model. In *Digital Forensics Research Workshop (DFRWS)*, Baltimore, Maryland, August 2004.

[2]. Beebe, Nicole Lang and Jan Guynes Clark. A hierarchical, objectives-based framework for the digital investigations process. In *Digital Forensics Research Workshop (DFRWS)*, Baltimore, Maryland, August 2004.

[3]. Broucek, Vlasti and Paul Turner. Forensic computing: Developing a conceptual approach for an emerging academic discipline. In *The 5th Australian Security Research Symposium*, Perth, Australia, 2001.

[4]. Bruschi, Danilo, Mattia Monga, and Lorenzo Martignoni. How to reuse knowledge about forensic investigations. In *Digital Forensics Research Workshop (DFRWS)*, Baltimore, Maryland, August 2004.

[5]. Carrier, Brian and Eugene H. Spafford. Getting physical with the digital investigation process. *International Journal of Digital Evidence*, 2(2): 1-20, Fall 2003.

[6]. Forensic Focus. Computer forensics software, an introduction. http://www.forensicfocus.com/computer-forensics-software-intro.php, September 2004.

[7]. AccessData Forensic Toolkit (FTK). http://www.accessdata.com.

[8]. Hirsh, Mark. Proposal to formalize test and evaluation activities within the forensic and law enforcement communities. In *Digital Forensics Research Workshop (DFRWS)*, Baltimore, Maryland, August 2004.

[9]. Lippmann, Richard P., David J. Fried, Isaac Graf, Joshua W. Haines, Kristopher R. Kendall, David McClung, Dan Weber, Seth E. Webster, Dan Wyschogrod, Robert K. Cunningham, and Marc A. Zissman. Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition*, volume 2, 2000.

[10]. Lippmann, Richard P., Joshua W. Hainesand David J. Fried, Jonathan Korba, and Kumar Das. Analysis and results of the 1999 DARPA off-line intrusion detection evaluation. In *Proceedings of the Third International Workshop on Recent Advances in Intrusion Detection*, Lecture Notes In Computer Science, pages 162-182, 2000.

[11]. McKemmish, R. What is forensic computing. *Trends and Issues in Crime and Criminal Justice*, 118, 1999.

[12]. Mohay, George, Alison Anderson, Byron Collie, Oliview de Vel, and Rodney McKemmish. *Computer and Intrusion Forensics*. Artech House, Boston, 2003.

[13]. Northcutt, Stephen. *Network Intrusion Detection: An Analyst's Handbook*. New Riders Publishing, 1999.

[14]. Palmer, Gary. A road map for digital forensics research - report from the first Digital Forensics Research Workshop (DFRWS). Technical Report DTR-T001-01 Final, Air Force Research Laboratory, Rome Research Site, 2001.

[15]. Pollitt, Mark. A framework for digital forensic science. In *Digital Forensics Research Workshop (DFRWS)*, Baltimore, Maryland, August 2004.

[16]. Reith, Mark, Clint Carr, and Gregg Gunsch. An examination of digital forensic models. *International Journal of Digital Evidence*, 1(3): 1-12, 2002.

[17]. Encase Forensic Software. http://www.encase.com.

[18]. Vogon Forensic Software. http://www.vogon-international.com.

[19]. WordNet®. http://wordnet.princeton.edu.