available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/diin

ELSEVIER

Digital Investigation

Digital forensics research: The next 10 years

Simson L. Garfinkel

Naval Postgraduate School, Monterey, USA

Keywords:
Forensics
Human subjects research
Corpora
Real data corpus
Realistic data

ABSTRACT

Today's Golden Age of computer forensics is quickly coming to an end. Without a clear strategy for enabling research efforts that build upon one another, forensic research will fall behind the market, tools will become increasingly obsolete, and law enforcement, military and other users of computer forensics products will be unable to rely on the results of forensic analysis. This article summarizes current forensic research directions and argues that to move forward the community needs to adopt standardized, modular approaches for data representation and forensic processing.

1. Introduction

Digital Forensics (DF) has grown from a relatively obscure tradecraft to an important part of many investigations. DF tools are now used on a daily basis by examiners and analysts within local, state and Federal law enforcement; within the military and other US government organizations; and within the private "e-Discovery" industry. Developments in forensic research, tools, and process over the past decade have been very successful and many in leadership positions now rely on these tools on a regular basis—frequently without realizing it. Moreover, there seems to be a widespread belief, buttressed on by portrayals in the popular media, that advanced tools and skillful practitioners can extract actionable information from practically any device that a government, private agency, or even a skillful individual might encounter.

This paper argues that we have been in a "Golden Age of Digital Forensics," and that the Golden Age is quickly coming to an end. Increasingly organizations encounter data that cannot be analyzed with today's tools because of format incompatibilities, encryption, or simply a lack of training. Even data that can be analyzed can wait weeks or months before review because of data management issues. Without a clear research agenda aimed at dramatically improving the efficiency of both our tools and our very research process, our hard-won capabilities will be degraded and eventually lost in the coming years.

This paper proposes a plan for achieving that dramatic improvement in research and operational efficiency through the adoption of systematic approaches for representing forensic data and performing forensic computation. It draws on more than 15 years personal experience in computer forensics, an extensive review of the DF research literature, and dozens of discussions with practitioners in government, industry, and the international forensics community.

1.1. *Prior and related work*

Although there has been some work in the DF community to create common file formats, schemas and ontologies, there has been little actual standardization. DFRWS started the Common Digital Evidence Storage Format (CDESF) Working Group in 2006. The group created a survey of disk image storage formats in September 2006, but then disbanded in August 2007 "because DFRWS did not have the resources required to achieve the goals of the group. (CDESF working group, 2009)" Hoss and Carver discuss ontologies to support digital forensics (Carver and Hoss, 2009), but did not propose any concrete ontologies that can be used. Garfinkel introduced an XML representation for file system metadata (Garfinkel, 2009), but it has not been widely adopted.

E-mail address: simsong@acm.org

1742-2876/$ – see front matter © 2010 Digital Forensic Research Workshop. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.diin.2010.05.009

Richard and Roussev reviewed requirements for "Next-generation digital forensics." Their work stressed system requirements, and argued that inefficient system design, wasted CPU cycles, and the failure to deploy distributing computing techniques is introducing significant and unnecessary delays that directly translate into unnecessary delays (Richard and Roussev, 2006). Elements of a modular computer forensics system exist in both Corey et al.'s design of a network forensics analysis tool (Corey et al., 2002) and in Cohen's PyFlag (Cohen, 2008), although the rest of the DF research community has generally failed to appreciate how these architectures can satisfy Richard and Roussev's requirement for parallelism. Ayers ignored all of the previous work on this topic in his "second generation computer forensic analysis system," presented at DFRWS 2009 (Ayers, 2005). In general, it seems that very few DF systems designers build upon previous work—instead, each new project starts afresh.

Following the first DFRWS, Mocas proposed a framework to help build "theoretical underpinnings for digital forensics research (Mocas, 2004)." The purpose of the framework was to "define a set of properties and terms that can be used as organizing principles for the development and evaluation of research in digital forensics." Mocas suggested that research should consider context in which evidence is encountered, data integrity, authentication, reproducibility, non-interference and the ability of proposed techniques to comply with federal minimization requirements.

Pollitt reviewed 14 different models for digital forensics investigation but did not attempt to evaluate or catalog them given time constraints (Pollitt, 2007). Most of these investigation models rely on the ability to make the best use of digital evidence that is found. An alternative approach is *proactive digital forensics*—for example, Ray et al.'s design for a system that predicts attacks and changes its collection behavior *before* an attack takes place (Allen Ray, 2007). Bradford et al. likewise argue that it is unwise to depend upon "audit trails and internal logs" and the digital forensics will only be possible on future systems if those systems make proactive efforts at data collection and preservation; they present a mathematical model for deciding the content and frequency of proactive forensic event recorders (Bradford et al., 2004).

Pollitt et al. discussed how virtualization software and techniques can be productively applied to both digital forensics research and education (Pollitt et al., 2008). Any discussion of virtualization with respect to digital forensics faces an unwelcome tautology. In practice, the impact of virtualization on forensic examination can usually be ignored—except when it can't. That's because sometimes the virtualization is the subject of the forensic examination, and sometimes the virtualization is a tool it is used by the forensic examiner.

In June 2008 a brainstorming session at CISSE 2008 explored research categories, topics and problems in digital forensics. One of the results of this project was an article by Nance, Hay and Bishop that attempted to define a Digital Forensics Research Agenda (Nance et al., 2009). The authors identified six categories for digital forensics research: Evidence Modeling, Network Forensics, Data Volume, Live Acquisition, Media Types, and Control Systems. This taxonomy is useful, but believe that the tactical analysis must be accompanied by strategic thinking.

In January 2009 Beebe presented an invited talk at the Fifth IFIP WG 11.9 International Conference on Digital Forensics entitled "Digital Forensics: The Good, The Bad, and the Unaddressed (Beebe, 2009)." Beebe argued convincingly that digital forensics was no longer a niche discipline. "It is now mainstream knowledge that the digital footprints that remain after interactions with computers and networks are significant and probative. Digital forensics was once a niche science that was leveraged primarily in support of criminal investigations, and digital forensic services were utilized only during the late stages of investigations after much of the digital evidence was already spoiled. Now digital forensic services are sought right at the beginning of all types of investigations…Even popular crime shows and novels regularly incorporate digital evidence in their story lines."

As far as "The Bad" and "The Unaddressed," Beebe said that digital forensics largely lacks standardization and process, and what little widespread knowledge that we have is "heavily biased towards Windows, and to a lesser extent, standard Linux distributions." Unaddressed, Beebe says, is the problem of scalability, the lack of intelligent analytics beyond full-text search, non-standard computing devices (especially small devices), ease-of-use, and a laundry list of unmet technical challenges.

Finally, Turnbull et al. performed a detailed analysis on the specific digital media formats being collected by the South Australian Police Electronic Crime Section; theirs appears to be the first quantitative analysis of its kind (Turnbull et al., 2009), although the FBI's Regional Computer Forensic Laboratory program publishes an annual report with the amount of media and cases that it processes (Regional Computer Forensics Laboratory, 2008). More case studies such as these are needed so that researchers can use actual evidence, rather than their own personal experiences, to direct their problem-solving efforts.

2. Digital forensics: a brief history

Today DF is an important tool for solving crimes committed with computers (*e.g.* phishing and bank fraud), as well as for solving crimes against people where evidence may reside on a computer (*e.g.* money laundering and child exploitation). Forensic tools have also become a vital tool for Information Assurance because of their ability to reconstruct the evidence left by cyber attacks.

2.1. *The early days*

DF is roughly forty years old. What we now consider forensic techniques were developed primarily for data recovery. For example, Wood et al. relate a story about two local data recovery experts working for 70 h to recover the only copy of a highly fragmented database file inadvertently erased by a careless researcher (pp.123–124 Wood et al., 1987). By the late 1980s utilities were being widely advertised that could perform a variety of data recovering, including "Unformat, Undelete, Diagnose & Remedy"(p.57 Display ad 57, 1987).

These early days were marked by:

- Hardware, software, and application diversity.
- A proliferation of data file formats, many of which were poorly documented.
- Heavy reliance on time-sharing and centralized computing facilities; rarely was there significant storage in the home of either users or perpetrators that required analysis.
- The absence of formal process, tools, and training.

In these early days forensics was largely performed by computer professionals who worked with law enforcement on an *ad hoc*, case-by-case basis. Astronomer Cliff Stoll's foray into network forensics was one of the most celebrated examples of the time (Stoll, 1988, 1990).

There was also a limited need to perform DF. Evidence left on time sharing systems frequently could be recovered without the use of recovery tools. And because disks were small, many perpetrators made extensive printouts. As a result, few cases required analysis of digital media. The FBI started a "Magnetic Media Program" in 1984, but only performed examinations in three cases during its first year (CVJCTS, 2004).

Computer hacking was in the popular consciousness, as evidenced by the 1983 movie "WarGames." But prior to the passage of the Computer Fraud and Abuse Act of 1984, computer hacking was not even a crime, further limiting the need to subject systems to forensic analysis.

2.2. *The Golden Age of digital forensics*

The years from 1999–2007 were a kind of "Golden Age" for digital forensics. During this time digital forensics became a kind of magic window that could see into the past (through the recovery of residual data that was thought to have been deleted) and into the criminal mind (through the recovery of email and instant messages). Network and memory forensics made it possible to freeze time and observe crimes as they were being committed—even many months after the fact. Forensics became so widespread and reliable that it escaped from the lab and onto the TV screen, creating the so-called "CSI Effect." (Shelton, 2008)

This Golden Age was characterized by:

- The widespread use of Microsoft Windows, and specifically Windows XP.
- Relatively few file formats of forensic interest—mostly Microsoft Office for documents, JPEG for digital photographs; and AVI and WMV for video.
- Examinations largely confined to a single computer system belonging to the subject of the investigation.
- Storage devices equipped with standard interfaces (IDE/ATA), attached using removable cables and connectors, and secured with removable screws.
- Multiple vendors selling tools that were reasonably good at recovering allocated and deleted files.

The widespread dominance of the so-called "WinTel" platform meant that many digital forensics examiners could be successful knowing that system and no others. The widespread failure of the market to adopt encryption technology for data-at-rest (Garfinkel and Shelat, 2002) made it relatively easy to develop and sell forensic tools that were actually useful to a wide range of customers. These tools allowed someone with relatively limited training to search for email messages, recover deleted files and perform basic file carving.

The Golden Age was also marked by a rapid growth in digital forensics research and professionalization. Universities around the world started offering courses in DF. Today there are 14 schools offering certificate programs in DF, 5 schools offering associates degrees, 16 bachelor programs, 13 masters programs, and two doctoral programs, according to the Digital Forensics Association (2010).

2.3. *The coming digital forensics crisis*

Today much of the last decade's progress is quickly becoming irrelevant. Digital Forensics is facing a crisis. Hard-won capabilities are in jeopardy of being diminished or even lost as the result of advances and fundamental changes in the computer industry:

- The growing size of storage devices means that there is frequently insufficient time to create a forensic image of a subject device, or to process all of the data once it is found.
- The increasing prevalence of embedded flash storage and the proliferation of hardware interfaces means that storage devices can no longer be readily removed or imaged.
- The proliferation of operating systems and file formats is dramatically increasing the requirements and complexity of data exploitation tools and the cost of tool development.
- Whereas cases were previously limited to the analysis of a single device, increasingly cases require the analysis of multiple devices followed by the correlation of the found evidence.
- Pervasive encryption (Casey and Stellatos, 2008) means that even when data can be recovered, it frequently cannot be processed.
- Use of the "cloud" for remote processing and storage, and to split a single data structure into elements, means that frequently data or code cannot even be found.
- Malware that is not written to persistent storage necessitates the need for expensive RAM forensics.
- Legal challenges increasingly limit the scope of forensic investigations.

Today's examiners frequently cannot obtain data in a forensically sound manner or process that data to completion once they obtain it. Evidence, especially exculpatory evidence, may be routinely missed.

These problems are most obvious to examiners faced with cell phones and other mobile computing platforms. There are thousands of cell phone models in use around the world, with five major operating systems (Android, Apple, Blackberry, Windows Mobile, Symbian), more than a dozen "proprietary" systems, and more than 100,000 downloadable applications. There are dozens of "standard" cell-phone connectors and chargers.

It is vital for forensics examiners to be able to extract data from cell phones in a principled manner, as mobile phones are a primary tool of criminals and terrorists. But there is no standard way to extract information from cell phones. While some manufacturers are rumored to provide assistance to some law enforcement organizations, in many cases it is impractical to involve manufactures due to time, cost, or security concerns. The NIST *Guidelines on Cell Phone Forensics* recommends "searching Internet sites for developer, hacker, and security exploit information" when confronting a cell phone that is password-protected (National Institute of Standards and Technology, 2007).

Similar problems with diversity and data extraction exist with telecommunications equipment, video game consoles and even eBook readers. These last two pose the additional problem that techniques used by to protect their intellectual property also make these systems resistant to forensic analysis. Yet all of these systems have been used as the instrument of crimes and may contain information vital to investigations.

Our inability to extract information from devices in a clean and repeatable manner also means that we are unable to analyze these devices for malware or Trojan horses. For example, the persistent memory inside GPUs, RAID controllers, network interfaces, and power-management co-processors is routinely ignored during forensic investigations, even though it can be utilized by attackers.

The vast size of today's storage devices means that time-honored and court-approved techniques for conducting investigations are becoming slower and more expensive. Today a 2TB hard drive can be purchased for $120 but takes more than 7 h to image; systems and individuals of interest can easily have more storage than the police crime lab responsible for performing the analysis.

Encryption and cloud computing both threaten forensic visibility—and both in much the same way. No matter whether critical information is stored in an unidentified server "somewhere in the cloud" or stored on the subject's hard drive inside a TrueCrypt volume, these technologies deny investigators access to the case data. While neither technology is invincible, both require time and frequently luck to circumvent (Casey and Stellatos, 2008). Cloud computing in particular may make it impossible to perform basic forensic steps of data preservation and isolation on systems of forensic interest.

In recent years there has been substantial interest in RAM-based forensics to defeat encryption and to find malware that is not written to persistent storage. RAM forensics can capture the current state of a machine in a way that is not possible using disk analysis alone. But RAM DF tools are dramatically more difficult to create than disk tools. Unlike information written to disk, which is stored with the intention that it will be read back in the future—possibly by a different program—information in RAM is only intended to be read by the running program. As a result there is less reason for programmers to document data structures or conserve data layout from one version of a program to another. Both factors greatly complicate the task of the tool developer, which increases tool cost and limits functionality.

Although the market for DF tools appears to be growing, it continues to be dominated by relatively small companies that face extraordinarily high research-and-development costs. Product lifetimes are short because new developments in the marketplace must be tracked and integrated into tools, or else the tools become rapidly obsolete. A few commercial players heroically struggle to keep their products up-to-date, but their coverage of the digital systems in use today is necessarily incomplete.

Among digital forensics professionals, the best approach for solving the coverage problem is to buy *one of every tool on the market*. Clearly, this approach only works for well-funded organizations. Even then, there are many situations in which commercial tools fail and practitioners must rely on open source software. Although some of this software is very good, other tools are poorly documented, out-of-date, and even abandoned. Sadly, even though many professionals rely on open source tools, there is no recognized or funded clearing house for open source forensics software.

Training is a serious problem facing organizations that deliver forensic services. There is a lack of complex, realistic training data, which means that most classes are taught with either simplistic manufactured data or else live data. Live data cannot be shared between institutions, resulting in dramatically higher costs for the preparation of instructional material. As a result, many organizations report that it typically takes between one and two years of on-the-job training before a newly minted forensics examiner is proficient enough to lead an investigation.

Lastly, a variety of legal challenges are combining to make the very process of computer forensics more complicated, time consuming, and expensive. In US *v*. Comprehensive Drug Testing (Comprehensive Drug Testing, Inc, 2009) the Court wrote *dicta* that ran counter to decades of digital forensics practice and has dramatically limited the scope of federal warrant searches. Equally problematic is the international landscape. In the two decades since Stoll discovered the difficulty of executing an international wiretap order (Stoll, 1988), US law enforcement organizations have made significant strides in their ability to work with their foreign counterparts (Cerezo et al., 2007). Nevertheless, recent attempts by academics to unravel the economics and technology of large-scale botnets indicate that cyber-criminals remain the masters of international cooperation (Kanich et al., 2009).

Fortunately, the capabilities of DF rose to such great heights during the Golden Age that we have a long way to fall before DF becomes useless. After explaining why current approaches to DF research are not up to the task (Section 3), this paper proposes a new way for the research community to move forward (Section 4).

3. Today's research challenges

This section describes the landscape of today's computer forensic research activities. It starts with a discussion of the driving factors for today's computer forensic tools. It then discusses the "visibility and search" model employed by today's forensic tools. It finally argues that much of this resulting research is tactical reverse engineering that poorly integrates with existing tools and fails to create new models that could ultimately lower the cost of forensic research.

3.1. Evidence-oriented design

There are two fundamental problems with the design of today's computer forensic tools:

- Today's tools were designed to help examiners find specific pieces of evidence, not to assist in investigations.
- Today's tools were created for solving crimes committed against people where the evidence resides on a computer; they were not created to assist in solving typical crimes committed *with* computers or *against* computers.

Put crudely, today's tools were creating for solving child pornography cases, not computer hacking cases. They were created for finding evidence where the possession of evidence is the crime itself.

As a result of this bias, today's tools are poorly suited to finding information that is out-of-the-ordinary, out-of-place, or subtly modified. Today's tools can (sometimes) work with a case that contains several terabytes of data, but they cannot assemble terabytes of data into a concise report. It is difficult to use these tools to reconstruct a unified timeline of past events or the actions of a perpetrator. Such tasks are instead performed more-or-less manually when forensic tools are used for investigations, incident response, e-discovery, and other purposes.

Evidence-oriented design has limited both the tools' evolutionary path and the imagination of those guiding today's research efforts:

- The legitimate desire not to miss any potential evidence has caused developers to emphasize completeness without concern for speed. As a result, today there are few DF tools that can perform a useful five-minute analysis.
- The objective of producing electronic *documents* that can be shown in court has stunted the development of forensic techniques that could operate on data that is not readily displayed. For example, despite the interest in residual data analysis, there are no commercially available tools that can perform useful operations on the second half of a JPEG file. Indeed, it was only in 2009 that academics showed it was even *possible* to display the second half of a JPEG file when the first half is missing (Sencar and Memon, 2009).
- The perceived impermissibility of mixing evidence from one case with another has largely blocked the adoption of cross-drive analysis techniques (Garfinkel, 2006), even though cross-case searches for fingerprints and DNA evidence is now a vital law enforcement tool.

Today's tools must be re-imagined to facilitate investigation and exploration. This is especially important when the tools are used outside of the law enforcement context for activities such as cyber-defense and intelligence.

3.2. The visibility, filter and report model

Most of today's DF tools implement the same conceptual model for finding and displaying information. This approach may be terms the "Visibility, Filter and Report" model.

1. All of the data on the collected media is analyzed and made *visible* in a user interface. The visibility process typically consists of four specific steps:
   1.1. Data to be analyzed is viewed as a tree, with the root of the tree being a critical data structure from which all other data can be reached. Examples of roots include the partition table of a disk; the root directory of a file system; a critical structure in the kernel memory; or a directory holding evidence files.
   1.2. Starting at the root, metadata is recursively examined to locate all data objects. Examples of data objects include files, network streams, and application memory maps.
   1.3. Information regarding each data object is stored in a database. Some tools use in-memory databases, while others use external SQL databases.
   1.4. Some tools additionally use *carving* (Mikus, 2005) to locate data objects that cannot be reached from the root. Some tools recursively process carving results with step 1.3, while other tools will simply instantiate each carved object as a file that must then be manually processed.
2. The user is presented with a tabular display of all the data objects. Individual data objects can be explored.
3. The user can apply *filters* to shorten the display.
4. The user can perform *searches* for keywords, names, phone numbers, and other specific content.
5. Finally, the user generates a *report* about what was found and the process followed to find it. Most modern computer forensic tools assist with some aspect of the report writing.

This model closely follows the tasks are required for evidence-oriented design (Section 3.1). For example, the model allows the analyst to search for a specific email address, but does not provide tools for extracting and prioritizing all email addresses that may be present. Because files are recovered before they are analyzed, certain kinds of forensic analysis are significantly more computationally expensive than they would be with other models. While some processes can be automated using scripting facilities, automation comes only at great expenses and has had limited success. Finally, this model does not readily lend itself to parallel processing. As a result, ingest delays are *increasing* with each passing year.

3.3. The difficulty of reverse engineering

Many of today's DF engineering resources are dedicated to reverse engineering hardware and software artifacts that have been developed by the global IT economy and sold without restrictions into the marketplace. But despite the resources being expended, researchers lack a systematic approach to reverse engineering. There is no standard set of tools or procedure. There is little automation. As a result, each project is a stand-alone endeavor, and the results of one project generally cannot exchange data or high-level processing with other tools in today's forensic kit.

3.4. Monolithic applications

There is a strong incentive among a few specific vendors to deploy their research results within the context of all-in-one forensic suites or applications. These vendors largely eschew the tools-based philosophy of Unix and have instead opted to create applications that resembles Microsoft Office. This approach may simplify user training and promote product lock-in, but it also increases costs for the field as a whole.

Support for file systems, data formats, and cryptographic schemes is a competitive advantage for vendors and development teams. But when these capabilities are bundled into a single application it is not possible for end-users to easily mix-and-match these capabilities as operational requirements dictate.

3.5. *Lost academic research*

A considerable amount of digital forensics research worldwide is being performed at universities and funded by various organizations at the state and national level. The National Science Foundation, the National Institute of Justice and the National Institute of Standards and Technology have all funded digital forensics research. Several conferences and journals exist to publish the results of this work. Many forensic programs have thesis requirements, creating yet more research and potentially useful tools.

Yet despite this frenzy of forensic activity, there are relatively few cases of academic research being successfully transitioned to end users:

1. Academic researchers can distribute open source tools that can be directly used, but most end users lack the skills to download tools and use them.
2. Academic researchers can license their technology to a vendor, which then either sells the technology directly or incorporates it into an existing tool. It is difficult to find an instance of this happening.
3. Vendors can read and learn from academic papers, perhaps creating their own parallel implementations of the work presented. But after numerous discussions with vendors it has become clear that they are relatively uninformed regarding the current state of academic forensic research.

Transitioning any technology from academia to end users is often difficult, of course. But attention to technology transfer is especially important in forensics given the scale of the problem, the relatively small developer community, and the small budgets for tool development. We cannot afford to waste the technology that academia is developing.

4. A new research direction

DF research needs to become dramatically more efficient, better coordinated, and better funded if investigators are to retain significant DF capabilities in the coming decade.

Faced with growing complexity, the standard tool of the computer scientist is abstraction and modularization (Saltzer and Frans Kaashoek, 2009). Given the staggering amount and complexity of data faced by forensic practitioners, it would make sense for forensic researchers or users to demand standards for data and code interchange. The key to improving research is the development and adoption of standards for case data, higher-level data abstractions, and composable models for forensic processing.

4.1. *Forensic data abstraction*

Today there are only five widely used forensic data abstractions:

Disk images are archived and transferred as raw or EnCase E01 files.
Packet capture files in bpf (McCanne and Jacobson, 1993) format are used to distribute network intercepts.
Files are used to distribute documents and image.
File signatures are distributed as MD5 and SHA1 hashes.
Extracted Named Entities such as names, phone numbers, email addresses, credit card numbers, *etc.*, are distributed as ASCII text files or, in some cases, Unicode files. Named entities are typically used for *stop lists* and *watch lists*.

Efforts to develop new formats and abstractions have largely failed (CDESF working group, 2009). The DF community specifically needs to create a wide range of abstractions—standardized ways for thinking about, representing, and computing with information ranging from a few bytes to a person's lifetime data production. For example:

Signature metrics for representing parts of files or entire files, including n-grams, piecewise hashes, and similarity metrics.
File metadata *e.g.* Microsoft Office document properties, JPEG EXIF information, or geographical information.
File system metadata *e.g.* such as timestamps, file ownership, and the physical location of files in a disk image.
Application profiles *e.g.* the collection of files that make up an application, the Windows Registry or Macintosh plist information associated with an application, document signatures, and network traffic signatures.
User profiles *e.g.* tasks the user engages in, which applications the user runs, when the user runs them, and for what purpose.
Internet and social network information associated with the user, *e.g.* the collection of accounts that the user accesses, or user's Internet "imprint" or "footprint" (Garfinkel and Cox, 2009).

The lack of standardized abstractions and standardized data formats slows progress by forcing researchers to implement more parts of a system before they can produce initial results. Researchers are forced to spend more time acquiring and preparing data. It is harder to compare research products. And most importantly, the lack of interchange formats limits the ability to create tools that can inter-operate.

Digital Forensics XML (Garfinkel, 2009) can be used to interchange a wide range of forensic metadata. The XML representation can be created by the tools fiwalk and frag_find today, and is being considered as a "Data Carving Log" format for PhotoRec (Grenier, 2009) as well as other carving tools. Having both filesystem extraction tools and file carvers produce the same XML makes it possible to create a forensic processing pipeline that preserves semantic content while allowing later stages of the pipeline to be insensitive to the manner in which data is extracted by the earlier stages.

Having a single format supported by multiple carvers also makes it possible to cross-validate carvers, build a single "meta" file carver that logically combines the results of multiple carvers and carving algorithms, or perform extensive regression tests.

SQL is another tool that can be productively used by the forensics community. Instead of integrate through pipes and XML files, the tools can integrate through an a SQL database. Already some tools, such as FTK3, store case information in an SQL database. But exploiting SQL to its potential requires the adoption of standardized data models and schemas.

4.2. *Modularization and composability*

Similar to the lack of standardized data format is the lack of a standardized architecture for forensic processing.

Today forensic software is largely developed in C, C++, Java, perl, Python, and Guidance Software's proprietary EnScript language. The software is run on Microsoft Windows, Apple Macintosh, and Linux operating systems.

Faced with such diversity of development choices, other developer communities have created frameworks that enable cross-language, cross-platform development. Typically these frameworks include standardized processing models, cross-language APIs and straightforward data marshaling. Some of the best examples of such a framework include the Apache webserver module system, the Mozilla Firefox web browser, and the Eclipse development platform. No such framework exists for processing DF information.

For example, a framework might allow plug-ins for file systems, processing of sectors, IP packets, bytestream "objects" (*e.g.* files, TCP streams and extracted contents from archive files), timestamps, email address, proper names, and so on. The framework could have a variety of correlation subsystems including an object-based hierarchical store and a temporal events database. The output subsystem could allow any plug-in to emit structured data; this data could then be used for textual reports, interactive reports, visualizations, or to drive some kind of automated event system.

Plug-ins for a forensics computation framework should be based on a callback model. This model allows for the same plug-in to be used in single-threaded, multi-threaded, or multi-server implementation. Although SleuthKit (Carrier, 2005) and fiwalk.py (Garfinkel, 2009) provide for limited callbacks, their APIs need to be extended to include reporting and forensic "meet points" for correlation. Such an API could also allow for forensic modules to be used in both interactive and batch forensic tools.

Today PyFlag (Cohen, 2008), OCFA (Dutch National Police Agency, 2010) and DFF (Baguelin et al., 2010) are all examples of frameworks. Yet none of these systems provide the scale, hooks for experimentation, or workflow automation required for a research framework. Such a framework would lower research costs by allowing researchers to focus on the specific algorithm that they were creating, rather than forcing them to learn many details of digital forensics. A framework would further allow the same algorithms to be deployed on small handheld systems, multi-core desktops, large blade clusters with hundreds or thousands of computation nodes. A stable API, popularized by this framework, could be adopted by commercial products such as NetIntercept (Corey et al., 2002), EnCase (Guidance Software, 2007), and FTK (Access Data, 2005), allowing for the easy transition of technology from the research laboratory to the user community.

As indicated above, SQL can also be used as an integration framework. Extraction tools find forensic details and store it in the database; visibility tools search the database and display the results. The use of a database can also enable multi-user forensic tools. But using a SQL database in this manner requires more careful planning that a standardized schema. First, the software developer must make a commitment to store *all* relevant information in the database, since data stored in the filesystem may not be visible to all applications. Developers must preserve and maintain old fields of the database even when they are no longer used, because a plug-in may make use of them. SQL can also severely limit performance.

4.3. *Alternative analysis models*

A logical first step in constructing a modular forensic processing framework would be to create a system that implements the "Visibility, Filter and Report" model introduced in Section 3.2. But a properly factored framework could then be used to experiment with alternative processing approaches.

4.3.1. *Stream-based disk forensics*

An alternative processing model suggested by Roussev (Rosusev, 2006) is to process an entire disk image as a byte-stream, starting at the beginning and reading until the end. This approach eliminates the time that the drive head spends seeking and assures that no data on the disk will be left untouched, but it can require a significant amount of RAM in order to reconstruct the file system hierarchy or to determine file boundaries. Of course, it may be possible to recover a significant amount of useful information from the drive without building the hierarchy, as previous research has shown that most files of forensic interest are not fragmented (Garfinkel, 2007).

Stream-based disk forensics is clearly more important for hard drives than for SSD drives, which have no moving head to "seek." But even without a seek penalty, it may be computationally easier to scan the media from beginning to end than make a first pass for file-by-file recovery followed by a second pass in which the unallocated sectors are examined.

4.3.2. *Stochastic analysis*

Yet another model for forensic processing is to sample and process randomly chosen sections of the drive. This approach has the advantage of potentially being very fast, but has the disadvantage that small pieces of trace data may be missed.

4.3.3. *Prioritized analysis*

Prioritized analysis is a triage-oriented approach in which forensic tasks are sequenced so that the operator will be presented with critical information as quickly as possible. Today the only commercial system that implements this approach is I.D.E.A.L. Technology Corp.'s STRIKE (System for TRIaging Key Evidence). STRIKE is a handheld forensics platform designed to process digital media and display what is found on its touch-screen user interface as new information is encountered (I.D.E.A.L., 2010). Unfortunately, STRIKE has not been demonstrated widely to either academic or commercial customers.

4.4. *Scale and validation*

Scale is an important issue to address early in the research process. Today many techniques that are developed and demonstrated on relatively small data sets ($n < 100$) fail when they are scaled up to real-world sizes ($n > 10{,}000$). This is true whether n refers to the number of JPEGs, TB, hard drives or cell phones.

At the same time, researchers are failing to develop a range of techniques that don't work at a small scale but perform quite well when run in a data-rich environment. For this reason researchers must have access early in the development process to realistic large-scale corpora.

Forensic researchers and tool developers need to hold themselves to a level of scientific testing and reproducibility that is worthy of the word "forensic." New detection algorithms should be reported with a measurable error rate—ideally with both *false positive* and *true positive* rates reported. Many algorithms support one or more tunable parameters. In these cases the algorithms should be presented with *receiver operating characteristic* (ROC) curves graphing the true positive rate against the false positive rate (Fig. 1) for a variety of parameter settings. Finally, consistent with the US Supreme Court's *Daubert* ruling (Daubert v. Merrell Dow Pharmaceuticals, 1993), the research community should work to develop digital forensic techniques that produce reportable rates for error or certainty when they are run.

Sponsors, researcher advisers and reviewers need to insist that new algorithms be tested with significant data sets—larger than a few dozen documents chosen from the experimenter's own system. To satisfy this requirement the DF research community needs to create, maintain and use standardized forensic corpora (Lyle, 2008; Garfinkel et al., 2009).

4.5. *Moving up the abstraction ladder*

Given the ability to treat collections of data and metadata as self-contained objects and to treat advanced forensic processing across multiple drives and data streams as simple function calls, researchers will be able to move up the abstraction ladder. We will then be able to create a new generation of forensic techniques, tools and procedures to help address the coming digital forensic crisis. Specific opportunities include:

4.5.1. *Identity management*

Given that the vast majority of the work in digital forensics involves attributing results to individuals, we need approaches for modeling individuals in a manner that is both principled and computable. Such an abstraction would include representations for simple data elements like names, email addresses and identification numbers, but should also extend to ways for formally representing a person's knowledge, capabilities and social network. Work in this area will allow for improved Internet search, identity resolution and disambiguation, and ultimately the creation of systems that can identify likely suspects, appropriated accounts, and other types of anomalies.

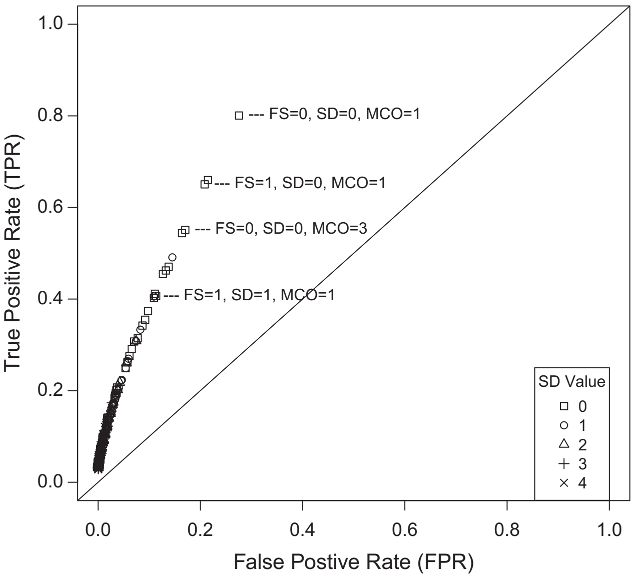

Fig. 1 – A receiver operating characteristic (ROC) curve for a forensic recognition experiment, showing how the true and false positive rates change when a set of tuning parameters are adjusted. This experiment attempted to distinguish random from compressed data by comparing the histograms of a block of data with the histogram of the autocorrelated block of data. The parameter FS ("First Slope") is the difference between the count of the most common element and the second-to-most common element. MCO is the Maximum Correlation Offset. SD is the Slope Difference between two autocorrelation tests. This visualization made it clear that the differentiation approach, while successful, presented an unacceptably high false positive rate and that the rate would not come down by adjusting the parameters. Ultimately a modified autocorrelation approach was followed and is presented in (DFRWS, 2010).

4.5.2. *Data visualization and advanced user interfaces*

Current tools use the standard WIMP model (window/icon/menu/pointing device), which are poorly suited to presenting large amounts of forensic data in an efficient and intuitive way. Research is needed to develop and adopt new approaches to visualizing and presenting forensic targets.

4.5.3. *Visual analytics*

Next-generation forensic tools need to integrate interactive visualization with automated analysis techniques, which will present data in new ways and allow investigators to interactively guide the investigation.

4.5.4. *Collaboration*

Since forensics is increasingly team effort, forensic tools need to support collaboration as a first class function. Additionally, new collaboration modes need to be discovered and implemented so that users can collaborate in real time, asynchronously, remotely and even on disconnected networks.

4.5.5. *Autonomous operation*

New, advanced systems should be able to reason with and about forensic information in much the same way that analysts do today. Programs should be able to detect and present outliers and other data elements that seem out-of-place. These systems will be able to construct detailed baselines that are more than simply a list of registry entries and hash codes for resident files.

Realistically, the only way that DF researchers and practitioners can cope with the challenges posed by the increasing diversity and size of forensic collections is to create more powerful abstractions that allow for the easier manipulation of data and the composition of forensic processing elements.

5. Conclusion

This paper predicts an impending crisis in digital forensics given a continuation of current trends that have been identified by many observers. But whereas other papers looking at the future of forensics have focused on specific tactical capabilities that need to be developed, this paper discusses the need to make digital forensics research more efficient through the creation of new abstractions for data representation forensic processing.

Given the diversity of the research and vendor communities, the development of these abstractions will not be sufficient to assure their success. Funding agencies will need to adopt standards and procedures that use these abstractions for the testing and validation of research products, and customers will need to demand that these abstractions be implemented in tomorrow's tools. With careful attention to cooperation, standardization, and shared development, the digital forensics research community can simultaneously lower development costs and improve the quality of our research efforts. This is probably one of the few techniques at our disposal for surviving the coming crisis in digital forensics.

Acknowledgments

George Dinolt, Joshua Gross, Elisabeth Rosenberg, Vassil Roussev, and the anonymous reviewers provided substantive comments on drafts of this article.

REFERENCES

Access Data. Forensic toolkit—overview, http://www.accessdata.com/Product04_Overview.htm?ProductNum=04; 2005.

Ayers Daniel. A second generation computer forensic analysis system. In: Proceedings of the 2009 Digital Forensics Research Workshop. DFRWS, http://www.digitalforensicssolutions.com/Scalpel/; August 2005.

Baguelin Frederic, Jacob Solal, Mounier Jeremy, Percot Francois. Digital forensics framework, http://www.digital-forensic.org/; 2010.

Beebe Nicole. Digital forensics research: the good, the bad, and the unaddressed. In: Fifth annual IFIP WG 11.9 international conference on digital forensics; January 2009.

Bradford Phillip G, Brown Marcus, Perdue Josh, Self Bonnie. Towards proactive computer-system forensics. In: Proceedings of the international conference on information technology: coding and computing (ITCCâ™04); 2004.

CDESF working group, 2009.

Carrier Brian. The sleuth kit and autopsy: Forensics tools for Linux and other Unixes, http://www.sleuthkit.org/; 2005 [accessed 06.03.09].

Carver DL Hoss AM. Weaving ontologies to support digital forensic analysis. 2009.

Casey Eoghan, Stellatos Gerasimos J. The impact of full disk encryption on digital forensics. SIGOPS Oper Syst Rev 2008;42 (3). ISSN: 0163-5980:93–8.

Cerezo Ana I, Lopez Javier, Patel Ahmed. International cooperation to fight transnational cybercrime. In: Second international workshop on digital forensics and incident analysis (WDFIA 2007); August 27–28 2007. p. 13–27.

Cohen MI. PyFlag: an advanced network forensic framework. In: Proceedings of the 2008 Digital Forensics Research Workshop. DFRWS, http://www.pyflag.net; August 2008 [accessed 06.03.09].

Corey Vicka, Peterman Charles, Shearin Sybil, Greenberg Michael S, Bokkelen James Van. Network forensics analysis. IEEE Internet Comput 2002;6(6). ISSN: 1089-7801:60–6.

Commonwealth of Virginia Joint Commission on Technology and Science. Regional computer forensic laboratory (rcfl) national program office (npo), http://jcots.state.va.us/2005%20Content/pdf/FBI-RCFL.pdf; September 8 2004.

Daniel Allen Ray. Developing a proactivedigital forensics system. PhD thesis, Tuscaloosa, AL, USA, 2007. Adviser-Bradford, Phillip G.

Display ad 57. The New York Times; February 8 1987.

Dutch National Police Agency. Open computer forensics architecture, http://ocfa.sourceforge.net/; 2010.

Digital Forensics Association. Formal education: college education in digital forensics, http://www.digitalforensicsassociation.org/formal-education/; 2010.

Daubert v. Merrell Dow Pharmaceuticals, 1993. 509 US 579.

Garfinkel Simson, Shelat Abhi. Remembrance of data passed. IEEE Security Privacy; January 2002.

Garfinkel Simson L. Forensic feature extraction and cross-drive analysis. In: Proceedings of the 6th annual digital forensic research workshop (DFRWS). Lafayette, Indiana: Elsevier, http://www.dfrws.org/2006/proceedings/10-Garfinkel.pdf; August 2006.

Garfinkel Simson L. Carving contiguous and fragmented files with fast object validation. In: Proceedings of the 7th annual digital forensic research workshop (DFRWS); August 2007.

Garfinkel Simson L. Automating disk forensic processing with sleuthkit, xml and python. In: Proceedings of the fourth international IEEE workshop on systematic approaches to digital forensic engineering. IEEE; 2009.

Garfinkel Simson L, Farrell Paul, Roussev Vassil, Dinolt George. Bringing science to digital forensics with standardized forensic corpora. In: Proceedings of the 9th Annual Digital Forensic Research Workshop (DFRWS); August 2009.

Grenier Christophe. Data carving log, http://www.cgsecurity.org/wiki/Data_Carving_Log; 2009.

Guidance Software, Inc. EnCase forensic, http://www.guidancesoftware.com/products/ef_index.asp; 2007.

Guidelines on cell phone forensics. Technical report. Gaithersburg, MD: National Institute of Standards and

Technology, http://csrc.nist.gov/publications/nistpubs/800-101/SP800-101.pdf; May 2007.

I.D.E.A.L. Technology Corporation. STRIKE (System for TRIaging Key Evidence), http://www.idealcorp.com/; 2010.

Kanich Chris, Kreibich Christian, Levchenko Kirill, Enright Brandon, Voelker Geoffrey M, Paxson Vern, Savage Stefan. Spamalytics: an empirical analysis of spam marketing conversion. Commun ACM 2009;52(9). ISSN: 0001-0782:99–107.

Lyle Jim. The CFReDS project, http://www.cfreds.nist.gov/; 2008.

McCanne Steven, Jacobson Van. The bsd packet filter: a new architecture for user-level packet capture. In: Proceedings of the USENIX Winter 1993 conference. Berkeley, CA, USA: USENIX Association; 1993. p. 2.

Mocas Sarah. Building theoretical underpinnings for digital forensics research. Digit Invest 2004;1:61–8.

Nance Kara, Hay Brian, Bishop Matt. Digital forensics: defining a research agenda. In: Proceedings of the 42nd Hawaii international conference on system sciences; 2009.

Nicholas Mikus. An analysis of disc carvingtechniques. Master's thesis, Naval Postgraduate School, March 2005.

Opinion by Chief Judge Kozinski. 11860 us v. Comprehensive Drug Testing, Inc, http://www.ca9.uscourts.gov/datastore/opinions/2009/08/26/05-10067eb.pdf; August 2009.

Pollitt Mark, Nance Kara, Hay Brian, Dodge Ronald C, p Craiger Phili, Burke Paul, Marberry Chris, Brubaker Bryan. Virtualization and digital forensics: a research and education agenda. J Digit Forensic Pract 2008;2(2). ISSN: 1556-7281:62–73.

Pollitt Mark M. An ad hoc review of digital forensic models. In: Proceedings of the second international workshop on systematic approaches to digital forensic engineering (SADFE'07); 2007.

Regional Computer Forensics Laboratory. Annual report for fiscal year 2008, 2008.

Richard III Golden G, Roussev Vassil. Next-generation digital forensics. Commun ACM 2006;49(2). ISSN: 0001-0782: 76–80.

Saltzer Jerome H, Frans Kaashoek M. Principles of computer system design: an introduction. Morgan Kaufmann; 2009.

Sencar Husrev, Memon Nasir. In: Identification and recovery of jpeg files with missing fragments, vol. 6; 2009, http://www.dfrws.org/2009/proceedings; 2009.

Shelton Donald E. The 'CSI Effect': does it really exist? NIJ J March 2008;259, http://www.ojp.usdoj.gov/nij/journals/259/csi-effect.htm.

Garfinkel Simson, Cox David. Finding and archiving the internet footprint. February 9–11 2009.

Stoll Clifford. Stalking the wily hacker. Commun ACM 1988;31(5). ISSN: 0001-0782:484–97.

Stoll Clifford. The cuckoo's egg: tracking a spy through the maze of computer espionage. Random House; 1990.

Turnbull Benjamin, Taylor Robert, Blundell Barry. The anatomy of electronic evidence â quantitative analysis of police e-crime data. In: International conference on availability, reliability and security, (ARES '09); March 16–19 2009. p. 143–9. Fukuoka.

Using purpose-built functions and block hashes to enable small block and sub-file forensics. In: DFRWS 2010, 2010.

Vassil Rosusev. Personal communication, 2006. group discussion at DFRWS 2006.

Wood Charles Cresson, Banks William W, Guarro Sergio B, Garcia Abel A, Hampel Viktor E, Sartorio Henry P. In: Garcia Abel A, editor. Computer security: a comprehensive controls checklist. John Wiley & Sons; 1987.