The Expanding World of Digital Forensics

SIMSON L. GARFINKEL

Simson Garfinkel was the program chair of the DFRWS 2015 Conference and researches digital forensics in Arlington VA. He is also an Affiliate Faculty Member of George Mason University in Virginia. simsong@acm.org

Digital forensics is on its way to becoming a mainstream part of computer science. With the right tools, forensic examiners can trace the source and determine the extent of a cyberattack. They can find evidence on a cell phone to help convict a bank robber or pimp. They can pull apart the firmware in cars and embedded devices, finding privacy leaks and security vulnerabilities. To accomplish this wizardry, the field has had to grow beyond its roots of simple data extraction and file system analysis, and now incorporates a wide variety of leading-edge computer science techniques, including big data analytics, visualization, multilingual processing, and program analysis. It's never been a better time to be a digital forensics researcher—the people who are charged with discovering how to exploit new technologies and building the tools that systemize that knowledge. At the same time, increased technical complexity and diversity is making the job of front-line forensics examiners more challenging every day.

Digital forensics is fragmenting. Ten years ago, it was common for examiners to be masters of the entire field—and perhaps developing their own tools as well. But as computer systems have become more complex, there's been an increasing need for examiners to specialize. As a result, having one or a few forensic specialists on staff is no guarantee that an organization can perform the necessary forensic tasks when the times arise. Instead, organizations increasingly rely on specialized teams that deeply practice and research a particular modality, then use partnerships to cover other forensic areas.

This article provides a brief overview of digital forensics as it is practiced today. I then present some of the recent advances in digital forensics research. Finally, I discuss the profound changes the field is likely to encounter over the next few years as a result of the growing attention that society is paying to privacy and security.

Digital Forensics Comes of Age

Modern digital forensics got going in the 1990s when law enforcement agents started encountering digital media during the course of criminal investigations. Some of these were classic examples of what we call cybercrime today—an outsider breaking into some kind of networked computer system, or an authorized user planting malware. Other cases were traditional crimes involving drugs, theft, or extortion, with the added twist that a computer system was used by the suspect to convey a threat, keep records, or communicate with co-conspirators.

The Internet's explosive growth in the late 1990s was accompanied by a similar increase in child pornography proliferation [1]. At the time, much of the public discussion focused on technologies for detecting and preventing criminals from downloading child porn over the network. But when law enforcement actually made an arrest, agents faced computers that needed to be examined. As the cases started to mount, agents at the FBI and other law

enforcement agencies realized that they needed standardized, repeatable, and accepted approaches for preserving digital evidence, making digital duplicates for use by examiners, searching for items of interest, and other aspects of the new digital laboratory practice [2].

To understand why standards were necessary, consider the example of deleted file recovery. Since the 1980s, programs like Norton Disk Doctor and Mace Utilities could "unformat" hard drives and "undelete" files that users accidentally deleted [3]. Law enforcement professionals were interested in such tools as well, since perpetrators would frequently delete files or otherwise try to hide their criminal activity if they suspected they were in danger of being apprehended. Evidence extracted from a suspect's own system proved to be incredibly useful for establishing guilt.

But it's one thing for a consumer to use an off-the-shelf program to recover a file that's been accidentally lost, another thing entirely for a law enforcement officer to recover a file on a suspect's computer and submit that file as evidence in a court of law. In the latter case, the officer needs some way to prove that the "deleted" file was really on the suspect's computer and not the result of contamination from another case or from running the recovery tool. To address these and similar concerns, practitioners developed techniques for both preserving and isolating case data. Eventually, the Federal Crime Laboratory Directors formed the Scientific Working Group on Digital Evidence (SWGDE) in February 1998 to help formalize the profession's standards [4].

Vendors responded by perfecting software that would reliably copy all of the data from a hard drive into a single "image file" (complete with checksums, case notes, and timestamps), and devices called "write blockers" that fit between a hard drive and a computer, allowing data to be copied off the drive but blocking attempts from the computer to overwrite sectors on the drive. Disk imaging proved to be more complex than first thought, as data could be hidden (and occasionally was hidden) in the "host protected area" or "device configuration overlay" of ATA-hard drives. Likewise, some hard drives contain bad blocks, and many imaging tools did not behave in a reliable and consistent manner when bad blocks were encountered [5]. Hardware write blocking proved necessary because many operating systems would overwrite sectors on a hard drive even when the drive was mounted "read-only." For example, some Linux "live CD" distributions will mount and use a Linux swap partition, potentially overwriting important evidence [6]. Write blockers also protected against an examiner's mistakes.

The aftermath of September 11, 2001, demonstrated that the power of digital forensics to recover deleted documents and report about a computer's past usage could be used for more than child exploitation cases. Although the hijackers are reported to have had no computers of their own, investigators searched computers in public libraries and copy shops frequented by the hijackers and discovered the systems had been used "to review and order airline tickets" used in the attack [7]. Computer records, including data found on the laptop of Zacarias Moussaoui, the alleged "20th hijacker," were featured prominently at Moussaoui's trial and significantly assisted the prosecution [8]. In the years that followed, digital forensics was widely used by coalition forces in Iraq and Afghanistan to gain intelligence from captured cell phones and laptops [9]. In May 2015, for example, the Office of the Director of National Intelligence released a trove of documents that had been seized as part of the 2011 raid on Osama bin Laden's compound [10].

Data extraction and file recovery remained primary goals of digital forensics researchers and developers in the decade following the 9/11 attacks. Smartphones increased the importance of digital forensics. Far more personal than a laptop, smartphones are often intimately involved in the planning and commission of crimes. Criminals use smartphones to communicate with their co-conspirators and with their victims (in the case of sexual assault) and even to document their crimes [11]. Unfortunately, it can be quite complicated to get the information out of a smartphone—unlike a hard drive, an examiner can't simply connect the phone to a "write blocker" and copy out all of the data. But once a phone's memory is dumped, the use of SQLite databases, JSON data structures, and text files made phone content relatively straightforward for an examiner to understand.

Many forensic processes designed for copying and analyzing data from simple IDE hard drives have been adopted for RAID arrays, SSDs, digital cameras, GPS devices, mobile phones, and an increasingly dizzying array of devices. There have been challenges. For example, many modern systems do not statically preserve data in the same way that magnetic drives do—SSDs that implement the TRIM command will slowly clear the blocks associated with deleted files if the drive is powered up, without any help from the operating system. This requires changing not just operating procedures, but underlying assumptions about the nature and goals of digital forensics.

Today there are mature commercial and open source tools available for digital forensics practitioners. Programs like EnCase, FTK, X-Ways Forensics, and Autopsy allow an examiner to view the contents of a disk image, perform keyword searches, and even recover deleted files. These tools typically support a range of file systems, including Windows FAT, XFAT and NTFS, Macintosh HFS+, and Linux EXT 2/3/4. A similar set of tools from companies like Celebrite and NowSecure provide these functions for iOS and Android-based phones. These tools implement a model I call "visibility, filter and report." First, they find all of the forensically interesting data on the media being

analyzed and make them visible, typically by putting them in a database and displaying a section of that database in a graphical user interface. Next, they allow the examiner to filter the data according to rules. Finally, the tools produce reports such as timelines, indications of known malware, and user activities. These tools have proven to be remarkably effective in helping examiners perform a wide variety of functions, from malware analysis to crime fighting.

Handling of non-English text has become increasingly important as examiners routinely encounter text in other languages. Early forensic tools simply could not display many non-US languages. Today, support for UNICODE is uneven but improving, and some tools are beginning to incorporate machine translation, allowing examiners to get the sense of a case without having to bring in a human translator.

Forensics has also grown beyond analyzing data at rest. Consider network forensics. Fifteen years ago, civil libertarians were outraged over the FBI's development and use of a network wiretapping program called "Carnivore" [12]. But at roughly the same time, network examiners were developing an open source network forensics program called Ethereal, now known as Wireshark. Unlike Carnivore, Wireshark can decrypt SSL-encrypted traffic (provided that the examiner has a copy of the server's private key). It's a great tool for "forensicating" networks, as the pros call it. Meanwhile, there's a whole new generation of malware analysts who consider digital forensics to be the study of opcodes, execution paths, and trust elevation exploits.

Digital Forensics Research

"Digital forensics" has become an umbrella term for any systematic examination of digital artifacts, code and data, no matter where they may be formed. And as the field has grown and matured, so too has the need for systematic research into the processes that create those digital artifacts and techniques for helping examiners to make sense of the massive amount of information that systematic examination produces.

The idea of digital forensics as an area of academic or industrial research dates to August 2001, when the Air Force Research Laboratory sponsored a two-day workshop in Utica, New York, on Digital Forensics Research. The results of that workshop were a 42-page report, "A Roadmap for Digital Forensics Research," and the annual Digital Forensics Research Workshop, later renamed "DFRWS."

Much of the past 15 years of research has been devoted to understanding the nature of stored data. In practice, there is little publicly available documentation for the vast majority of systems that have been deployed. An added complication is that many vendors have made their own changes to the data structures used by various operating systems and applications, sometimes in an attempt to get better performance from these systems, other times because they were not interested in maintaining compatibility with their competitors' systems. But even when software is open source, there is a big difference between having a copy of a program's source code and being able to understand the information that a program writes into a file, in a database, or on a disk.

Consider the case of SQLite, the cross-platform open source database that's widely used on mobile phones and desktop computers. Even though SQLite's code has been public since its initial release in May 2000, it's only in the past few years that forensic examiners have understood how to recover data that's been deleted in SQLite databases or partially overwritten database files. The reason: simply having a program's code may give insight into how the programs run, but the only way to really understand the data that a complex program produces is to methodically trace the program's execution and painstakingly track the output. In hindsight, this is just another application of the Church-Turing thesis.

Likewise, even though Microsoft has published specifications that describe FAT file systems in great detail and there have been open source implementations for nearly two decades, neither explains how to recover deleted files, or how to carve FAT32 directories from a drive that was reformatted by a computer running Windows XP, a version of Windows for which formatting wiped the root directory of the drive but did not overwrite most of the drive's actual storage.

Researchers have also spent considerable effort understanding the internal structure of various file formats. Simply trying to identify a file's "type" and extract the file's "text" proved to be a difficult problem in many cases. Although identifying a Microsoft Word document is relatively easy, there are thousands of different word processors, graphics tools, and financial programs in use. In many cases different versions of these programs write files with subtly different formatting. Since it's frequently not practical for examiners to acquire and run the precise version of the software that was used to create a file, other approaches need to be developed for understanding formats and extracting their text. Understanding these formats has another benefit as well: frequently, files contain hidden information that the normal end-user application doesn't show. Such information can be used to gain additional insight or intelligence about a crime or criminal organization.

Some of the most interesting data analysis work involves the reconstruction of files when some information is overwritten or missing but other binary data remains. For example, file carving is an approach for extracting files from media based on their content, rather than using file system metadata. Early file carvers could find and identify JPEGs by searching for the two

characteristic bytes that start and end every JPEG file (hex FF D8 and FF D9, respectively), then saving the header, the footer, and all of the bytes between into a new file. This approach, called header-footer file carving, generated a large number of false positives. Researchers discovered ways for discarding the invalid JPEGs. Researchers have since developed techniques for reassembling fragmented JPEG files [13], and even for rendering a fragment of a JPEG photograph when large parts from the file's beginning and end are missing and all that's available are the blocks in the middle [14]. These techniques need to be extended to the multitude of video formats.

Another important research area is memory analysis. Tools like the open source Volatility memory analysis framework [15] provide building blocks for understanding memory dumps, converting virtual memory addresses to physical, and decoding many of the operating system structures. Plugins written for Volatility can reconstruct the process list and find rootkits residing on Windows, Macintosh, and Linux systems. The contents of the Windows clipboard can be extracted and printed, open files can be displayed, and typed command lines can be found and recovered. Such tools have been tremendously important for incident response. Unfortunately, they are incredibly expensive to develop, creating the need for new development approaches.

Malware analysis has also become hugely important—so much so that it is largely its own area with specialty tools like Ida Pro and OllyDbg. Most malware analysis is done manually, with tools performing disassembly, search, and clerical support, but there is growing work in techniques that are largely automatic, relying either on static analysis or else running malware in environments that are highly instrumented.

Data extraction is still an important research area, although these days the hard problems are overcoming encryption and dealing with the data volumes of modern storage systems. For example, at the 15th DFRWS, which concluded August 2015 in Philadelphia, the Best Paper Award went to a pair of researchers who developed a technique for selectively imaging a small portion of a hard drive while still acquiring information necessary to solve a case [16]. Demonstrating the increased emphasis on scientific method and validation in the world of forensics, the paper characterized the accuracy of the technique with a synthetic publicly available data set [17] and on an actual case involving employee misconduct. Such techniques are soon to be extended to mobile devices and the cloud.

Privacy Cuts Both Ways

The public's increasing concern about digital privacy is increasingly a strong motivator and a growing barrier to digital forensics. A significant amount of research is now the result of work of researchers in related fields using current digital forensics tools and developing new approaches to perform privacy assessments of computers, mobile devices, and embedded systems. What they do with the results of these assessments will determine the future of the field.

For example, at the October 2015 ACM Conference on Computer and Communications Security, a pair of papers from researchers at Purdue explore sensitive information left in the memory of Android mobile phones. In one paper, the authors show that they can recover images that were taken with the camera but never stored in the phone's flash memory, such as frames from a Skype call or preview images from the camera [18]. In the second paper, the same authors show that they can recover Android GUIs from memory fragments that haven't been cleared [19]. These two papers rely on existing forensic techniques to image the phones' memory and provide new techniques that might be hugely useful in criminal investigations. On the other hand, these papers, and others in the same vein, provide developers with roadmaps of privacy leaks that need fixing—and in so fixing them, removing the possibility that the leaks might be used in future forensic examinations. In the past, forensics researchers typically did not widely publicize their findings for fear that vendors would fix the very privacy bugs that were helping to put criminals in prison.

Tool developers are actively searching for approaches that will let a forensics examiner visualize the massive amounts of data that a typical examination can recover. Examiners need tools that can automatically construct activity timelines, digest documents, and summarize video. These approaches need to automatically adjust themselves as the data scales multiply by orders of magnitude—from a few hundred photos that might be on a person's cell phone, to a few million that might reside on a server in a datacenter.

To leverage the attention of human examiners, some of the most important work being done on the algorithmic front is to identify new similarity techniques. The idea is to have digital forensics tools reliably find and cluster documents, photographs, and movie clips that are similar, so that examiners can spend their time looking at objects that are different from what's been seen before. Once clustered, other techniques like random sampling and machine learning could be used to characterize the variety.

These tools that can digest huge amounts of data are beginning to raise the concerns of civil liberties activists—just like in the days of Carnivore—who say that forensic capability needs to be weighed against privacy concerns. As a result, some jurisdictions are actively limiting the extent that examiners are allowed to search on a suspect's computer.

Meanwhile, forensic examiners are faced with the growing proliferation of devices, operating systems, and data formats. Examiners frequently master one kind of device just as something new

shows up on the market. That keeps the job interesting, but it means that the tools and technologies used by the examiners are almost never current, and developers are constantly struggling to keep up. In the fast-changing world of digital forensics, what's needed most are not ways for automating forensic analysis, but faster ways for doing digital forensics research and deploying new tools.

References

[1] Kathryn C. Seigfried-Spellar, Gary R. Bertoline, and Marcus K. Rogers, "Internet Child Pornography, U.S. Sentencing Guidelines, and the Role of Internet Service Providers," in Gladyshev and Rogers (eds), *Digital Forensics and Cyber Crime: Third International ICST Conference*, ICDF2C 2011, Dublin, Ireland, October 2011.

[2] Carrie Morgan Whitcomb, "An Historical Perspective of Digital Evidence: A Forensic Scientist's View," *International Journal of Digital Evidence*, vol. 1, no. 1 (Spring 2002): https://utica.edu/academic/institutes/ecii/publications/articles/9C4E695B-0B78-10593432402909E27BB4.pdf.

[3] Peter McWilliams, "Mace Utilities Can Recover Disk Data That Drives Away," *Chicago Tribune*, February 1, 1987: http://articles.chicagotribune.com/1987-02-01/business/8701080849_1_hard-diskfiles-unformat.

[4] Scientific Working Group on Digital Evidence: https://www.swgde.org/.

[5] James R. Lyle and Mark Wozar, "Issues with Imaging Drives Containing Faulty Sectors," *Digital Investigation,* vol. 4 (September 2007), pp. 13–15: doi:10.1016/j.diin.2007.06.002.

[6] Ahmed Fathy Abdul Latif Mohamed, Andrew Marrington, Farkhund Iqbal, Ibrahim Baggili, "Testing the Forensic Soundess of Forensic Examination Environments on Bootable Media," *Digital Investigation*, vol. 11 (2014), S22–29: http://www.dfrws.org/2014/proceedings/DFRWS2014-3.pdf.

[7] "9/11 Hijackers Used Public Libraries," *The Washington Times*, April 28, 2005: http://www.washingtontimes.com/news/2005/apr/28/20050428-115527-9817r/.

[8] Copies of those computer records, including a chilling letter from M. Atta requesting information about pilot training in the United States, can be found at http://www.vaed.uscourts.gov/notablecases/moussaoui/exhibits/prosecution.html.

[9] Stephen Pearson and Richard Watson, *Digital Triage Forensics: Processing the Digital Crime Scene*, Syngress Publishing, 2010.

[10] Documents that had been seized as part of the 2011 raid on Osama bin Laden's compound: http://www.dni.gov/index.php/resources/bin-laden-bookshelf.

[11] Dana Hedgpeth, "'Fire Selfies' after a Jealous Rage Lead to Maryland Man's Arrest," *The Washington Post*, September 1, 2015: https://goo.gl/nOq9x6.

[12] For a full description of the Carnivore program, including more than a thousand pages of documents obtained under the Freedom of Information Act, see https://epic.org/privacy/carnivore/.

[13] Anandabrata Pal, Husrev T. Sencar, Nasir Memon, "Detecting File Fragmentation Point Using Sequential Hypothesis Testing," *Digital Investigation*, vol. 5 (2008), S2–S13.

[14] Jusrev T. Sencar and Nasir Memon, "Identification and Recovery of JPEG Files with Missing Fragments," *Digital Investigation*, vol. 6 (2009), S88–S98.

[15] The Volatility Foundation, Volatility Memory Forensics Framework: http://www.volatilityfoundation.org/.

[16] Jonathan Grier and Golden G. Richard III, "Rapid Forensic Imaging of Large Disks with Sifting Collectors," *Digital Investigation*, vol. 14 (2015), S34–44.

[17] S. Garfinkel, P. Farrell, V. Roussev, and G. Dinolt, "Bringing Science to Digital Forensics with Standardized Forensic Corpora," *Digital Investigation*, vol. 6 (2009), S2–11.

[18] Brendan Saltaformaggio, Rohit Bhatia, Zhongshu Gu, Xiangyu Zhang, Dongyan Xu, "VCR: AppAgnostic Recovery of Photographic Evidence from Android Device Memory Images," ACM CCS, October 2015.

[19] Brendan Saltaformaggio, Rohit Bhatia, Zhongshu Gu, Xiangyu Zhang, Dongyan Xu, "GUITAR: Piecing Together Android App GUIs from Memory Images," ACM CCS, October 2015.