NIST Internal Report
NIST IR 8354

Digital Investigation Techniques: A NIST Scientific Foundation Review

James R. Lyle
Barbara Guttman
John M. Butler
Kelly Sauerwein
Christina Reed
Corrine E. Lloyd

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8354

NATIONAL INSTITUTE OF
STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Internal Report
NIST IR 8354

Digital Investigation Techniques: A NIST Scientific Foundation Review

James R. Lyle
Barbara Guttman
Software and Systems Division

John M. Butler
Kelly Sauerwein
Christina Reed
Corrine E. Lloyd
Special Programs Office

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8354

November 2022

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

NIST IR 8354
November 2022

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST Technical Series Policies

NIST Technical Series Publication Identifier Syntax

Publication History
Approved by the NIST Editorial Review Board on 2022-10-19

How to Cite this NIST Technical Series Publication
James R. Lyle, Barbara Guttman, John M. Butler, Kelly Sauerwein, Christina Reed, Corrine E. Lloyd (2022) Digital Investigation Techniques: A NIST Scientific Foundation Review. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Internal Report (IR) NIST IR 8354. https://doi.org/10.6028/NIST.IR.8354

NIST Author ORCID iDs
James R. Lyle: 0000-0001-8838-1941
Barbara Guttman: 0000-0001-9706-389X
John M. Butler: 0000-0001-6472-9157
Kelly Sauerwein: 0000-0001-9855-3030
Christina Reed: 0000-0002-4881-1465
Corrine E. Lloyd: 0000-0003-4575-2655

Abstract

This document is an assessment of the current scientific foundations of digital forensics. We examined descriptions of digital investigation techniques from peer-reviewed sources, academic and classroom materials, technical guidance from professional organizations, and independently published sources. Digital investigation techniques are based on established computer science methods and when used appropriately are considered reliable. The process of evaluating, for example, the contents of a computer hard drive does not create information that was not there before the investigation started. However, because the field is rapidly changing, there are limitations that practitioners and stakeholders need to be aware of: (1) as with any crime scene not all evidence may be discovered; (2) when recovering deleted files, the results may include extraneous material; (3) examiners need to understand that as software (operating systems and applications) is revised the meaning and significance of digital artifacts created by different versions of the software can be different.

In addition, because there are often multiple ways to search for information, two examiners may find different subsets of all potentially relevant information. The methods used in digital investigations are often not peer-reviewed in a formal process, but trustworthiness is established by members of the digital forensic community trying out proposed methods, testing, and circulating updates within the community. This process strengthens an examiner's awareness of the capabilities and limitations of their techniques.

Keywords

digital forensics, digital evidence, computer forensics, digital investigation, scientific foundations.

Table of Contents

List of Tables

List of Figures

Preface

Forensic science plays a vital role in the criminal justice system by providing scientifically based information through the analysis of physical and digital evidence. The National Institute of Standards and Technology (NIST) is a non-regulatory scientific research agency within the U.S. Department of Commerce with a mission to advance measurement science, standards, and technology and has been working to strengthen forensic science methods for almost a century. In recent years, several scientific advisory bodies have expressed the need for reviews of the scientific basis of forensic methods and identified NIST as an appropriate agency for conducting them. A scientific foundation review, also referred to as a technical merit evaluation, is a study that documents and assesses the foundations of a scientific discipline, that is, the trusted and established knowledge that supports and underpins the discipline's methods. Congress has appropriated funds for NIST to conduct scientific foundation reviews in forensic science. These reviews seek to answer the question: "What established scientific laws and principles as well as empirical data exist to support the methods that forensic science practitioners use to analyze evidence?" Background information on NIST scientific foundation reviews is available at https://doi.org/10.6028/NIST.IR.8225.

Acknowledgments

Gregory Haber and Will Guthrie of the NIST Statistical Engineering Division performed the capture recapture analysis in Sec. 2.7.2. Helpful feedback was provided by Simson Garfinkel, Eoghan Casey, Brandon Epstein, Bill Eber, and Tracy Walraven. The Scientific Working Group on Digital Evidence and the Organization of Scientific Area Committees for Forensic Science Digital Evidence Subcommittee served as valuable sources of information about current practices in digital forensics. Members of the NIST Computer Forensics Tool Testing team Rick Ayers, Craig Russell, Jenise Reyes-Rodriguez, and Mary Laamanen provided needed expertise.

Executive Summary

Every interaction with a digital device has the potential to leave a trail of what we did, who we did it with, where we were, and when some event took place. This trail is made up of digital artifacts, which are created in the routine operation of a digital device. This trail can assist an investigator to discover and explain what happened. Computers generate many artifacts, most of which do not contribute to understanding what happened. The challenge is finding useful information and separating it from irrelevant information. Digital investigation techniques can extract this information and construct a narrative of the events. The analysis of digital devices for investigative purposes is widely practiced and, as this report shows, in at least 11,000 digital forensic laboratories in the United States.

In recent years, several scientific advisory bodies have expressed the need for scientific foundation reviews of forensic disciplines and identified NIST as an appropriate agency for conducting them. The purpose of a scientific foundation review is to document and consolidate information supporting the methods used in forensic analysis and identify knowledge gaps where they exist [3]. In addition to this report on digital investigation techniques, the initial scientific foundation reviews conducted by NIST include DNA mixture interpretation, bitemark analysis, and firearm examination.

To address the question of the scientific basis of digital investigation, NIST examined the scientific literature on digital forensics as well as information from multiple other sources (see Sec. 2.8 and Sec. 3). The examination was limited to the technical validity of the techniques used in digital forensics without considering whether the techniques are used properly by forensic examiners. The review was led by a senior computer scientist and a multidisciplinary team from various areas at NIST. The team studied seven categories of digital forensic activities.

Obtaining input from experts outside of NIST is an integral component of a NIST scientific foundation review. As described in Chapter 3, the NIST team followed the process outlined in NISTIR 8225 [3] for conducting this review in terms of obtaining input from the community including:

- collecting and evaluating the peer-reviewed literature relating to digital forensics,
- assessing publicly available data from interlaboratory studies, proficiency tests, and laboratory validation studies,
- exploring other available information including position statements and non-peer reviewed literature, and
- obtaining input from members of the relevant digital forensics community through interviews, workshops, working groups, and other formats for the open exchange of ideas and information.

The overall finding of this report is that digital evidence examination rests on a firm foundation based in computer science. Several of the techniques had already been extensively studied and documented in the peer-reviewed literature. Others are documented more informally through community discussion forums. The application of these computer science techniques to digital investigations is sound, only limited by the difficulties of keeping up with the complexity and rapid pace of change in IT.

There are many ways to organize tasks performed in digital investigations. For this report, the following grouping of tasks is used:

1. Protect original data from unintended modification. This is accomplished using a variety of approaches depending on the type of device that contains the data. This is discussed in Sec. 4.1.
2. Acquire digital data. This step is accomplished by copying data to an image file. Copying digital data accurately is based on established engineering techniques such as error detecting and correcting codes. This is discussed in Sec. 4.2.
3. Ensure integrity of acquired data. Cryptographic hashing is used to ensure that if acquired digital data are changed inadvertently or deliberately the change can be detected. This is discussed in Sec. 4.3.
4. Recover deleted data. In some situations, recovery and reconstruction of deleted data make it possible to bring back deleted files (in whole or in part) or internal records from within an application file. Recovery of deleted data has several risks including missing data and conflating unrelated data. Any recovered item must be evaluated by an examiner for indications of problems. This is discussed in Sec. 4.4.
5. Navigate the acquired digital data. This is accomplished by unraveling, i.e., parsing the layout of the acquired data. This is best performed using a software tool. There is the risk that an incorrect implementation will not correctly interpret the structure of a particular file system, e.g., not show all acquired active files. This is discussed in Sec. 4.5.
6. Identify and extract data artifacts. Items of interest are identified, located, and extracted. This is discussed in Sec. 4.6.
7. Analyze. Examination of extracted artifacts can help develop a narrative or reconstruction of relevant events for inclusion in a final report. This is discussed in Sec. 4.7.

The following 12 key takeaways have been identified in this report. Their number (#x.y) corresponds to which chapter they are located in (x) and their sequence within that chapter (y).

1. **KEY TAKEAWAY #2.1**: In routine operations, computers store much more data than what is typically presented to the user. Examples include storing time and location data on photos, extra copies of data, and data about system activities. Forensic tools and techniques can reveal these data to provide a window into activities that have taken place.
2. **KEY TAKEAWAY #2.2**: Digital forensics is dependent on an understanding of computers and how they work. Any activity that is performed by a computer can potentially be a target for a forensics tool or technique.
3. **KEY TAKEAWAY #2.3**: Computer technology evolves rapidly; however, some attributes of computers last for decades and some only for a few weeks.
4. **KEY TAKEAWAY #2.4**: The forensic examiner needs to be aware of changes in computing technology relevant to the examination being performed. Changes in digital technology introduce the possibility for incomplete analysis or for misunderstanding of the meaning of artifacts.
5. **KEY TAKEAWAY #2.5:** Every digital forensic technique should undergo peer review, formal testing, or error rate analysis. The digital forensics community

performs an informal review by providing timely feedback about the usefulness and validity of techniques through blogs, whitepapers, and videos on the internet. This general acceptance process allows for techniques to be quickly evaluated and revised. While this process is not comprehensive, it does provide significant benefits. Efforts to promote additional rapid peer assessment should be promoted.

6. **KEY TAKEAWAY #4.1:** When using techniques to recover deleted or hidden artifacts the examiner must determine the relevance of the recovered information as it may be incomplete or improperly merged with irrelevant information.
7. **KEY TAKEAWAY #4.2:** Searching tools have limitations based on the multiple ways that computers store information. Limitations include the type of files, types of encoding and many other parameters. In general, digital search tools are very effective at finding information, but there is a possibility that data will be missed because a tool does not have the capability to find it.
8. **KEY TAKEAWAY #4.3:** If someone has taken steps to change information in digital evidence to mislead an examiner, it may be difficult to detect the changes. Identification of deliberate obfuscating changes relies on the skill of the examiner.
9. **KEY TAKEAWAY #4.4:** Digital processes tend to have systematic errors rather than random errors. Therefore, an error mitigation analysis provides more information and is the correct way to manage uncertainty. An error rate is only useful where there are random errors.
10. **KEY TAKEAWAY #4.5:** When error rates are provided, it is important for the user to understand the context of the numbers. For some forensic techniques, the error rates may vary significantly based on attributes of the technology and usage patterns.
11. **KEY TAKEAWAY #4.6:** It is not feasible to test all combinations of tools, run time environments, and digital evidence sources.
12. **KEY TAKEAWAY #4.7:** Extensive testing of over 250 widely used digital forensic tools showed that most tools perform their intended functions with only minor anomalies.

Because of the breadth of digital evidence tools and techniques, it is challenging to properly communicate the results of a digital examination. Some of the basic topics are familiar to most lay people, but the more advanced topics can be difficult to understand. We hope this report will be helpful in communicating the underlying science and its limitations.

1 Chapter 1: Introduction

Digital devices have become ubiquitous in our lives. Many of our everyday tasks are intertwined with the use of mobile digital devices such as cell phones and tablets, personal computers, embedded digital devices and other digital devices. Every interaction with a device has the potential to leave a trail of what we did, who we did it with, where we were and when the event took place. Digital forensics is the application of the scientific method to make sense of the trail left by the interaction with a digital device. All scientific methods have limitations. One must understand those limitations to use a method appropriately. This is especially important in forensic science as critical decisions impacting life and liberty are often based on the results of forensic analysis.

This document is a review of the scientific foundations of digital forensics, and seeks to answer the following:

> *"What empirical data exist to support the methods that digital forensic practitioners use to identify and characterize evidence and associate it with people, places, and things from past events."*

Our approach is to identify and classify the methods and techniques used by the digital examiner and locate relevant literature validating the reliability of the method and to determine whether the scientific approaches and practices used by digital forensics examiners are well-supported and suitable for use. We also discuss knowledge gaps and areas needing further improvement.

1.1 Scope

Due to the breadth of potential topics, the scope of this document is limited to techniques for examining digital data stored in mobile device memory, computer memory, or secondary storage in an active computer. Secondary storage includes devices such as hard drives, flash drives, removable drives, or "external" storage media such as CD, DVD, or memory cards. This document considers the fitness-for-purpose (validity) of the techniques and is not an examination of how well the techniques are used in practice, the best practices for implementing techniques, or limitations placed on usage by the courts. Other digital forensics topics such as network analysis and multimedia (video, audio) forensics are not discussed in detail. This report also does not address other issues such as improved methods for tool validation and verification, privacy, or legal issues, and managing forms of bias within forensic practice. These are outside the scope of this document.

1.2 Who Conducted This Review?

The review team consisted of six individuals from the National Institute of Standards and Technology (NIST) whose diverse expertise permitted examination of the issues from many perspectives, including lessons learned in other fields. Table 1. lists members of the NIST review team, their NIST operating unit, and their area of expertise. Assistance in finalizing this report was provided by several additional NIST employees or contractors as noted in the acknowledgements. Early drafts of this report were sent to several members of the digital investigation community to seek their inputs and reactions.

Table 1. NIST Review Team and their Areas of Expertise.

Name	NIST Operating Unit	Areas of Expertise
James R. Lyle	Software & Systems Division	Computer Scientist
Barbara Guttman	Software & Systems Division	Digital Forensics Research Management
John M. Butler	Special Programs Office	Forensic DNA and Scientific Literature
Kelly Sauerwein	Special Programs Office	Forensic Anthropology
Christina Reed	Special Programs Office	Communication and Science Writing
Corrine E. Lloyd	Special Programs Office	Management Analyst

1.3 Related Work

NIST also performed an interlaboratory study [4] as part of its work on the scientific foundation of digital forensics. The study did not attract enough participants to draw statistically significant conclusions but did demonstrate that digital forensic examiners could answer difficult questions related to the analysis of mobile phones and personal computers. Responses to the study underscored the size, variety, and complexity of the field.

1.4 How is This Report Structured?

This report contains five chapters. Following this introductory chapter, Chapter 2 provides information on the history of digital forensics and background concepts related to computer science. Chapter 3 lists and describes the data sources used and how they were located. Chapter 4 discusses the reliability of specific tasks critical to digital investigations. Chapter 5 provides conclusions and thoughts on the future directions for the field.

The initial release of this report is a draft document, and we welcome comments and feedback from readers. All relevant submitted comments will be made publicly available and will be considered when finalizing this report. Do not include personal information, such as account numbers, Social Security numbers, or names of other individuals. Do not submit confidential business information, or otherwise proprietary, sensitive, or protected information. We will not post or consider comments that contain profanity, vulgarity, threats, or other inappropriate language or similar content. During the 60-day comment period, comments may be sent to scientificfoundationreviews@nist.gov.

1.5 Comparison of Non-Digital to Digital Investigation

Digital investigation techniques are based in practices and knowledge from the field of computer science. This field can often be daunting to the uninitiated, as a significant time investment is required to learn obscure technical concepts and terminology. However, understanding the process of a forensic examination of digital data is not as difficult as one might first suspect and is analogous to many elements of a non-digital investigation. This section compares tasks in a digital investigation to a non-digital investigation to illustrate their similarities.

Consider a search of an office or residence to find something relevant to an event of interest, possibly a crime, an accident, or other event that needs to be better understood. After obtaining proper authorization and a warrant for a search, a search can proceed. Digital evidence only differs in minor ways from physical evidence in the concept of search and seizure. For a physical search, the authorization covers a search of the location and the seizure of objects of possible evidentiary value. For a digital search, all the contents of digital storage devices are preserved (copied) so that only authorized portions can be examined later.

Just as in a non-digital investigation, the digital investigation seeks to create a timeline of events (to identify what actions occurred), reconstruct fragmented artifacts, identify people involved in an incident., determine means (how the incident occurred), establish opportunity, and find other relevant evidence. The object of the search could be records of nefarious economic activity, possession of contraband, weapons or tools used in a crime or indications of movement of a suspect. The location searched could be anything from a small apartment (a small computer) to a large farm (a server farm with many computers and removable devices) with barns and outbuildings (offline storage and archives), vehicles (mobile devices) and out of the way hiding places (box of CDs/DVDs in a closet). A search of a large property may uncover a skeleton in an unmarked grave, and an examination of the bones[1] may reveal relevant details about the person (just as the search of a server farm may turn up storage devices that can be examined for deleted files using clues from metadata or by file carving[2]).

In both digital and non-digital circumstances, the examiner is interested in learning more details about some event of interest, and a search of the property is expected to uncover evidence that can be used to inform decision makers such as a judge. Likewise, search of a digital device (computer, mobile phone, removable storage, cloud, or other digital device) seeks to find relevant evidence related to an event. Non-digital investigations are guided by the principle that "Forensic science seeks to establish connections (or lack thereof) between evidence and its source . . . we consider the probability of the evidence in light of competing hypotheses" [5]. In like manner, a digital investigation generates hypotheses, and the investigator searches for data artifacts, e.g., files, logged events with a time stamp, emails, that can be used in evaluating observed evidence considering alternative (opposing) hypotheses.

Examples of items relevant to a non-digital investigation and possible corresponding items relevant to a digital investigation are presented in **Table 2**. This is intended to help the reader who is not familiar with digital devices see the digital world in more familiar terms. Keep in mind that the correspondences are only approximate and should not be carried too far.

It is important to recognize that the goals of both a digital and a non-digital investigation are the same. Both types of investigations revolve around questions critical to identifying the actors and their actions involved in the events under consideration.

[1] The examination may require a specialist to do the examination.

[2] The deleted data recovery may require use of an additional tool for the data recovery.

Table 2. Physical Investigative Items and Corresponding Digital Items

Physical-World	Digital-World
Crime scene or a place to search for evidence: could be a small site like an apartment or a large site like a farm or business.	Computer, mobile device, storage device: a device to be examined; a server farm with many computers.
An item of evidence that is fragmented: shredded document, buried body.	Deleted data: evidence that isn't apparent with the usual computer user tools and can't be examined without some reassembly.
On-site records such as a filing cabinet or desk. There may or may not be a log kept of access to files.	Files stored on the computer hard drive, or removable media. An access log may be kept automatically.
Offsite records such as at a business branch office, a summer home, or a storage locker.	Files stored on a cloud server or off-line on removable media.
Burglar's tools or weapons.	Hacking or obfuscating tools.
Names, phone numbers and addresses from a list of contacts, e.g., address book on paper.	Contact list from a mobile device.

It is important to recognize that the goals of both a digital and a non-digital investigation are the same. Both types of investigations revolve around questions critical to identifying the actors and their actions involved in the events under consideration.

There are general principles of forensics [6] that guide the examination of evidence, building on principles developed earlier [5]:

- Authentication – Is there sufficient confidence that a claim is true?
- Identification – Is there sufficient confidence that something is what it is claimed to be?
- Classification – Is there sufficient confidence that something has been assigned to the appropriate category?
- Reconstruction – Have the elements of the case been organized in the most likely grouping of capabilities, patterns in time and linkages among entities?
- Evaluation – Is there enough information to provide input into a decision process?

Note: in digital forensics "authentication" is defined by the SWGDE Digital & Multimedia Evidence Glossary as "the process of substantiating that the data is an accurate representation of what it purports to be" [7].

In applying these principles, a non-digital investigation may require a variety of forensic tasks such as:

- Surveying the crime's location.
- Identifying items found at the crime scene, e.g., blood, a bullet, or something dropped by someone present.
- Attempting to identify the source of a particular item.
- Extracting useful DNA from biological samples that might be a single source or a mixture.
- Identifying the owner of an item or determining who used the item last.
- Determining what discrete events occurred and their order.

Other more detailed examples of investigative tasks, both digital and non-digital, are available [6]. A digital investigation usually involves a slightly different, but similar, set of tasks. Some example tasks are:

- Acquiring (or gaining access to) the digital data.
- Ensuring the integrity of the data.
- Reconstructing and recovering deleted artifacts.
- Identifying relevant artifacts.
- Extracting relevant artifacts.
- Classifying relevant artifacts.
- Assembling a narrative of what happened.

In a digital investigation there can be a long list of tasks associated with the analysis, each with a different technique required to obtain a resolution. The tasks considered are context sensitive to the type of crime, type of information needed from digital evidence, and types of digital evidence that are available. A digital investigation can encompass many seemingly unrelated artifacts that need to be assembled to make a more complete narrative of events.

It is important to recognize that digital evidence is generally a part of a larger investigation. The following example shows how digital evidence can be used as part of an investigation using the hierarchy of propositions (source, activity, offense) from the hypothetico-deductive method [8, 9].

Rancher Alejandro reports that his favorite horse, an Appaloosa named Spunky appears to have been stolen last Saturday. Alejandro notes that there is a boot print in the ground by the door next to Spunky's stall. A ranch hand, Big Jake, has been identified as a suspect. There are three levels to consider:

- Level I: Source: The boot print was made by Big Jake's boot versus alternatives such as some other boot left the impression.
- Level II: Activities: Big Jake took Spunky from his stall versus alternatives such as someone else took Spunky.
- Level III: Offense (to be considered by the trier of fact): Big Jake stole Spunky from his stall.

Often both evidence from the physical world and the digital world are combined to create a complete picture of events. For example, an examination (after proper authorization is obtained) of Big Jake's mobile device yields the following items:

- A picture of an Appaloosa horse with a pattern of markings consistent with Spunky's was found on Big Jake's mobile device with a time/date stamp of Sunday at 10:37 p.m., after Spunky had been reported stolen, and geolocation data matching the location of Big Jake's brother's farm.
- Text messages to a livestock market asking about selling an Appaloosa.

Together the physical and digital evidence paint a basic picture of the events, but additional case work must be done, of course. This example illustrates how elements of both the physical world and the digital world fit together to build a case and how statements about sources, activities, and offences form a hierarchy for consideration.

Much of the burden of completing these digital tasks is carried out by software tools that interpret the bit patterns of digital objects and implement the underlying algorithms designed to accomplish each task. In the end, the job of the digital examiner is to use tools to find relevant information from digital evidence. The questions that the tools can answer range from very general (e.g., show the actual bits stored in a specific location) to very specific (e.g., display the email sent to John Smith on January 1, 2020).

2 Chapter 2: Computer Science Background and History of Digital Forensics

2.1 Computer Background

Before discussing specific tasks, it is helpful to review some background about how computers work, encode and organize digital data.

Computer software is what makes a digital device useful. A user interacts with a digital device through an operating system and application software. The application software uses the environment provided by the operating system to complete tasks requested by the user. Any interaction (storing, changing, or deleting data) with a digital storage device happens through the file system. As illustrated in Fig. 1, the user views data stored on data storage hardware through these layers of software.

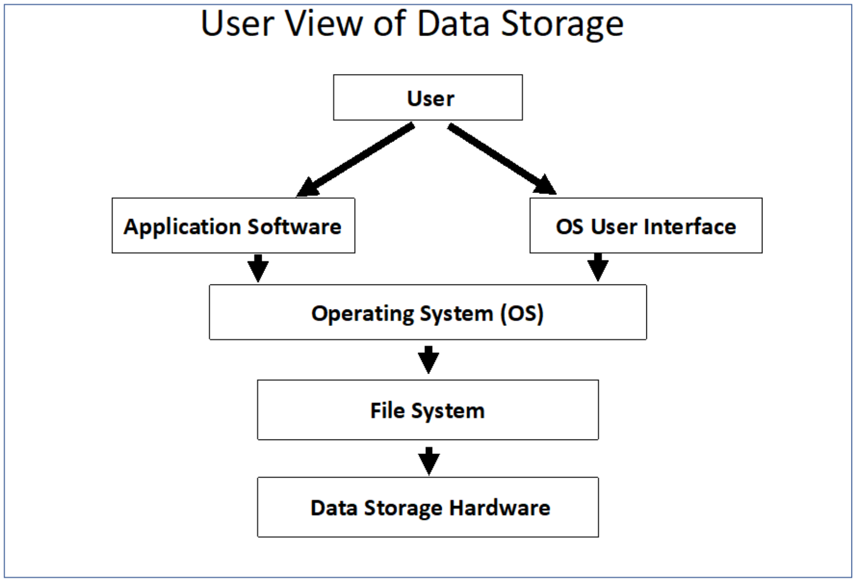


Fig. 1. User View of Data Storage.

Computers automate many kinds of tasks, such as mathematical calculations, record keeping, machine tool control, etc. The computer accomplishes an assigned task by following a list of instructions called a program, also known as computer software or computer code. The instructions describe the task in fine detail with steps such as “move this data item over there” and “add those two data items” or “if these two data items are the same, skip the next instructions and continue running from another part of the program.” Most program instructions are a variation of these three types of instructions (move data, do math with data, and if-condition-is-true-go-to alternative set of instructions). These instructions are too detailed for a person to write a program quickly, so usually an easier-to-understand programming language is used that is then translated into the machine language of the computer.

Early in the development of digital computers the need for reliability was recognized and means to ensure reliable data transfer was developed. Transferring data within a computer has to be extremely reliable because “in a digital computer … a single failure usually means the complete failure [that] . . . if it escapes detection then it invalidates all subsequent operations of the machine." [10] The reliability of copying data within a computer system is ensured by error correcting codes (ECC) incorporated in the actual representation of data. These codes are used to protect a block of data from changes introduced by random noise as the data are moved from one location to another (possibly from one memory location to another within the same device, or a transmission from one physical location, e.g., satellite in orbit to a receiving station on Earth). These codes are implemented by computing a signature for a block of data to be protected and then transmitting the code (the ECC) with the protected block of data. A function is applied to the received data and compared to the transmitted ECC, and the result indicates if the received data are error free or if an error occurred in transmission. The ECC may be designed to indicate which bits of the transmission have been modified and can, therefore be corrected.

Data storage has evolved with frequent changes to details of how things are done, but the basic organization has stayed the same, as illustrated in **Fig. 2**: a raw unformatted storage device contains a sequence of bits (or bytes) with no meaning. After formatting, the device contains a map of the layout of partitions on the device (the device metadata). Each partition is formatted with a selected file system including the layout of files on the device in partition metadata and any stored data. At each level, metadata keeps descriptive data about what objects are being stored on the device. Digital data created by the actions of computer users are collected into files that are organized on a digital storage device in a file system by the operating system. The next 3 figures (**Fig. 3**, **Fig. 4** and **Fig. 5**) provide additional details on topics discussed in this section.

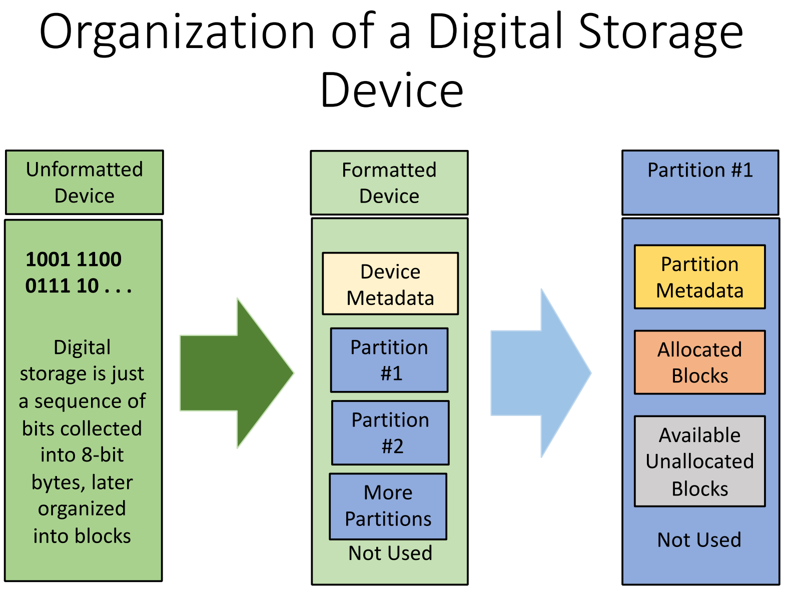


Fig. 2. Storage Organization for a Single Device.

A method for representing numbers using a string of symbols selected from a fixed set of symbols that are assigned value based on the relative position within the string[1]. The usual method is called base 10 (there are ten symbols in the set: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9). Other bases that are encountered in everyday life include: base 12 and base 24 for telling time, base 60 for minutes and seconds. Base 60 is also encountered in measuring angles with degrees, minutes, and seconds.

Most computers use base 2, known as binary, because of the ease of representation as just *on* or *off*. Binary numbers are usually represented with strings of 1s and 0s. The problem for a human is that binary numbers require rather long strings of 1's and 0's to represent a number. At least 16 binary digits are required to represent numbers greater than 32,768. Because strings of binary digits are rather long, a more compact form is often used. Binary numbers are easily grouped into sequences of three or four binary digits. Sequences of three binary digits can be represented in base 8 by octal digits (0, 1, 2, 3, 4, 5, 6 and 7). Sequences of four binary digits can be represented in base 16, also called hexadecimal. Base 16, is used to present binary data by grouping 4 binary digits together as a single hexadecimal digit. The 16 symbols usually used are the digits 0-9 for the first ten symbols and the letters A-F for the remaining 6.

Fig. 3. Place Value Notation.

An isolated bit string (sequence of binary digits) can represent a variety of digital objects. If the meaning of a bit string is to be understood correctly, the appropriate interpretation must be applied. For example, the single byte 0x61 (0110 0001 in binary) represents the eight-bit (one byte) integer value 97 in decimal. However, if this is part of a block of text, it represents the letter 'a' if the text is encoded as ASCII, but if the text is from an IBM mainframe using Extended Binary Coded Decimal Interchange Code (EBCDIC), then 0x61 represents a slash ('/') (the letter 'a' would be encoded as 0x81 in EBCDIC).

Fig. 4. Assigning Meaning to a Bit String.

It is important to note that a great variety of information can be stored in digital form, e.g., numbers, text, pictures, videos, or time of an event. Modern computers usually represent data in base 2, also known as binary. Instead of using 10 symbols to represent a number, binary uses only two symbols, 0 and 1. The encoding of objects such as a picture, a music recording or a document is accomplished by representing each item of information as a sequence of numbers, i.e., binary digits, because computers only operate directly on numbers; everything is represented as a sequence of numbers. It is critical to understand the encoding of each digital object so that it can be correctly interpreted.

2.2 Encoding Data

The fundamental unit of digital data in contemporary computers is the eight-bit byte taking on values from 0 to 255 (this can also be considered as values from -128 to +127). A byte is made up of 8 binary digits or 2 hexadecimal digits (base 16).

The oldest encoding schemes, EBCDIC (Extended Binary Coded Decimal Interchange Code) and ASCII (American Standard Code for Information Interchange), used one byte per character, but this limits the number of languages that can be represented. The ISO/IEC 8859 encoding exploited that ASCII only used 7 bits of each byte and used the extra bit to encode other character sets and languages. The weakness of the ASCII and ISO/IEC 8859 encodings is that it covers only 16-character sets and is only suitable for a few languages, mostly European, southeast Asian, and Middle Eastern, due to the limited number of symbols that can be represented and is entirely unsuited to representing the thousands of symbols needed for most Asian languages. Before Unicode there were independent character encodings in China, Taiwan, Japan, and Korea. Vietnamese uses Latin characters with diacritics.

Unicode replaced all this. Unicode was developed to address these problems and can represent millions of symbols allowing for text not only in Asian languages, but in most languages of the world, and other representations such as Egyptian hieroglyphs and emojis, i.e., pictograms.

Fig. 5. Encoding Text.

From this simple foundation a vast array of digital objects can be represented, for example:

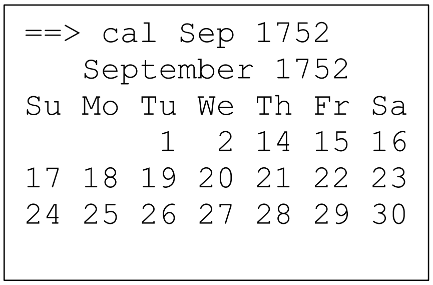
- **Integers of arbitrary size.** There are often capabilities built into the hardware for integers of varying size. Such integers are built by putting together a sequence of bytes, doubling the size at each level. Binary numbers of 8, 16, 32 and 64 bits are typically supported by the hardware. Larger size integers must be manipulated by software.
- **Fractional numbers of arbitrary scale.** Numbers that range from the atomic scale to the cosmic scale would be tedious to write and difficult to understand and manipulate without a compact notation such as scientific notation, e.g., 6.02 x 10^{23}. Computers represent scientific notation by a pair of integers, a fractional part, and an exponent, e.g., the pair of integers (602, 21) is used to represent the number 6.02 x 10^{23} in scientific notation.
- **Text.** Text is just a string of symbols. An encoding scheme (see **Fig. 4** and **Fig. 5**.) assigns each unique symbol a unique number. However, there is more than one encoding scheme available to use, e.g., ASCII, ISO/IEC 8859, Unicode (7-bit, 8-bit, 16-bit and 32-bit), EBCDIC (old IBM), and various non-Unicode Asian character sets.
- **Images.** The basic abstract representation of an image (a picture) is an array of pixels. Each pixel represents the color and brightness of a point in the array. There are several standardized formats for storing the pixels of an image, e.g., JPG, GIF, and PNG. These different formats offer various tradeoffs in space and capabilities, e.g., exactness of representation of the original.
- **Audio.** Digital audio recordings can be autonomous or part of a video file stream. Typical formats include .wav, .mp3, .wma, and .m4a.
- **Video.** There are several standard formats to store a video represented as a sequence of frames (individual images in some format), e.g., mp4 or mov.

- **Encryption.** Any digital object can easily be encrypted so that a decryption key is required to examine an encrypted object. Recovering the unencrypted digital object is essentially impossible unless enormous computational resources are employed or the implementation of the encryption is faulty, e.g., the decryption key is exposed somewhere.
- **Compression.** Files, folders, and file systems are sometimes stored in a compressed format to save space. Compressed data must be properly expanded to be examined.
- **Time Stamp.** Used to record when an event occurred. There are many options for representing time and dates to choose from.

2.3 Time

Times and dates can often exhibit subtle nuances that are prone to misunderstanding. For example, George Washington has two birth dates. He grew up celebrating his birth date as February 11 under the Julian Calendar, which the Colony of Virginia was using when he was born. But beginning with his 21st birthday, he began celebrating it (as well) on February 22. The previous September, Great Britain, and by law her colonies, switched to the Gregorian Calendar. The day changed due to the removal of 10 days (3-13) from September 1752; and his year of birth changed from 1731 to 1732 because the new year began on March 25th under the Julian Calendar and the change to the Gregorian Calendar shifted the beginning of the year from March to January.

This calendar anomaly illustrates the complexity of evaluating software correctness. Computers have software to display a calendar for a given date. In the results produced by the UNIX **cal** command (**Fig. 6**) for September 1752, note that September 2 is followed by September 14.



```
==> cal Sep 1752
   September 1752
Su Mo Tu We Th Fr Sa
       1  2 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
```

Fig. 6. UNIX cal Command Output.

The calendar produced by the UNIX **cal** command is correct for Virginia, but not for before September 14, 1752, in San Antonio, Texas because in 1752 Texas was part of Mexico and was already using the Gregorian Calendar. The **cal** command output is correct or not depending on the context. This is occasionally the case for forensic software, too. For a simple sounding task as "identify files that contain one or more social security numbers" different forensic tools might use different definitions of "what does a social security number look like" and then produce different results.

The Julian/Gregorian issues are not likely to impact dates and times in digital forensics, but it does illustrate that there can be unexpected issues with understanding time. For digital forensics there are other issues with representing time and date that need to be considered. For example:

- What data format is being used, e.g., a character string, or an integer offset from some epoch. A time might be recorded as the string "01:35," but this is ambiguous. It could be either AM or PM (and the time zone is unknown) without additional context. If the time/date is stored as an offset, the value is stored as an integer and needs to be converted to an understandable form.
- A date stored as a character string may be ambiguous. For example, 1/2/03 could be interpreted to mean:
    - January 2, 2003 (month-day-year, in the US).
    - February 1, 2003 (day-month-year, not the US).
    - The year might be 1903 or 1803 or something else.

There is an international standard for dates: YYYY-MM-DD, i.e., 2003-01-02.

- If the time is stored as an offset from the beginning of some epoch, the date/time of the epoch and the granularity of the values must be known. For example, the Unix epoch began on 1 January 1970 00:00:00 UTC[3], with a granularity of 1 second. The MS Windows epoch began on 1 January 1601 and has a granularity of 100 nanoseconds. Other epochs and granularities are also in use. If a forensic tool reports a date of January 1, 1601, for data obtained from an NTFS file system, the most likely interpretation is that the time field contained a zero instead of a valid date and time.
- The configured time zone must be known. The time zone setting might be incorrect, e.g., the default time for some systems is Pacific Time and a user might misconfigure the time zone. Not all jurisdictions conform to the recommended time zone standard, for example, Utah in the summer follows Mountain Daylight Time (MDT), but Arizona follows Mountain Standard Time (MST), an hour later.
- A computer system may store time in either local time or in GMT (Greenwich, London).
- The clock of a computer can be managed in several ways. Time is usually managed by reference to an internet time server (and times are usually within a second of being correct), but a user may not have internet access or may choose to set the system time manually. Time may be wrong due to the system clock drifting off the correct time or because a user intentionally set the clock to the incorrect time to mislead an investigator.

2.4 Types of Digital Data

There are several types of digital data that may be useful in an investigation. These include:

- Documents, emails, spreadsheets, pictures, videos, programs, applications, and so forth.
- Objects that are directly created by the computer user.
- Objects that are downloaded from a remote source by the computer user.

[3] Greenwich Mean Time (GMT) is the mean solar time at the Royal Observatory in Greenwich, London, reckoned from midnight. UTC is the time, based on a time standard, in the GMT time zone.

- Metadata associated with an object, such as, file MAC (Modify/Access/Create) times, file ownership, file permissions, picture EXIF data, and document metadata.
- System log files. An operating system records a multitude of events as they occur. For example, if a user ever connected a removable storage device to a computer that action might be recorded in a logfile somewhere on that computer. The location and format of the log varies with the operating system.
- System configuration files and various types of associated metadata. These files describe options chosen by the computer user that affect system behavior. For example, a partition table is metadata that describes the layout of a storage device.
- Other system files, for example a volume shadow copy or page files.
- System memory. This is volatile, i.e., changes very quickly over time, and needs to be acquired in a timely manner by someone trained in memory acquisition.
- Remote access network traffic.

2.5 Operating Systems

Computers usually run what is called an operating system, a program that manages the computer operation and does routine tasks such as logging on users, switching between running programs, and interacting with a file system for updating secondary storage as directed by the running programs. The running operating system of a computer establishes a collection of artifacts in various configuration and log files. These files are a rich source of artifacts that track user activity and can be extracted for forensic analysis.

There are several families of operating systems likely to be encountered in an investigation. The main operating system branches are Microsoft Windows-based and Unix-based. Operating systems are continually evolving. Each new version has differences, some major and many minor, from the previous version. Sometimes an operating system branch splits into two or more development lines. For example, the Unix family split early into the Berkeley (BSD) version and the AT&T version. The result is more than 10 individual Unix variants today, each with a separate history. Also, two independent clone branches were developed, Minix and Linux, that look like Unix but are independently developed without sharing any source code with any Unix variant. In the Unix world, Apple Mac OSX and iOS for mobile devices are based on a BSD variant; the Solaris OS, often used for network servers, is an AT&T-based system; and the independently developed Linux-based operating systems are used in a variety of applications such as file servers and Android mobile devices [11].

Each version of an operating system has a similar set of extractable artifacts. With a new software version, an artifact could be found in a different location, or the exact interpretation of the information in the artifact may change. For examples of changes introduced by new versions of software, see Fig. 7.

On an iPhone running operating system version iOS 4.3.2 or earlier, location services information (a record of where the phone has been) are saved in files along a path that includes the subdirectory "mobile," for iPhones running operating system version iOS 4.3.3 or later, the information is moved to a different location in the file system along a path that replaces the subdirectory "mobile" with "root."

An example of changing interpretation would be the first 512 bytes of a storage device, called the "boot sector." Starting in the 1980s, the boot sector contained a "master boot record" (MBR) which included a map describing the layout of partitions on the storage device, known as the "partition table." As storage devices evolved to larger sizes, the interpretation of the information in the partition table changed, e.g., one change was a switch of disk addresses from one format to another. Another change that occurred later (late 1990s early 2000s) was the introduction of an alternative partition table format using "globally unique identifiers" (GUIDs). Starting with OS versions introduced since 2000 either scheme can be used when a storage device is set-up [2].

Fig. 7. Examples of Introduced Changes

2.6 File Systems

Storage devices need an organization scheme so that desired data can be managed, found, and retrieved. Such an organization scheme is called a file system. File systems specify how files are organized on secondary storage as directed by operating systems and application software. The file system manages the details of placing, creating, updating, and deleting files as directed by the user.

There are several approaches to placing file systems on storage hardware that might be followed:

- The file system takes up the entire device. This is often true for flash drives.
- The device is partitioned into several areas such that each partition contains its own file system. The first few sectors of the device contain a partition table that describes the layout of the partitions on the storage device. There are several partition-table schemes. The most often encountered are Master Boot Record and Globally Unique Identifier (GUID) Partition Tables [12].
- To improve the reliability and performance of a file system, it can be duplicated across multiple independent devices as a Redundant Array of Independent Disks (RAID) in one of several ways, referred to as RAID levels. Each level gives different tradeoffs between cost, reliability and performance.

Common file systems on devices used with Microsoft Windows systems are NTFS, ExFAT and FAT. LINUX systems use ext2, ext3, ext4, and FAT. Apple Macs use HFS+, APFS, FAT and ExFAT. Except for FAT file systems, most file systems are specific to a particular operating system with capabilities, limitations, supporting operating systems and available artifacts varying by system. Sometimes limited or third-party support is available for file systems not native to a particular operating system.

The implementation of a file system tries to minimize both access time to the stored data and time required to keep file metadata up to date. When the computer reads or writes a file, it might trigger an update to file metadata, e.g., writing data to a file might also cause an update to the modification time of the file. The underlying physical design of the storage hardware and the storage capacity have a major impact on achieving the goal of minimizing time to interact with a storage device. The storage technology has evolved from spinning magnetic media to solid-state devices. With spinning media, access time depends on the time required to move physical storage device components into position to interact with the magnetic media. Placement of data has significant impact on the time required to read or write data or metadata. With a solid-state device considerations of data placement no longer apply; access time is constant and not affected by placement of data.

The capacity of storage devices has also evolved over the past 25 years from less than 2GB to several terabytes. As storage capacity increased, the protocol for specifying a data location evolved from a three-part address of cylinder/head (or track)/sector reflecting the design geometry of the spinning magnetic media to a three-part address created by the BIOS to allow for specification of a larger address space, to a 24-bit logical block address (LBA) and then to a 32-bit LBA. The unit of addressable data, the sector, had been fixed at 512 bytes. To accommodate expanded storage capacity, the sector size of the latest drives has been increased by multiples of 512 bytes usually to 4096 bytes [13]. This is just one more consideration for forensic tool design that could be overlooked.

2.6.1 Creating Files

When a file is created, several metadata artifacts are also created. Most of these artifacts have a unique interpretation for each file system type. Understanding the differences in interpretation is required for correct reporting of results in a forensic examination. For example, file access time for FAT file system has a resolution of 1 day, i.e., the date when the file was accessed is recorded but not the time of day. However, on NTFS file systems, file access time has a resolution of 100 nanoseconds, i.e., the time of day (according to the system clock) down to within 100 nanoseconds is recorded, not just the date. Of course, access time tracking might be disabled (with no record of file access recorded) or reenabled.

Windows-based file systems may have a short file name abbreviation for each file in addition to a longer file name. Metadata specifying ownership by an account on the computer and access permissions is created at the same time as a file, but what is recorded varies across file system types. Examples of file metadata include full file name, a short file name, owner account, access permissions, or modify-access-create (MAC) times. As a simplified example of the variation of some recorded details across file systems, note that FAT file systems do not keep permissions or file ownership but, NTFS file system keeps a list of file access permissions by specific users, while Unix specifies permissions on a file by groups of users.

2.6.2 Updating Files

Updating a file creates traces and artifacts that provide the forensic examiner opportunities for tracking a suspect’s behavior over time. An application or text editor might copy a portion or even all of a file to a temporary location and the copy may persist for some time before being overwritten. Some update procedures create a new copy of the updated file with the original file left intact but marked deleted. The deleted original file may be recoverable, in part or entirely. The file system may update file times to indicate that the file has been changed.

When a file is copied to another location the metadata for the copy might differ or be preserved, depending on the options given to the copy operation. For example, depending on the options given to the copy command, MAC times, file permissions or ownership might change or stay the same when a file is copied.

2.6.3 Deleting Files

When a file is deleted, a file system typically does not erase the file content from a spinning magnetic storage device but leave the content in place and just mark the file as deleted with the allocated sectors added to a list of sectors available for reuse. This reduces activity on the spinning media while ensuring that the deleted file name is no longer visible to the user. This was often done on the earliest file systems such as FAT to improve storage device performance times. Forensic tools can exploit such behavior to recover files that have been deleted. However, some operating systems and file systems, such as Mac OS with HFS+, may offer the user an option to overwrite any content associated with a deleted file.

With the introduction of solid-state drives, new strategies have emerged. New device commands were introduced (TRIM usually for SATA type devices and UNMAP for SCSI devices) that mark a block of storage as unused and a candidate for trimming (erasure of block content). The storage device removes content at a convenient later time. This can lead to surprising results such as if a device is imaged and hashed just after arrival in a forensic lab and then later the device is imaged and hashed again. The two hashes might not agree if there are "trimmed" data not yet removed at the time of the first image that are then removed by the solid-state firmware in before making the second image.

> **KEY TAKEAWAY #2.1**: In routine operations, computers store much more data than what is typically presented to the user. Examples include storing time and location data on photos, extra copies of data, and data about system activities. Forensic tools and techniques can reveal these data to provide a window into activities that have taken place.
>
> **KEY TAKEAWAY #2.2**: Digital forensics is dependent on an understanding of computers and how they work. Any activity that is performed by a computer can potentially be a target for a forensics tool or technique.
>
> **KEY TAKEAWAY #2.3**: Computer technology evolves rapidly; however, some attributes of computers last for decades and some only for a few weeks.
>
> **KEY TAKEAWAY #2.4**: The forensic examiner needs to be aware of changes in computing technology relevant to the examination being performed. Changes in digital technology introduce the possibility for incomplete analysis or for misunderstanding of the meaning of artifacts.

2.7 Digital Forensics Overview

Digital forensics is not a single technique but many independent techniques, which operate on digital data that must be used within the limits imposed by the court. However, these limits do not influence the technical capabilities of digital forensic techniques.

The techniques applied to a specific case depend on the type of information likely to be useful for understanding what happened. For example, browsing the contents of a digital device can identify records of financial misconduct, communication with others indicating collusion in illegal activities, or possession of contraband material.

Techniques for digital forensic analysis have been developed by digital forensic examiners and tool vendors (just as in other fields) trying to answer the classic questions required to resolve a case. Because computing technology is changing rapidly, there is a possibility that no tool will be able to find or correctly parse all the information in each piece of digital evidence, especially for more recently introduced or upgraded technology.

A digital investigation begins with an evaluation of the case context and the digital devices being examined. An examination of a seized mobile phone from a suspected drug dealer might begin by reviewing contacts (possible customers and collaborators) and messages (setting up illegal transactions). A suspected espionage case might require the examiner to look for contraband (classified documents), removable device history (moving the contraband around), geolocator information (places the suspect has visited), contacts (identify collaborators), messages (extraction of planned actions) and deleted documents (hiding activity).

2.7.1 Digital Case Example

An example of deleted data being critical to an investigation is the Bind-Torture-Kill **(**BTK) case [14]. A serial killer operating in Kansas off and on for over ten years sent messages to the police to taunt them. The subject of interest sent a message on removable storage media that was examined for anything present in unallocated space. A deleted document was recovered from the unallocated space of the device and the document metadata yielded a name, Dennis, and an organization. An examination of the membership of the organization revealed that Dennis Rader was president. While this did not establish anything definitive, it did create potential investigative leads for the examiner, such as:

- Dennis could have been a previous user of the device that BTK later obtained.
- BTK could have obtained a copy of a file with the name “Dennis” from somewhere and placed the copy on the device before deleting the file.
- Dennis could be involved (and after additional investigation Dennis was determined to be BTK).

Until the recovered document was examined, the investigation was not progressing. With the information from the storage device additional investigation including DNA results obtained from a hospital biological specimen of Dennis Rader’s daughter solved the case.

2.7.2 Characterizing the Digital Forensics Community

There is a lack of consensus about what constitutes a digital forensics lab or how many of them there are in the USA. There are 400 US crime labs, but clearly there are many more digital forensics labs. While most crime labs have a digital forensics section, there are many other organizations that also perform digital forensics examinations. For example, digital forensics labs are also found within specialized labs that only process digital evidence, such as Internet Crimes Against Children (ICAC) labs or Regional Computer Forensics Labs

(RCFLs managed by the Federal Bureau of Investigation (FBI)), in law enforcement agencies based at the federal, state, and local levels, inspector general offices, and prosecutors' offices. They are also found in corporate offices that work closely with law enforcement, and there is substantial overlap with incident response and other cyber security operations. Digital forensics also has a significant presence within the intelligence community and is widely used in civil cases, where it is often referred to as "eDiscovery". Because of this breadth, it is difficult to estimate the size of the digital forensics community.

One method that has been proven to accurately estimate population sizes is called capture-recapture and has been widely used in biology and ecology to investigate the dynamics of biological populations. The method has also been utilized in epidemiological studies of human samples and is applicable for estimating the size of a population from multiple lists of individuals as is the case here [15, 16]. An initial sampling event attempts to 'capture' a significant sample of the population. Then, the population is resampled (i.e., recaptured), and the number of individuals in each sample and the number common to both samples are used to estimate the total population. The number of individuals missed, or not captured in either event, can then be estimated [17].

To use the capture-recapture methodology, we first identified lists that contained information about digital forensics labs. We define "digital forensics laboratories" as any entity that processes or uses digital evidence. We obtained lists from the following organizations, selecting only US labs:

- The International Association of Chiefs of Police (IACP) maintains a directory of cybercrime labs through the Law Enforcement Cyber Center.4 As of August 11, 2021, there were 354 unique groups on the list.
- The International Association of Computer Investigative Specialists (IACIS) includes 2,214 groups in their training list. IACIS provides training and certification to the worldwide digital forensics community and counts federal, state, and municipal law enforcement agencies as well as other professional digital forensic practitioners amongst its members.
- The ANSI National Accreditation Board (ANAB) is the largest accreditation body in North America and provides training and accreditation to both public and private organizations, including digital forensic labs. Ninety-one of the active ANAB accredited organizations in the United States process digital evidence.
- The Scientific Working Group on Digital Evidence (SWGDE) seeks to foster communication, cooperation, and to ensure quality and consistency within the digital and multimedia forensic communities through the development of guidance documents. The 56 members of SWGDE come primarily from federal, state, and local law enforcement agencies as well as academic, corporate, and civil forensics groups.
- The National White Collar Crime Center (NW3C) offers training and professional development courses in the prevention and investigation of high-tech crimes for federal, state, local, and tribal law enforcement, prosecutorial, and regulatory agencies. They provide training and analytical technical support in computer forensics, financial and cybercrime, and intelligence analysis. Their list of training participants from 2020 contains 4,008 unique groups.

The total number of unique US digital evidence groups represented by these lists is 5,457. While these lists are current as of August 2021, it is then assumed that they accurately capture a stable population at a single point in time. However, there are other organizations representing digital evidence processors that might not be included in this assessment, therefore an estimation method is needed to get a better idea of the true number of processors in the digital forensics field.

The capture-recapture method yielded a lower bound estimated population size of 11,000 with a 95 % confidence interval of (9,900, 12,600). Due to the overlap between the lists and the fact that some of the total population has a zero probability of being selected in any list, the final value is interpreted as a lower bound estimate, rather than an absolute population size. This value of 11,000 US digital forensics organizations contrasts with the 409 publicly funded crime labs reported by the Bureau of Justice Statistics [18]. The decentralization of the digital forensics community in the United States is apparent in digital forensics labs: they are not only in federal, state, and local crime labs, but also in prosecutor's offices, private consulting firms, and corporate cybersecurity operations.

2.8 Information Sources used by Examiners

There are many sources of information available to an examiner. Information falls in several broad categories: general background information of computer science and digital examination, information about how to use various tools and techniques, and specialized information about operating systems, applications, and the artifacts created by them. This last category can also include specialized techniques that are not used on a routine basis. The primary sources of knowledge about digital forensic techniques include the following:

- Training organization classes (independent of specific tool vendors)
- Tool vendor offered classes
- Forensic tool vendor white papers and other support documents
- Forensic professional organizations
- Standards organizations
- Online training videos
- Blog posts
- Academic peer reviewed papers in conferences and journals
- Academic course work
- Reference books
- Operating system and computer hardware vendors support documents
- Reverse engineering of software: operating system, file system or application

2.8.1 Training organization classes (independent of specific tool vendors)

There are several independent training organizations that provide a range of classes from introductory to advanced topics in digital forensics. This is one of the main pathways for the practitioner to learn the methods and techniques of digital forensics.

NCFI – National Computer Forensics Institute

The NCFI is a federally funded training center dedicated to instructing state and local officials in digital forensics and cybercrime investigations and is operated by the United States Secret Service's Criminal Investigative Division and the Alabama Office of

Prosecution Services. The center offers classes for first responders, individuals who are newly assigned to conduct digital forensic exams, advanced classes for individuals currently conducting forensic examinations and classes to prepare judges and prosecutors to effectively preside over and prosecute cases involving digital evidence.

FLETC – Federal Law Enforcement Training Center
FLETC offers basic and advanced classes for law enforcement personnel in areas such as digital evidence acquisition, digital evidence analysis and evidence recovery.

NW3C – National White-Collar Crime Center
The NW3C delivers training in computer forensics, cyber and financial crime investigations, and intelligence analysis. They also offer analytical technical support to agencies investigating and prosecuting white collar and related crimes. They conduct original research on all facets of white-collar crime.

DC3 Cyber Training Academy
The DC3 Cyber Training Academy provides Training for DOD service members, active duty, civilian, Reserve or National Guard personnel involved in investigating cybercrime. The academy's training prepares DOD personnel to do computer forensics as part of their assigned tasks.

SysAdmin, Audit, Network, and Security (SANS Institute)
SANS is a for-profit training company that offers many computer security focused classes; in addition, SANS offers several classes in many aspects of digital forensics.

Professional Organizations
Professional organizations, such as the High Technology Crime Investigation Association (HTCIA), and the International Association of Computer Investigative Specialists (IACIS), are two examples of professional organizations that provide training on a wide selection of digital forensics topics.

2.8.2 Digital Forensic Tool Vendor Offered Training

Digital forensic tool vendors often offer classes in basic principles of digital forensics, but usually with an emphasis on using products offered by the vendor. This is a major path to learning the methods of digital forensics, but, since the training is usually focused on the tools available to the practitioners in their lab work environment, limitations on the vendor's tool or better tools might not be emphasized.

2.8.3 Digital Forensic Tool Documentation

In general, digital forensic tool vendors provide detailed documentation about using their products. Most tool vendors offer online support to help the user accomplish the goals of an investigation. This documentation can be found on the vendor website, but a service contract for the vendor's tools or a tool purchase may be required to access the documentation.

2.8.4 Materials and Guidelines Developed by Professional Organizations

Some digital forensics professional organizations publish guidelines and best practices for conducting a digital forensics examination. These include Scientific Working Group for Digital Evidence (SWGDE), Organization of Scientific Area Committees (OSAC-DE), High Technology Crime Investigation Association (HTCIA), International Association of

Computer Investigative Specialists (IACIS), European Network of Forensic Science Institutes (ENFSI), and International Society of Forensic Computer Examiners (ISFCE).

SWGDE develops best practices and other guidance for digital forensics practitioners. Some examples of SWGDE documents include best practices for mobile phone forensics [19], best practices for computer forensic examination, and other guidelines [19-28].

The OSAC Digital Evidence Subcommittee focuses on standards and guidelines related to information of probative value. OSAC-DE is in the process of adding documents to the OSAC document registry [29].

Both HTCIA and IACIS have large libraries of white papers on various techniques; these white papers are available to members.

ENFSI publishes a variety of documents related to digital forensic techniques. For example, they publish a best practices manual for digital forensics [30]. The manual covers a wide variety of topics including definitions of terms, validation of methods, estimation of uncertainty of measurement, proficiency testing, evidence handling, case assessment, reconstruction of events, evaluation and interpretation and other topics.

The ISFCE provides computer forensics certification based on a test to demonstrate proficiency in core digital forensics competencies. They also provide study materials that are useful references for the competency examinations.

2.8.5 Standards Organizations

Some standards organizations, e.g., American Society for Testing and Materials (ASTM) International and International Organization for Standardization (ISO), produce standards for digital forensics. Standards organizations usually require a fee to obtain a copy of a standard. Some ASTM Standards related to digital forensics include:

- E2678-09(2014) Standard Guide for Education and Training in Computer Forensics
- E2825-19 Standard Guide for Forensic Digital Image Processing
- E2916-19e1 Standard Terminology for Digital and Multimedia Evidence Examination
- E3016-18 Standard Guide for Establishing Confidence in Digital and Multimedia Evidence Forensic Results by Error Mitigation Analysis
- E3017-19 Standard Practice for Examining Magnetic Card Readers
- E3046-15 Standard Guide for Core Competencies for Mobile Phone Forensics
- E3115-17 Standard Guide for Capturing Facial Images for Use with Facial Recognition Systems
- E3148-18 Standard Guide for Postmortem Facial Image Capture
- E3149-18 Standard Guide for Facial Image Comparison Feature List for Morphological Analysis
- E3150-18 Standard Guide for Forensic Audio Laboratory Setup and Maintenance

Some ISO digital forensics standards include:

- ISO/IEC 27037:2012 — Information technology — Security techniques — Guidelines for identification, collection, acquisition, and preservation of digital evidence
- ISO/IEC 27041:2015 — Information technology — Security techniques — Guidance on assuring suitability and adequacy of incident investigative method
- ISO/IEC 27042:2015 — Information technology — Security techniques — Guidelines for the analysis and interpretation of digital evidence
- ISO/IEC 27043:2015 — Information technology — Security techniques — Incident investigation principles and processes
- ISO/IEC 27050:2018-2021 — Information technology — Security techniques — Electronic discovery (parts 1 through 4 published)

2.8.6 Online Videos

Training organizations, independent practitioners and forensic tool vendors produce online videos (usually found on YouTube) to illustrate digital forensic techniques. This is a valuable source of information for the forensic practitioner. As with other information sources, the examiner needs to be aware that the content can be out of date, misleading, incomplete, or inaccurate. Even with the caveat that the information may be flawed, it often provides new information for the examiner that can be verified when the examiners use their knowledge, skills, and experience and possibly some trial-and-error experiments to evaluate the usefulness of the presented material. This serves as an informal peer review of the described technique and can provide feedback to the developer.

2.8.7 Blog Posts

Blog posts are often created by practitioners who want to share techniques that they have developed. Like online videos the blog posts often provide useful information about emerging techniques that can help a practitioner extract or understand artifacts relevant to a specific investigation. However, the developed techniques are not usually formally peer-reviewed and risk being misleading, incomplete, or inaccurate. Like online videos the examiners must use their knowledge, skills, and experience and possibly some trial-and-error experiments to evaluate the usefulness of the presented material. This serves as an informal peer review of the described technique and can provide feedback to the developer.

2.8.8 Conference and Journal Articles

There are a several professional conferences that are devoted to digital forensics, including:

- Digital Forensics Research Workshop (DFRWS).
- Digital Forensics Research Workshop – Europe (DFRWS-EU).
- International Federation for Information Processing Working Group 11.9 (IFIP WG 11.9).
- American Academy of Forensic Sciences (AAFS).
- Association of Digital Forensics Security and Law (ADFSL).

The main journals that regularly include papers on digital forensics include:

- *Forensic Science International: Digital Investigation* (formerly *Digital Investigation*),
- *Journal of Forensic Sciences*,

- *Science and Justice,*
- *Australian Journal of Forensic Sciences,*
- *Journal of Digital Forensics, Security, and Law,*
- *International Journal of Computer Applications,*
- *Computers and Electrical Engineering, International Journal of Computer Science and Network Security,*
- *International Journal of Digital Crime and Forensics,* and
- *Security and Communication Networks.*

Additionally, the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE) publish research in digital forensics, especially topics related to computer security in their journals including:

- *ACM Computing Surveys,*
- *IEEE Cloud Computing,*
- *EEE Security and Privacy,*
- *IEEE Transactions on Information Forensics and Security, and*
- *IEEE Transactions on Software Engineering.*

2.8.9 Academic Course work

Colleges and universities offer undergraduate and graduate degrees and a wide variety of courses in digital forensics. This is where new practitioners get a basic education in digital forensics. Academic courses tend to be based on established best practices.

2.8.10 Reference Books

A wide variety of general digital forensics reference books are available [31-47].

In addition to general reference books, there are many specialty references for specific topics, such as operating systems [2, 48-57], and tool-specific reference books [58, 59]; and there are many reference books addressing specific issues such as child exploitation, malware, networks, corporate crime, and mobile devices [60-74].

2.8.11 Software Developer Documentation

Documentation about the internal organization of operating systems is often available and provides a rich source of information about trace artifacts that may be of forensic value. Documentation of individual applications and sometimes the source code of the applications might be available. As with any software documentation, it can be incomplete, out of date, misleading, misunderstood, or wrong and must be evaluated in light of actual observations of the running software or source code (if available).

2.8.12 Reverse Engineering Software

There might not be an obvious way to find and extract an artifact that would answer a question that has arisen during an investigation. In this case, the question could be skipped, or the examiner could resort to experiments to observe software behavior, often called reverse engineering. The process of reverse engineering to determine how something works may provide a better understanding of software activity under test conditions. For example, it might be informative if it can be shown that a mobile phone was connected to a particular vehicle. If it is suspected that the VIN (vehicle identification number) information is recorded on the phone, a similar phone could be connected to a known vehicle to test if that model device makes a record of the VIN. The test phone could then be searched for the VIN of the known vehicle, and the file where VIN data are stored could be identified. Then the evidence phone can be examined and (now that the file name and directory path to the VIN storage location are known) the VIN data file can be extracted and examined for a list of vehicles associated with the phone. This is a simplified example, but the same principles apply to reverse engineering how to find and access other artifacts. Since documentation of the latest software and hardware is often incomplete or nonexistent, forensic tool vendors often need to reverse engineer new software to understand its operation, including how to find and interpret available artifacts that may be informative to an investigation.

2.9 Summary of Sources

There is a wide variety of information sources available to examiners. Given the rapid pace of change in information technology, an examiner needs current information. This will necessarily include non-peer reviewed sources such as blogs and other internet postings. Examiners must, therefore, stay aware of the potential unreliability of their sources. Efforts in the community to provide rapid peer assessment of information sources should be promoted.

> **KEY TAKEAWAY #2.5:** Every digital forensic technique should undergo peer review, formal testing, or error rate analysis. The digital forensics community performs an informal review by providing timely feedback about the usefulness and validity of techniques through blogs, whitepapers, and videos on the internet. This general acceptance process allows for techniques to be quickly evaluated and revised. While this process is not comprehensive, it does provide significant benefits. Efforts to promote additional rapid peer assessment should be promoted.

3 Chapter 3: The Digital Forensic Data Sources Reviewed

The previous section addresses sources of information available to examiners. This section describes the key background information used for this report. The primary sources for understanding current practice in digital forensics were vendor tool documentation, peer-review literature, NIST Computer Forensic Tool Testing (CFTT) reports and discussions with forensic practitioners. This indicates the state-of-the art in digital investigations because, while forensic tools are not required for an investigation, an investigation proceeds very slowly and may be incomplete without tools. Secondary sources include:

- Application and operating system documentation that provides information on artifacts generated by the software,
- Textbooks that describe the data structures and operating system artifacts found on computer systems,
- Other books related to forensic science,
- Conference proceedings,
- Government documents,
- Standards documents, and
- Best practice documents.

The peer-reviewed literature within the digital forensics domain was used, but it generally does not focus on the scientific basis of fundamental tasks but instead on specific techniques for solving problems associated with forensic investigations.

Several other sources were also used, including:

- Informal discussions with digital forensics practitioners. The authors of this report have been performing research and tool testing activities in digital forensics for over 20 years and have interacted with practitioners at working groups, and conferences.
- Tool vendor experts. Documentation and interaction with digital forensics vendors are sources of information about the capabilities of tools and the vendors' perspectives about the needs of the field.
- Blogs. Several vendors, practitioners and researchers publish material online on an ad hoc basis. This information often addresses issues that have recently been discovered and potential solutions.

While assessing the available material on digital forensics, it became apparent that many important topics were not covered. Many of these, however, are covered by the computer science literature because the field of digital forensics intersects with a subset of computer science, much of the peer-reviewed material dedicated to digital forensics addresses specialized problems in the field.

This document draws on material from both digital forensics and general computer science.

4 Chapter 4: Scientific Foundations of Specific Tasks

There is no single technique that can be called "Digital Forensics." There are hundreds if not thousands of individual techniques that might be employed in a digital forensic examination. There are several useful models of a digital forensic examination, each with a different emphasis. We did not want to create yet another model, but we needed a way to classify the activities associated with digital data examinations. For purposes of this study, we classify the steps of a digital investigation as shown in **Fig. 8**:

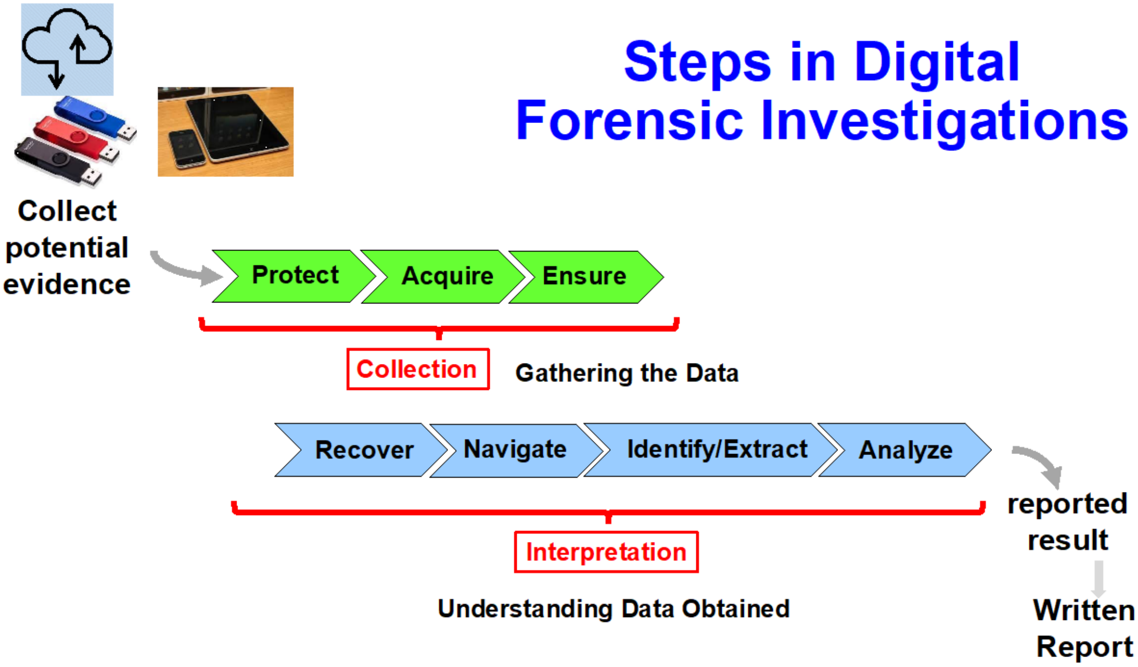


Fig. 8. Steps in a Digital Investigation.

The investigation begins with a triggering event that indicates a need for an investigation. This could be a suspected crime, a civil lawsuit, suspected employee misbehavior or another trigger. Depending on the type of event legal requirements vary and may require a search warrant or compliance with other legal requirements. Collection of potential evidence may include computers, mobile devices, storage devices, copies of data from cloud accounts and other sources. The collection steps ensure the integrity of the acquired evidence to provide a stable source for the analysis of the data and, if possible, protect the original data from accidental modification during the acquisition. The result of the digital investigation is a written report describing the findings of the digital analysis and may represent the bulk of the overall investigation or just a portion of a larger investigation.

The ability to demonstrate the reliability and validity of computer forensic tools based on scientific theory is an important requirement for digital evidence to be admissible.

This section discusses the foundations of the main digital forensic tasks, such as:

1. Protect original data from unintended modification. This is accomplished using a variety of approaches depending on the type of device that contains the data. This is discussed in Sec. 4.1.

2. Acquire digital data. This step is accomplished by copying data to an image file. Copying digital data accurately is based on established engineering techniques such as error detecting and correcting codes. This is discussed in Sec. 4.2.
3. Ensure integrity of acquired data. Cryptographic hashing is used to ensure that if acquired digital data are changed inadvertently or deliberately the change can be detected. This is discussed in Sec. 4.3.
4. Recover deleted data. In some situations, recovery and reconstruction of deleted data make it possible to bring back deleted files (in whole or in part) or internal records from within an application file. Recovery of deleted data has several risks including missing data and conflating unrelated data. Any recovered item must be evaluated by the examiner for indications of problems. This is discussed in Sec. 4.4.
5. Navigate the acquired digital data. This is accomplished by unraveling, i.e., parsing the layout of the acquired data. This is best performed using a software tool. There is the risk that an incorrect implementation will not correctly interpret the structure of a particular file system, e.g., not show all acquired active files. This is discussed in Sec. 4.5.
6. Identify and extract data artifacts. Items of interest are identified, located, and extracted. This is discussed in Sec. 4.6.
7. Analyze the artifacts. Examination of extracted artifacts can help develop a narrative or reconstruction of relevant events for inclusion in a final report. This is discussed in Sec. 4.7.

The following subsections discuss in more detail what occurs in each of the seven steps. Sec. 4.8 gives an overview of verification and validation for digital forensic tools and Sec. 4.9 discusses requirements for testing and validation of the techniques in each category.

4.1 Protecting Data During Acquisition

While protecting data happens as a part of acquisition, it is a sufficiently critical concern that it is normally addressed separately. There are several ways that data can be acquired. The data may be directly accessible from a storage device such as a hard drive or flash drive or may require more indirect techniques such as for cloud storage, mobile device memory or computer memory. This section discusses data protection techniques during acquisition from a storage device such as a hard drive, flash drive or other discrete device. Protection of original data for other situations such as mobile devices, computer memory, or cloud data is discussed with the acquisition technique.

Before any identified data can be copied (acquired) directly from a storage device, the device must be attached to a computer to access the data and make the copy. (This includes special-purpose hardware devices that only make a copy of the data on an attached device. These devices often have a built-in write blocker.) This can be a problem if the computer makes any changes to the device content before the copy operation takes place. There are several reasons this can happen, for example, the operating system may examine several files and thereby change the file access times as part of the startup process. The current approach is to use a device called a write blocker to monitor activity on the connection to suppresses any commands that might make a change to the device content [75]. However, write blockers do not guarantee that no changes occur to a storage device. Sometimes changes are triggered by the storage device itself. These changes are usually transparent to the user of the device and have no impact on user-created active data. For example, a counter is kept on some storage devices that is incremented every time the device is turned on (electrical power is applied). On solid-state drives, the wear-leveling algorithm moves data blocks to new hardware locations while not changing the logical address of the data block to balance the amount of usage for each block of solid-state storage and deleted data, marked with the TRIM command might be reset to all zero bits by the device at a later time after being marked.

In some situations, write blocking is not feasible, e.g., acquisition from a running system, acquisition of active memory, acquisition of remotely stored data, i.e., cloud data, or acquisition of data from a mobile device. In these situations, digital data are acquired imperfectly in that there are differences between the actual data present on the device and the acquired data. Some examples:

- When acquiring data from a running system or a remote system, other user activity may change file content during the acquisition.
- To acquire data from a mobile device, write blocking technology cannot usually be used; in addition, a tool might need to be loaded to the device to enable the acquisition. Of course, the tool overwrites the memory where it is loaded. See Sec. 4.2.2 for more details on acquiring data from a mobile device.
- For acquisition of data from computer memory, as with mobile devices, write blocking technology cannot be used and acquisition might require loading a tool into memory (overwriting a portion of existing memory).

4.2 Acquisition of Digital Data

In the early days of digital forensics, the acquisition of digital data focused on the contents of computer hard drives, floppy disks, and compact disc read-only memory (CD-ROMs). The process was referred to as disk imaging and usually included every byte of data on the device or selected partition. As digital storage devices have evolved it is more correct to refer to this process as digital data acquisition and may be a finer grained extraction, e.g., just selected files and metadata. This is the most fundamental task of digital forensics. The basic technique is to make a copy of the data to be examined. Copying data is a straightforward and reliable process ensured by error correcting codes [10] that are constantly employed by computers to ensure that a complete and accurate copy is produced.

The acquired data are usually placed into a container file that represents the acquired data; sometimes the source device is copied directly to another device (called a “clone”). There are currently more than 30 different container file formats in use to contain digital data [76]. The most widely used image formats are raw images (dd format) and e01 (Expert Witness) [77], but a number of other formats (usually specific to a tool vendor) are sometimes used. The need for a standard format has been recognized [78] and a standard, Advanced Forensic Format, has been proposed and is offered by some digital forensic tool vendors [79-82].

The procedures followed for acquiring digital data vary slightly for different types of devices such as hard drives, flash drives, mobile devices, remote data, and other devices. In addition, digital data can sometimes be acquired from social media.

4.2.1 Storage Device (Hard Drive & Flash Drive) Data Acquisition

One of the first commercial digital forensic tools was SafeBack, a tool to create a forensic image of a hard drive [83]. Various procedures were developed to attach a hard drive to a computer that can run the imaging tool in conjunction with a write blocker so that a copy can be made without modification.

There are many special cases in data acquisition based on hardware or the type of acquisition. Different computer hardware models have unique features that require special consideration. In situations where only part of the source data is desired, techniques for selective acquisition and management of the fragmented data have been developed [84], along with other projects to implement different selective acquisition tools and techniques [85].

Whenever possible, acquisition from a hard drive or similar device should be done in conjunction with either a hardware write-blocking device or a software write-blocking tool to avoid modification of the original data.

4.2.2 Mobile Device Acquisition

For mobile device forensics, there are many considerations and options for acquiring, protecting, and analyzing data [19, 21, 27]:

- Write blocking is not usually possible.
- The data stored on the device may change at any time.
- Mobile devices need to be isolated from the network to prevent changes from an external actor.
- Some methods require loading a tool to assist in acquiring the data stored on the device. Using such a tool overwrites some of the data on the device and those data are lost.

There are options for acquiring data from mobile devices.

- Logical acquisition: Extraction of a set of supported digital artifacts from the device. This is generally the easiest method. The original data are usually protected from modification because the tool only issues commands to extract data from the mobile device and does not attempt to write to the device.
- Selective acquisition: Extraction of a subset of supported digital artifacts from the device memory. This can be used to target specific data such as photos or contacts. Protection of original data is like that for a logical acquisition.
- File system acquisition: Extraction of the file system structure and content from the device. This allows acquisition of all data that are visible to the user. Protection of original data is like that for a logical acquisition.
- Physical acquisition: A copy of the device physical memory. Physical acquisition methods are categorized as either destructive or non-destructive. The chip-off extraction is a destructive method that involves removing the memory chip from the circuit board. JTAG is typically a non-destructive method that involves soldering wires to specific Test Access Ports (TAPs). This method may be destructive depending on the skill level of the examiner. A boot loader extraction is a non-destructive method that involves pushing a boot loader to the device memory providing the examiner with access to all memory. These methods are the most

complete and allows recovery of deleted data (SWGDE 2019b). However, due to the potential destructive nature of some of these processes, the original data are either lost or no longer easily accessible from the original device.

- Universal Integrated Circuit Card (UICC), also called a Subscriber Identity Module (SIM Card) acquisition: Extraction of the supported artifacts from a UICC.
- Some mobile devices support storage on a removable memory card. The original data can be protected with a write blocker or for some types of memory cards a built-in switch can turn on a read-only feature.

Each type of acquisition has advantages and limitations. Selection of an acquisition method depends on available tools and capabilities along with the make and model of device.

4.2.3 Remote and Cloud Data Acquisition

Remote acquisition of data, including cloud artifacts, over a live network has several unique challenges not found when acquiring from a single device that can be taken offline for examination. These issues include obtaining access to the remote computer and the infeasibility of using a write blocker in the acquisition. In fact, the data might change after the acquisition.

4.2.4 Other Device Data Acquisition

The embedding of digital devices into a variety of everyday items such as kitchen appliances, automobiles, smart watches, home security systems and other everyday items has given rise to forensics of the Internet of Things (IoT). Acquiring digital data from such devices is challenging and often requires destructive disassembly to acquire the data.

4.2.5 Social Media Data Acquisition

In addition to the acquisition of raw binary data from digital devices, a wealth of digital data can be harvested from social media. This can include contact lists, images, and locations visited.

4.3 Data Integrity Verification

After digital data have been acquired into an image file, steps should be taken to support later verification that no changes to the image file have occurred. Cryptographic hashing is a robust technique used in multiple high-security applications to detect inadvertent or deliberate changes. NIST publishes hashing standards as part of its cryptography program [86, 87]. The basic requirements for a cryptographic hashing algorithm are:

- Hash value can be computed quickly.
- It requires an unreasonable amount of computation to find two different files with the same hash value. This is defined as collision resistance and is useful because the digital data cannot be replaced with modified data and yet have the same hash.
- The original message cannot be recovered or reconstructed from the hash value.
- Any change to the original file brings about changes in the hash output value. On the average, a one-byte change to the original file causes about half of the bytes in the hash output to change.

4.4 Data Recovery

Since most operating systems do not immediately overwrite deleted data by default, these data can often be at least partially recovered. A complete file might or might not be reconstructed with the original content. In the situation where the storage device has had more than one owner, it is possible to recover data from previous owners, not just the current owner. This is one situation where apparently incriminating evidence can be found that has nothing to do with the current owner of the storage device.

There are three commonly used techniques for recovery of deleted data files in situations where the storage locations occupied by the deleted file is returned to the pool of data blocks available for allocation to a file:

- Metadata-based file recovery [88]. This technique exploits one design feature of file systems previously mentioned, those data are often not removed or overwritten when they are deleted. A notation is made to indicate that the data should not be seen and the storage space that they occupy can be reused. This file system metadata can help locate where the deleted data were stored.
- File carving [89]. Deleted files are identified by searching for data patterns that are unique to the beginning and end of files generated by some applications. That is, deleted data blocks are searched for these data patterns, possibly indicating deleted files. This technique is invoked when there might not be any metadata to guide recovery.
- Deleted record recovery [90]. Some applications (e.g., databases such as MySQL or SQLite or the Windows Registry) keep records (a set of related data values) that might be marked as deleted but not overwritten and have the potential for recovery. A recovery tool examines the internal data layout of an application file to identify deleted or updated data. Over time an application can add new records, update existing records, and delete some records. The application implementation can be exploited to identify and recover deleted data.

There are several considerations that have an impact on the quality of recovered data:

- If the deleted data have been overwritten or allocated to a new object, the deleted data cannot be recovered.
- Deleted data might be completely overwritten or only partially overwritten. It may not be possible to determine what data are original and what have been overwritten. The data presented as recovered might be mixed from several sources. The examiner can sometimes use context, metadata, and other clues to separate sources.
- Some file systems only preserve the location of the first storage block (FAT) when a file is deleted. Other file systems (e.g., NTFS) preserve more block locations, and other file systems (e.g., APFS) do not preserve any locations.
- Solid state drives might replace storage blocks marked by the OS (via a TRIM command) with a new block that only contains zero values. This could happen any time after the computer user has deleted a file, but before forensic acquisition of the device contents.

A tool must decide whether to include or not include certain data. Thus, different tool makers can offer similar deleted file recovery tools that nonetheless differ in what is recovered in particular "border line" or "edge" circumstances.

Recovered objects may have content from two or more sources. For example, a recovered photo may have content from two different pictures. In a document, there could be a shift of topic or some other unlikely shift, often within a sentence.

> **KEY TAKEAWAY #4.1:** When using techniques to recover deleted or hidden artifacts the examiner must determine the relevance of the recovered information as it may be incomplete or improperly merged with irrelevant information.

There are also situations where a file does not have the storage blocks returned to the available block pool and can still be recovered with no reassembly required. Some operating systems maintain a "trash can". When the user deletes a file, it is really just moved to the trash can directory. The file is not deleted until the user empties the trash can. Some Windows systems make an automatic backup snapshot of the file system, volume shadow copy, from time-to-time. Files that have been deleted can often be found in the volume shadow copy.

4.5 Parsing and Navigation

Once data have been acquired, the examiner needs to examine the acquired data. This is almost always done with some sort of interactive tool that presents the acquired data as seen in the original environment. The tool must recognize and interpret, i.e., parse, the data structures and metadata embedded in the acquired data so that the tool can navigate the file system to display content. It is challenging for tools to be able to parse the latest version of the file system or application files as well as all the older versions. Development of a parser for a file system frequently requires reverse engineering of the file system [91, 92] and then verification of the implementation. The common files systems are NTFS, ExFAT, FAT, ext4, HFS+, APFS, FAT and ExFAT. In addition, the tool needs to distinguish among the older file system versions, e.g., FAT comes in at least three major versions, 12-bit FAT, 16-bit FAT and 32-bit FAT [2]. The Linux file system also comes in ext2 and ext3.

Forensic tools may not support all file systems that might be encountered, e.g., the ExFAT is sometimes not supported. When new file systems are introduced by computer vendors there is usually a lag before the new file system is supported by forensic tools. If an unsupported file system is encountered, tools often treat the file system as unallocated space. Sometimes the support is incomplete or faulty at first. One of the most common failures is to not show all the object types. For example, NTFS has a feature to, in effect, have a collection of files under one name, this is called a primary data stream with multiple alternative data streams. A parsing tool might display only the primary data stream and ignore the alternative data streams. Some file systems have a feature called a link that allows more than one path through the directory tree to reach file content. There is potential for a defectively designed file system parser to produce incorrect results. The most likely impact is that the examiner would not see everything in the file system or see files in the wrong location. For example, if alternate data streams were not shown to the forensic tool user, then content within an alternative data stream would be overlooked.

Forensic tools must be designed to allow for application file data organization so that files representing complex objects such as documents, databases, or graphic files can be displayed. A faulty implementation can display the wrong data, not just fail to acquire some data.

4.6 Identification and Extraction of Artifacts

After parsing and navigating, the next step is to find items of interest. An examiner often follows an iterative process to answer questions arising in an investigation. The main assembly of a narrative to describe the events of interest of an investigation or answering questions that arise during an investigation involves identifying, finding, and extracting relevant artifacts. A question of interest might prompt an examiner to select a specific artifact for examination. The examiner then tries to locate the selected artifact and then extract the artifact for examination. Some methods to accomplish this are:

- Keyword search locates files that contain a specific string. Some files containing instances of a searched-for keyword might not be identified. Some situations where the keyword might not be found if the target string is:
    - in an encrypted file,
    - in a compressed file if the tool does not recognize compression and fails to expand the file and then search for the keyword,
    - represented with a text encoding method not searched for, e.g., only search UTF-8 but not search UTF-16, or
    - if text is in an application format that inserts formatting tags within words, e.g., inside the text of a word is a formatting tag to switch to bold font.
- Document retrieval locates files that discuss a specific topic.
- Metadata attribute matching locates files with metadata that match given criteria, e.g., file updated on a given date.
- Matching a given file property such as, a cryptographic hash of known contraband.
- Examining files known to contain specific content can identify needed information, e.g., contact list.
- Examining recovered files or recovered data records.

KEY TAKEAWAY #4.2: Searching tools have limitations based on the multiple ways that computers store information. Limitations include the type of files, types of encoding, and many other parameters. In general, digital search tools are very effective at finding information, but there is a possibility that data will be missed because a tool does not have the capability to find it.

Useful digital artifacts can be extracted in a variety of ways. The simplest way is directly extract known artifacts. Artifact lists are available to help practitioners interpret the significance of a given artifact [93, 94].

4.6.1 Locating Artifacts Indirectly

Sometimes a more indirect approach is required to locate the desired artifact. For example, consider a question such as “has a given mobile phone ever been connected to a specific vehicle?” When a mobile phone is connected to a car, it stores a record of the connection.

Each phone model may store the vehicle identification number (VIN) in a different place. Where a particular phone model stores this information is not always known. In such cases, examiners can try attaching the same model of phone to a test car and then search for that VIN number. That will probably reveal where the evidence phone stores the needed VIN number. In other cases, a pattern search can find files with strings in the same format as a VIN.

4.6.2 Locating Contraband

Cryptographic hashes can be used to identify known files from libraries of hashes of known files. There are collections of hashes of known contraband (e.g., child sexual abuse material). Other hash collection can be used to identify specific software packages, which, depending on context, could be significant. For example. A tool likely to be found in a system administrator or computer science researcher's tool kit might warrant further investigation if possessed by someone else. This is like finding lock picking tools. If in the possession of a locksmith, it is to be expected.

Identification of contraband can be accomplished in a variety of ways:

- Use a cryptographic hash of known contraband files to identify the presence of contraband. This method has a limitation in that the files must be identical. It does not identify files that are close, but not exact, matches.
- Use hashes of file fragments to identify isolated pieces of contraband files. This is sometimes able to detect deleted contraband.
- Use an approximate matching technique [95].
- Use DigitalDNA and similar methods to detect contraband images of children [37, 65, 96, 97].
- Use string searching to look for words, numbers or other text associated with the targeted contraband.

4.6.3 Other Examples of Locating Possibly Relevant Artifacts

There are many possible artifacts. Some of the more common locations where artifacts useful for an investigation might be found are:

- Memory. It is possible to retrieve artifacts from memory such as currently running programs and connections.
- Windows Registry. The Windows operating system keeps track of user activity and changes to hardware and software.
- File system metadata. File systems keep track of when files were created, opened, and modified.
- Email. Email contains not only messages, but attachments and timestamps for when email was sent and received and for the path it took.
- Internet activity. This includes browsing history, caches, and downloads.

There are several efforts to catalog artifact types. Some examples are the Artifact Genome Project at University of New Haven (See https://agp.newhaven.edu/about/start/) and the AXIOM Artifact Reference at Magnet Forensics [93, 94]. Additional projects are in progress.

There are many types of artifacts. For each type, an examiner needs to know what to look for and what it means. This can become quite complex. For example, the Windows Registry is designed for the Windows operating system to keep track of activity and to specify configurations and other system information. The meaning of each artifact needs to be understood within that version of an operating system or application.

4.7 Analysis of Results

There are several important considerations to evaluate results including:

- Considering and evaluating alternative hypothesis.
- Verifying that assembled pieces of the developed narrative are consistent, and any contradictions are noted.

Some examples of items that might be overlooked include:

- Does the examiner understand the meaning of each artifact relevant to an investigation?
- What steps have been taken to identify and mitigate human bias that might have crept into the work?
- Were anti-forensics employed to thwart any investigation?
- Is the temporal information accurate?
- Can artifacts and activities be attributed to a source such as the user of the machine or an external actor such as a hacker or malicious code, e.g., has false evidence been placed on the device by someone other than the user of the device?

4.7.1 Analysis Tools

There are several classes of analysis tools that can help an examiner understand the case data. Some examples include:

- A time-line tool can be used to put events into a temporal sequence to provide an overview of the relationships among events.
- Link analysis can look for relationships between entities in an investigation such as who is communicating with whom.
- Artificial Intelligence (AI) tools use a variety of techniques that allow the tool to improve performance over time. One such machine learning technique called *deep learning* can be used to uncover unseen relationships between case elements or search through data to recognize relevant items. Some AI applications have created negative impacts due to an inability to take different sources of bias into account [98]. One example is how facial recognition software can exhibit poor or misleading results for subjects with darker skin tones [99].

AI tools are powerful, but not perfect, and should be used with caution due to unexpected behaviors. An AI tool may inadvertently introduce bias from a variety of sources [98]. The tool outputs depend on the data set used to train the AI along with other factors and may not be relevant to the data at hand. Results could be misleading and should be verified or confirmed. As with other techniques, the examiner must use caution and check that AI-based findings are used in the appropriate context.

4.7.2 Anti-Forensics

There are many active measures a computer user can take to hide information. It is important to consider why the information is hidden; it could be done to ensure privacy, to intentionally cover up nefarious activity, or mislead investigators. The simplest method for hiding information is to delete incriminating files. However, deleted files might be recoverable. More effective anti-forensic techniques may be employed, such as using secure delete features of an operating system to overwrite deleted files. File wiping applications can also be employed to remove residual fragments. The presence of a file wiping application on a computer may indicate an effort to remove incriminating data.

Another common technique is to directly modify timestamps or set the system clock to the wrong time, to establish an alibi or create confusion in an investigation.

Other widely used methods to complicate [100] an investigation include deleting information from system log files (to remove a record of an event) or changing file MAC times (perhaps to create an alibi). Other methods attempt to hide data using file system properties [101] [102] or the system BIOS [103].

Steganography is a technique for hiding one set of data within another set of data. For example, pictures often have higher resolution than the viewer can perceive. So, it is possible to embed a hidden message in in the least significant bits of pixels representing a picture. Only the most significant bits make a difference in what is seen when viewing an image. There are many techniques for detecting [104] hidden data. For example, pixel values for color in an unmodified picture should cluster around the dominant colors; if the distribution of pixel values is, instead, uniform, it might suggest that something is hidden within the picture but is not human-perceptible.

> **KEY TAKEAWAY #4.3:** If someone has taken steps to change information in digital evidence to mislead an examiner, it may be difficult to detect the changes. Identification of deliberate obfuscating changes relies on the skill of the examiner.

4.8 Verification of Techniques and Validation of Tools

The industry standard for software verification and validation gives guidance to ensure that computer software correctly addresses the user needs and requirements [105]. However, a software vendor is not required to disclose the software development process followed during tool development.

When discussing tool testing, the forensic community needs to be aware of the usual meaning of the terms “validation” and “verification” within software engineering. In colloquial usage the terms verification and validation mean essentially the same thing: checking to see if something is correct. But, in a technical context there is an important difference.

For a forensic technique or method to be considered validated it should be shown to be fit for purpose, otherwise defined as “the process of providing objective evidence that the method is good enough to do the job required by the end user”. Validation alone can give a false indication of “fitness for purpose” that becomes apparent later.

Verification, on the other hand, is the demonstration that the implementation of the method correctly follows the tool design. It does not intend to show that the design is correct, but it may show that the implementation is incorrect.

Some examples to contrast verification and validation include:

- Consider building a tower. An engineer submits a design for a tower, it is reviewed and found to be like the design of other towers and approved for construction. That seemed to be a valid design. A contractor is hired, and work begins. At each step the contractor's work is checked and found to agree with the design. The tower is finished, but after completion the new tower begins to lean to one side. Although the design seemed valid, it is not fit for purpose at that building site. The soil conditions under one side of the tower are unable to support the weight of the tower. The design failed to account for this condition and should have been rejected. Another way to look at this is the design requirements were incomplete and something was missed. The builders verified that the construction (implementation) conformed to the tower design, but since the tower design was not fit for building on the selected site, the tower failed. This is one common way that the wrong tool gets built.
- Consider the scenario of selecting an algorithm for detecting if a digital object has changed (say, to verify image file integrity). This is an example of using validation to select an algorithm to implement that is fit for purpose. There are several candidates, e.g., CRC16, CRC32, MD4, MD5, SHA-1, SHA-2. The CRC algorithms have been used for decades to check whether a block of data has been transmitted without an error and was used in early imaging tools to verify image integrity [106]. The CRC is fit for detecting changes caused by random noise, however a malicious actor can easily modify a file in such a way that the CRC does not change. (This is called creating a collision.) Some additional requirements are needed for a hash algorithm to be fit for purpose in a forensic context:

    - Can be computed quickly.
    - It requires an unreasonable amount of computation to find two different files with the same hash value computed by the hash algorithm. This is defined as collision resistance.
    - Original message cannot be recovered or reconstructed from the hash value.
    - Any change to the original brings about changes in the hash output value.

  CRC does not meet all these criteria because CRC is not collision resistant. MD5 and SHA-1 were considered to meet these criteria until hash collision production algorithms were created for MD5 [107] and SHA-1 [108]. The work of Wang et. al. created concern about the use of MD5 and SHA-1, but these collision-creation algorithms are not relevant for digital forensic applications. [109]

  The SHA-2 and SHA-3 algorithms have been tested to meet these requirements and do not need to be further studied [86, 87]. However, a tool that computes either SHA-2 or SHA-3 needs to be verified to ensure that the implementation correctly computes the hash value.

There have been several papers published on validation of digital forensics methods [20, 110-121]. Some of these papers seem to confuse validation of a method and verification of a software tool and try to fold the two activities together instead of keeping them separate. The guidance from the UK Forensic Science Regulator [110] describes this well. It also includes consideration of risk assessment of the method, documentation of acceptance criteria, and possible outcomes.

The general validation and verification for a given version of a tool can be done once and shared. It does not need to be performed by every lab. The validation of a technique may need to be repeated when the implemented algorithm changes, e.g., to address changes in method to solve the intended forensic task. The implemented tool needs to be studied whenever the tool is changed or related technology changes. Each lab should ensure that personnel understand the basic capabilities and limitations of a tool, especially the relationship between the tool and the fast-changing information technology (IT) environment. However, it should be noted that the software environment for each forensic lab is slightly different and there may be differences in the outputs of the software tools.

4.9 Requirements for Testing Forensic Techniques

This section discusses digital forensic methods from the perspective of validation and verification. There are several approaches to show the reliability of a technique or that it is fit for purpose.

- An analysis or inspection of the algorithm to see whether it is sound and to identify potential limitations.
- The general intent of an algorithm may be known, but its details may be unknown. In this case, a direct analysis of the actual algorithm is not feasible, but an implementation can be tested to evaluate conformance to the intent of the algorithm.
- Part of the validation process should include an analysis of what can go wrong. This gives guidance for prioritizing and constructing test cases to evaluate an implementation.
- Implementations need to be tested to look for mistakes in the implementation and anomalies that occur within a given run time environment (hardware and operating system version).
- Testing is sometimes conducted to show that a technique can work and at other times to identify conditions when it does not work.

There are other considerations and important aspects of software that could and should be tested that can impact the tool's output or how the output is understood. Some important factors that are sometimes overlooked in tool testing include:

- Usability. A tool that is hard to use will be more likely to be misused. If tool outputs are not clearly presented to the user, they are more likely to be misunderstood.
- Security/software assurance. Forensic tools, like other software, can have bugs that may be exploited. While keeping forensic tools and processing off the internet is generally a good practice, there are other methods of attack. It is possible, for instance, for a person who anticipates being arrested to seed their own digital media

with code designed to trigger incorrect behavior from a forensics tool. There are no known instances of this happening in the real world, but it has been demonstrated.

- Processing speed. Digital forensic cases can be quite large. Some forensic techniques are computationally complex. The intersection of these factors can lead to very slow performance from tools for some operations. Tools that can process cases faster can have significant benefits for lab operations. Some digital forensic operations and processing, however, are hard to speed up.

4.10 Errors and Testing

This section discusses the meaning of error, error rates, and tool testing for digital forensics.

4.10.1 Error Rates

Some forensic disciplines use an error rate to describe the chance of false positives, false negatives, or otherwise inaccurate results when faced with a binary decision such as determining whether two samples come from the same source. But in digital forensics, there are fundamental differences in many processes that can make statistical error rates inappropriate or misleading.

The key point to keep in mind is the difference between random errors and systematic errors. Random errors may be characterized by error rates because they are based in the inability to perfectly measure natural processes. Systematic errors, in contrast, are caused by many different factors. In computer software, for example, an imperfect implementation can produce a correct result most of the time and an incorrect result every time a particular obscure condition, usually unknown, is met. Digital forensics – being based on computer science – is far more prone to systematic than random errors.

Digital forensics includes multiple tasks, which, in turn, use multiple types of automated tools. For each digital evidence forensic tool, there are underlying algorithms and implementations of the algorithms. There can be different errors and "error rates" with both the algorithm and the implementation. For example, hash algorithms used to determine whether two files are identical have an inherent false positive rate, but the rate is so small as to be essentially zero [86, 87, 122], while an error in the implementation of a hash algorithm might not manifest at all for some data sets but appear almost every time for other collections of data.

The classic concept of error rate as found in statistical hypothesis testing should apply to the intended algorithm of a statistical technique but does not usually apply to evaluating reliability of digital forensic tools [26, 123]. This is mostly due to the nature of the two activities. In hypothesis testing there is a simple binary decision. Something like "do two samples come from the same source with a given probability of a correct decision?" A digital forensics technique implementation in software may have multiple ways to fail with different risks associated with each failure mode ranging from significant to trivial. An error such as mislabeling a phone number as an email address could result in needed information not being found or be trivial since an examiner could easily correct this.

Another problem is that the properties and characteristics of digital data change with the software environment as the technology evolves over time and an error rate valid at one time might not apply at another time. This makes it impractical to develop error rates even for the random aspects of computer usage since the error rate will not apply to other cases.

> **KEY TAKEAWAY #4.4:** Digital processes tend to have systematic errors rather than random errors. Therefore, an error mitigation analysis provides more information and is the correct way to manage uncertainty. An error rate is only useful where there are random errors.

> **KEY TAKEAWAY #4.5:** When error rates are provided, it is important for the user to understand the context of the numbers. For some forensic techniques, the error rates may vary significantly based on attributes of the technology and usage patterns.

4.10.2 Observed Errors

The primary types of errors found in digital evidence forensic tool implementations are:

- Incompleteness: All relevant information has not been acquired or found by the tool. For example, an acquisition might be incomplete, or a search does not identify all existing relevant artifacts.
- Inaccuracy: The tool does not report accurate information. Specifically, the tool should not report artifacts that do not exist, should not group together unrelated items, and should not alter data in a way that changes the meaning. Assessment of accuracy in digital evidence forensic tool implementations can be categorized as follows:
  - Existence: Do all artifacts reported as present exist? For example, a faulty tool might add data that were not present in the original.
  - Alteration: Does a forensic tool alter data in a way that changes their meaning, such as updating an existing date-time stamp (e.g., associated with a file or e-mail message) to the current date?
  - Association: For every set of items identified by a given tool, is each item truly a part of that set? A faulty tool might incorrectly associate information pertaining to one item with a different, unrelated item. For instance, a tool might interpret a web browser history file incorrectly and report that a web search on "how to murder your wife" was executed 75 times when in fact it was only executed once while "history of Rome" (the next item in the history file) was executed 75 times, erroneously associating the count for the second search with the first search. There are many techniques to detect such errors such as peer review of the tool.
  - Corruption: Does the forensic tool detect and compensate for missing and corrupted data? Missing or corrupt data can arise from many sources, such as bad (unreadable) sectors encountered during acquisition or incomplete deleted file recovery or file carving. For example, a missing piece of data from an incomplete carving of the above web history file could also produce the same incorrect association.
- Misinterpretation: The results have been incorrectly understood. Misunderstandings of what certain information means can result from a lack of understanding of the underlying data or from ambiguities in the way forensic tools present information [26].

4.10.3 Software Testing (Tool Verification)

Doing software testing is like doing science. Just as Popper's [124] description of a scientific theory includes the idea that you cannot prove a theory is true, you can only disprove a theory or at least identify conditions where the theory does not apply. In keeping with the previous discussion of validation and verification is Sec 4.8 you cannot prove that a software program is correct by testing, you can only identify conditions where it fails.

There are several issues to note about testing digital forensic tools:

- There are no generally agreed requirements for each forensic task, i.e., no formal standard to test against.
- Tool vendors design, implement, and test their product but not transparently. For example, if a problem is reported to a vendor, the tool may be changed to fix the problem, silently with no notice given to tool users.
- Most forensic labs do not have sufficient resources to adequately test all tools used.
- The hardware environment of each lab is unique but testing a tool in one environment does not guarantee that the tool will behave the same in a different run time environment.

Other articles about digital forensic tool testing [20, 26, 112-119, 125-137] discuss various aspects of testing digital forensic tools.

> **KEY TAKEAWAY #4.6:** It is not feasible to test all combinations of tools, run time environments, and digital evidence sources.

4.10.4 NIST Tool Testing Results

NIST/CFTT [138, 139] develops tool specifications and test plans for testing various types of forensic tools, such as:

- Data Acquisition [140, 141],
- Write Blocking [142-144],
- Media Preparation [145, 146],
- File Carving [147, 148],
- Metadata-based Deleted File Recovery [149],
- Windows Registry [150, 151],
- Text String Searching[152, 153],
- SQLite Deleted Record Recovery [154] and
- Mobile Devices [155].

Department of Homeland Security (DHS) and National Institute of Justice (NIJ) have published forensic tool test reports for a variety of tool types.

NIST/CFTT has also published papers describing the testing techniques used by CFTT for write blocking [75, 156], general disk imaging [157], and imaging hard drives with faulty sectors[158].

There are NIST/CFTT digital forensic tool test reports for specific products and versions of:

- Disk Imaging and secondary storage acquisition tools [159-184],

- Write Blocking tools [185-231],
- File Carving tools [232-248],
- Metadata-based Deleted File Recovery tools [249-254],
- Windows Registry tools, [255, 256],
- Mobile Devices tools [257-318], and
- Key Word String Searching tools [319-324].

CFTT does not use the terms validation or verification but uses the term "tool testing" to avoid confusion about the terms. CFTT creates a requirements specification based on what tool vendors implement but considers it as descriptive of what the available tools do rather than a prescriptive specification of what they must do. CFTT tests for correct implementation against the CFTT-created specification.

The NIST/CFTT project has been testing tools since 2002. In general, the tools performed well with minor behaviors that needed to be kept in mind. Some examples from data acquisition testing include:

- Data acquisition might stop before all data on a device had been acquired. This was usually not a problem because the omitted data were not being offered to the computer user by the operating system for use. Typically, the operating system would group the basic unit of space on a hard drive (the 512-byte sector) into larger fixed-size blocks with some space left over. The left-over space could vary from a single sector [325] to over 5,000 sectors [326].
- The size as a count of sectors of a digital device can be reported several ways: BIOS size, visible size ignoring hidden sectors, visible sectors + size of host protected area (HPA), and visible sectors + HPA + device configuration overlay (DCO). These four sizes can all be different. A forensic tool may choose any size to use as the size to acquire. Using BIOS-reported size is obsolete now but in the late 1990s and early 2000s this was the preferred method since in allowed a BIOS-based software write blocker to be invoked to protect the computer hard drive from modification. After development of hardware write blockers, direct acquisition without risk of modification of the data became possible.
- As hard drives age some sectors may fail and become unusable. Sometimes during data acquisition, a sector may be unreadable and is reported as a bad sector. CFTT was able to develop and use techniques for testing tool behavior on encountering bad sectors [158]. Tools often omit readable sectors surrounding a bad sector, usually related to how the file system blocks disk sectors for the interface (USB, SATA, Firewire, etc.) used to access the hard drive.
- EnCase Version 4.22a [327] test results are an example of a tool having a small problem that sounds worse than it actually is. Seven sectors are imaged incorrectly, and one sector is omitted from the image. When the imaging tool FTK Imager 2.5.3.14 was tested on the same data set [168], the tool omitted all eight sectors. The reason for omitting the eight sectors is that the NTFS file system does not use these sectors to store any user data and should not contain any evidence. EnCase was trying to omit the data but made a mistake on when to stop writing the image file. From the EnCase Test Report [327]:

1. "If a logical acquisition is made of an NTFS partition, a small number (seven in the executed test) appear in the image file twice, replacing other sectors (DA–07–NTFS).
2. If a logical acquisition is made of an NTFS partition, the last physical sector of the partition is not acquired (DA–07–NTFS)."

The most serious failure ever observed was reported in 2003 for SafeBack Version 2.0 [328]. In one tool configuration, the acquired data of a SCSI drive were not the expected content and were incomplete. The tool gave no indication that there was a problem. A direct SCSI disk copy, using the Advanced SCSI Programming Interface (ASPI) driver for the SCSI adapter, copied only 2,097,270 sectors from a source disk with 17,921,835 sectors to an equal-size disk, leaving 15,824,565 sectors of the destination disk unchanged. SafeBack gave no indication of any problems and indicated a successful copy. An examiner might realize that the acquisition was corrupt but would not be able to continue the analysis. The vendor fixed the problem within two days of being notified.

NIST has developed several techniques for testing both software and hardware write blockers [329] that are widely used by digital forensic labs to verify operation of write blockers. NIST testing uses several techniques to generate traffic to see if a write blocker intercepts commands or lets them pass. Most write blockers on the market were able to block commands that would have changed a drive. The few exceptions were for uncommon commands or, in one case, where a vendor was unaware of a change to a chipset (which was quickly fixed). An analysis of the testing performed by NIST showed that write blocking is an effective technique. Special-purpose software write-blockers have been designed for situations such as virtual machines [330].

In general, the NIST test results often revealed minor anomalies. Typical results include:

- Sometimes acquisition tools miss data located in usually unused areas at the end of the device.
- Except for one model device, hardware write-block devices always blocked write commands. The firmware error for the one blocker that allowed write commands was quickly fixed by the vendor before CFTT tested the device.
- Deleted file recovery and file carving results need to be carefully examined and might contain missing data or data mixed from multiple sources. Any conclusions drawn from recovered files must be carefully evaluated.
- Mobile device results often have minor anomalies such as truncated strings and unsupported device models.
- Text string searching often misses some strings, especially for text encoded in Unicode 16-bit schemes.

While the analysis of write blockers performed by NIST and documented through test reports by NIJ and DHS shows that the technique has been demonstrated to be effective (see Sec. 4.10.4 on NIST Tool Testing), implementation and usage are critical. The tool must match the technology it is intended to be used with and be set up and used correctly. For example, some write-blocking devices use the same hardware for a write blocker as for a bridge used to switch between interfaces. If by mistake during a firmware upgrade the firmware for a bridge is uploaded to a write blocker then the device will no longer prevent changes to an attached storage device. This is a good example of why it is a good practice to retest forensic tools after an upgrade.

The analysis of string searching performed by NIST and documented through test reports by NIJ and DHS shows that the technique has been demonstrated to be effective at finding items. The test results demonstrated some systematic missing of text, e.g., Unicode 16-bit representation for languages with diacritical marks, but no false positives were observed. Sometimes two tools gave different search results for pre-defined search targets because the tools defined the targets differently. Implementation, usage, and analysis of results based on an understanding of the capabilities and limitations of the technique are critical.

> **KEY TAKEAWAY #4.7:** Extensive testing of over 250 widely used digital forensic tools showed that most tools perform their intended functions with only minor anomalies.

4.10.5 NIST Test Data Sets for Tool Testing

The CFReDS (Computer Forensic Reference Data Sets) project at NIST is a repository of digital storage device images. Examiners can use CFReDS in several ways including validating the software tools used in their investigations, checking that equipment is working properly, training examiners, and practicing use of forensic tools. Some images are produced by NIST, often from the CFTT (tool testing) project, and some are contributed by other organizations. The CFTT project posted its first document for public comment in March 2001, a specification for disk imaging including requirements and test assertions. CFTT submitted the first forensic tool test report, Red Hat GNU fileutils 4.0.36 dd, based on the final version of the specification, to the National Institute of Justice for publication in August of 2002.

The CFTT project approached forensic tool testing using a conformance testing model, often used to certify that a product conforms to a specific standard. The conformance testing model verifies that a product performs according to its specified standards. Because there were no published standards for forensic tools, CFTT wrote specifications for the tool functions they were tasked with testing. The tool specification included definitions of the function to be tested, a list of requirements the tool should meet, a list of test assertions to specify conformance to requirements, and a set of test cases to be run. CFTT also created software to create test data, test data sets, software to evaluate test case results, and procedures to follow when executing test cases. A formal test plan, test report and code review were published [331] for the test support software used with the disk imaging tool tests. No significant anomalies were found.

Before CFTT creates test data sets, CFTT first needs a tool function specification and a test plan with test cases. The steps for creating a data set are:

1. With the help of law enforcement representatives who advise the CFTT project, identify a forensic tool function for testing along with a list of candidate software or hardware tools.
2. Examine the selected tools and produce two lists of tool features offered: core features that are offered by all tools and optional features that are offered by some

tools. For example, all imaging tools are tested for acquiring an entire hard drive (a core feature), but only some tools support imaging of a single partition (an optional feature).
3. For each feature, create a list of requirements to specify what the feature is supposed to do.
4. Create a list of parameters that could impact tool behavior to help specify test cases. These may be tool settings that did not fit as tool features or run-time environment factors such as type of file system to be examined.
5. Create test cases based on test parameters.
6. After test cases are developed, test data sets must be created.

Testing is intended to find failures in tools. The more unique opportunities a tool is given to exhibit anomalies yet performs correctly, the more confidence in the correctness of tool results when a tool is used for a real investigation.

CFTT uses two data set creation approaches: static and on-the-fly. The static data sets are created so that it is convenient to make a disk image of the data and then the disk image can be imported into a forensic tool for examination. Creation of a test data set is often a combination of scripts and custom tools. The on-the-fly data sets are usually for some type of function that interacts directly with a device. Each test case has a set of procedures for preparing a device or test image for the test. In addition, there may be a set of custom tools to help evaluate the results for both test data creation approaches.

The on-the-fly testing usually follows this protocol:

1. Populate a device with test data designed to reveal anomalies by using custom tools and scripted user actions to set up the device.
2. Run the tool with the test data.
3. Examine and evaluate the results using custom tools. If the test case does not modify the device (e.g., hard drive), the device can be reused for testing another tool. Precautions are taken to back up the device in case it is modified when the test case is run, and it needs to be restored.

Tool functions at CFTT that use the on-the-fly approach include disk imaging, write blocking, forensic media preparation (drive wiping), and mobile device testing. Procedures for setting up test devices are posted on the CFTT web site along with a description of notable features of the data setup. For example, for disk imaging each sector of the device is given unique content that includes the LBA address of the sector. This allows easy diagnosis of misplaced sectors if a tool places an imaged sector in the wrong location in the image or places a given sector in the image more than once.

Sometimes writing the procedures is challenging because an unusual condition would require additional steps to finish the procedure. For example, when testing disk wiping, for a tool that allows use of the built-in security erase command it must be ensured that (1) the disk drive supports the security erase command and (2) the test computer BIOS does not disable the security feature set.

The static test data sets are usually provided as a set of small disk images for testing each tool function. The tool functions that use a static data set are file carving for graphic and video files, metadata-based deleted file recovery, and string searching.

For some test data sets, additional tools are needed to evaluate the test results. For example, it is often suggested to hash (MD5 or SHA-1) the result of deleted file carving or metadata-based deleted file recovery to see whether the file is correctly recovered. However, this only gives a yes or no answer and does not measure whether the recovered file is a total failure or a near miss. Rather than use an all-or-nothing measure, it is more useful to measure the quality of the recovered files. CFTT uses two measures for file carving: first, a visual evaluation to see whether the returned file can be viewed; second, an examination of the data returned to see how much of the original file is returned, how many data are omitted and how many are not from the original file. Both measures often reveal important aspects of the tool that a single measure did not show. One tool, for example, that was given a certain graphic file format returned an image file that did not produce a viewable image when displayed, but an examination of the returned data file revealed that the tool returned all the data except for the last block.

Some data sets do not need an evaluation tool. For example, each string search test case has a list of expected string instances that should be returned. Evaluating each test case is just a matter of comparing the list of expected hits to the actual hits returned. With the string search test cases, several test assertions are tested at the same time. For example, the string "DireWolf" appears 15 times in the CFTT test data set. Each instance of the target string is followed by a unique ID number so that an examination of a hit context confirms the actual instance returned. Some of the combined test assertions that can be tested:

- Find a string in an active file for each of seven file systems (FAT, ExFat, NTFS, ext4, HFS+ ignore case, HFS+ case sensitive, and APFS).
- Find a string in a deleted file for each of seven file systems.
- Find a string in unallocated space.

To give opportunities for testing to fail the CFTT data set also includes the strings "WOLF" (all caps), "Wolf" (mixed case), "wolf" (all lower case), and "WereWolf". These strings support several searching test assertions with enough strings that are almost matching to trigger some likely errors:

- Search for "wolf" with match case might fail by hitting "WOLF" or "Wolf" or "DireWolf."
- Search for "Wolf" as a whole word might fail by returning "WereWolf."

For testing UNICODE (UTF-8, UTF-16-BE and UTF-16-LE), 43 string instances are used. CFTT also considered types of character sets and includes Latin-based character sets with diacritic marks: Spanish, French, German, and Italian; A non-Latin character set: Russian; a right-to-left presentation: Arabic; and distinct Asian character sets: Chinese, Korean, and Japanese Kana. There were many other possibilities, but this covers most character set forms likely to occur.

Some considerations for constructing the CFReDS data sets include the following:

- It is often suggested that real-world data sets should be used. This has several advantages:
    - Fake real-world data sets can be constructed that are similar to real case data, but some things would likely be left out.
    - Creating a fake data set takes significant effort.
    - The data set is like the data that the forensic tool would encounter in investigative use.
    - The data set includes a large amount of noise, i.e., data that are not relevant to the investigation, that the tool must show that it can process successfully.
    - The actions of a computer user in the real world are in a random order and produce a variety of layouts so that the data set may include a situation that would cause the tool to fail. However, a constructed data set might not consider or include such a data layout.
- A real-world data set also has disadvantages:
    - An actual real-world data set from a real criminal case cannot be used without removing any personally identifiable information (PII) from the data set. It is difficult to accomplish the removal of PII and it would be easy to miss some data.
    - Fake real-world data sets are useful for training and evaluating investigators.
    - Data set ground truth is difficult to determine. The large amount of noise in the data is one factor in the difficulty.
    - Significant effort is required to obtain enough data sets so that there is coverage of all features included in the test plan.
    - Executing the test plan is time consuming when invoking the tool under test on several large image files.
    - The data sets are intended for sharing over the internet, and large image files take significant time to download.
    - Care must be taken to avoid including personally identifying information (PII) of the data set creator, i.e., social security numbers, credit card numbers, location information, etc.
- Constructed data sets can address the disadvantages:
    - It is easy to create data sets with known ground truth.
    - A constructed data set can focus on the features included in the test plan.
    - A constructed data set can be kept small.
    - Small data sets take much less time for a tool to scan and analyze.
    - Small data sets are quicker to download.

5 Chapter 5: The Digital Forensic Data Sources Reviewed

Digital investigation techniques are based on established computer science methods and are reliable when used with knowledge of how a tool functions and its limitations. The complexity and rapid change within the field do, however, introduce the possibility for incomplete analysis or for misunderstanding the meaning of artifacts.

Practitioners and stakeholders need to be aware of the following limitations with digital investigations:

- As with any crime scene, not all evidence may be discovered.
- When recovering deleted files, the results may include extraneous material.
- Examiners need to understand the meaning and significance of digital artifacts retrieved as they can change over different versions of operating systems or applications.

This analysis only addressed core digital evidence processes. It did not include several closely related areas such as network forensics, multimedia (audio, images, video) forensics, hacking, and malware analysis.

While developing this report, we encountered many areas that need further research and improved processes, including:

- Better sharing of forensic knowledge including new and changed artifacts, new techniques, tool limitations and workarounds, and other forensic insight. There are multiple blogs and other informal knowledge-sharing mechanisms, but a more structured approach would benefit the community.
- More efficient and consistent approaches to testing forensic tools. Currently, digital forensics labs are each testing the same tools with their own test data and requirements. This leads to varying test coverage and test results that are inconstant. A more structured approach could increase efficiency.
- Better sharing of forensic reference data. High-quality testing data are expensive to produce but are vital for tool testing, training and education, and research and development of new tools and techniques.
- Better analysis of how digital evidence is used and whether there have been incorrect or misleading conclusions. Having this information centrally collected would benefit the field.
- Better understanding of bias and effective bias minimization measures. Because of the nature of most digital evidence case work, forensic examiners are exposed to knowledge about people involved in a case, such as seeing their photos and reading their text messages. In addition, the forensic examiner may need to interact with an investigator.
- Better understanding of the types and characterization of mistakes examiners make in interpreting tool results.

The overall finding of this report is that digital evidence examination rests on a firm foundation based in computer science. Several of the techniques have already been extensively studied and documented in the peer-reviewed literature. Others are documented more informally through community discussion forums. The application of these computer science techniques to digital investigations is sound and only limited by the difficulties of keeping up with the complexity and rapid pace of change in IT.

References

[1] Knuth DE (1968) *The art of computer programming* (Addison-Wesley Pub. Co., Reading, Mass.,).

[2] Carrier B (2005) *File system forensic analysis* (Addison-Wesley, Boston, Mass. ; London), pp xx, 569 p.

[3] Butler J, Iyer H, Press R, Taylor MK, Vallone PM, Willis S (2020) NISTIR 8225: NIST Scientific Foundation Reviews. (NIST), 8225, December 2020. https://doi.org/https://doi.org/10.6028/NIST.IR.8225

[4] Guttman B, Laamanen MT, Russell C, Atha C, Darnell J (2022) *Results from a Black-Box Study for Digital Forensic Examiners* (National Institute of Standards and Technology, Gaithersburg, MD), National Institute of Standards and Technology.

[5] Inman K , Rudin N (2000) *Principles and practice of criminalistics : the profession of forensic science* (CRC Press, Boca Raton, Fla.), pp xx, 372 p.

[6] OSAC (2018) A Framework for Harmonizing Forensic Science Practices and Digital & Multimedia Evidence. February 2019. https://doi.org/http://dx.doi.org/10.29325/OSAC.TS.0002

[7] SWGDE (2016) SWGDE Digital & Multimedia Evidence Glossary Version 3.0.

[8] Cook R, Evett IW, Jackson G, Jones PJ, Lambert JA (1998) A hierarchy of propositions: deciding which level to address in casework. *Sci Justice* 38(4):231-239. https://doi.org/Doi 10.1016/S1355-0306(98)72117-3

[9] Cook R, Evett IW, Jackson G, Jones PJ, Lambert JA (1998) A model for case assessment and interpretation. *Sci Justice* 38(3):151-156. https://doi.org/Doi 10.1016/S1355-0306(98)72099-4

[10] Hamming RW (1950) Error Detecting and Error Correcting Codes. *Bell Syst Tech J* 29(2):147-160. https://doi.org/DOI 10.1002/j.1538-7305.1950.tb00463.x

[11] Silberschatz A, Galvin PB, Gagne G (2018) *Operating system concepts* (Wiley, Hoboken, NJ), 10th edition. Ed., p pages cm.

[12] Nikkel BJ (2009) Forensic analysis of GPT disks and GUID partition tables. *Digit Invest* 6(1-2):39-47. https://doi.org/10.1016/j.diin.2009.07.001

[13] IDEMA (2022) *Advanced Format Definitions, Abbreviations, and Conventions* (IDEMA). Available at http://idema.org/?page_id=2153.

[14] Ramsland KM (2016) *Confession of a serial killer : the untold story of Dennis Rader, the BTK killer* (ForeEdge, Hanover), pp x, 262 pages.

[15] Chao A, Tsay PK, Lin S-H, Shau W-Y, Chao D-Y (2001) The applications of capture-recapture models to epidemiological data. *Statistics in Medicine* 20(20):3123-3157.

[16] Chao A (1987) Estimating the population size for capture-recapture data with unequal catchability. *Biometrics* 43(4):783-791.

[17] Tilling K (2001) Capture-recapture methods - useful or misleading? *Int J Epidemiol* 30(1):12-14. https://doi.org/DOI 10.1093/ije/30.1.12

[18] Burch A, Durose M, Walsh K (2016) *Publicly funded forensic crime laboratories: Resources and services, 2014.*

[19] SWGDE (2016) Best practices for mobile phone forensics.

[20] SWGDE (2014) Recommended Guidelines for Validation Testing. *Scientific Working Group on Digital Evidence, version 20.*

[21] SWGDE (2016) Best practices for collection of damaged mobile devices.

[22] SWGDE (2017) Best practices for the acquistion of data from novel digital devices.

[23] SWGDE (2017) Best practices for maintaining the integrity of imagery.

[24] SWGDE (2018) Minimum requirements for testing tools used in digital and multimedia forensics.

[25] SWGDE (2018) Best practices for digital evidence collection.

[26] SWGDE (2018) Establishing confidence in digital forensic results by error mitigation analysis.

[27] SWGDE (2019) Best practices for mobile device evidence collection & preservation, handling, and acquisition.

[28] SWGDE (2019) Best practices for digital evidence acquisition from cloud service providers.

[29] National Institute of Standards and Technology (2021) *OSAC Registry*. Available at

[30] ENFSI (2015) Best Practice Manual for the Forensic Examination of Digital Technology.

[31] Casey E (2010) *Handbook of digital forensics and investigation* (Academic, Amsterdam ; Boston), pp xxvi, 567 p.

[32] Cowen D (2013) *Computer forensics : infoSec Pro guide* (McGraw-Hill, New York), pp xxiii, 318 pages.

[33] Britz M (2009) *Computer forensics and cyber crime : an introduction* (Pearson Prentice Hall, Upper Saddle River, N.J.), 2nd Ed., pp xxi, 340 p.

[34] Brown CLT (2006) *Computer evidence : collection & preservation* (Charles River Media, Hingham, Mass.), 1st Ed., pp xxii, 394 p.

[35] Davis C, Cowen D, Philipp A (2005) *Hacking exposed computer forensics : secrets & solutions* (McGraw-Hill/Osborne, New York), pp xxx, 444 p.

[36] Gardner RM (2012) *Practical crime scene processing and investigation* (CRC Press, Boca Raton, FL), 2nd Ed., pp xxxi, 466 p.

[37] Hayes DR (2015) *A practical guide to computer forensics investigations* (Pearson, Indianapolis, Indiana), pp xxi, 502 pages.

[38] Marcella AJ , Menendez D (2008) *Cyber forensics : a field manual for collecting, examining, and preserving evidence of computer crimes* (Auerbach Publications, New York), 2nd Ed., pp xxviii, 498 p.

[39] Nelson B (2015) *Guide to computer forensics and investigations : processing digital evidence* (Cengage Learning, Boston, MA), 5th edition. Ed., p pages cm.

[40] Philipp A, Cowen D, Davis C (2010) *Hacking exposed computer forensics* (McGraw-Hill/Osborne, New York), 2nd Ed., pp xxiv, 518 p.

[41] Rosenblatt KS (1995) *High-technology crime : investigating cases involving computers* (KSK Publications, San Jose, Calif.), pp xxiv, 603 p.

[42] Sammons J (2014) *The basics of digital forensics : the primer for getting started in digital forensics* (Elsevier, Waltham, MA), 2nd edition. Ed., p pages cm.

[43] Slade R (2004) *Software forensics : collecting evidence from the scene of a digital crime* (McGraw-Hill, New York), pp xvii, 215 p.

[44] Solomon M, Barrett D, Broom N (2005) *Computer forensics jumpstart* (Sybex, San Francisco), pp xvii, 283 p.

[45] Solomon M, Rudolph K, Tittel E, Broom N, Barrett D (2011) *Computer forensics jumpstart* (Wiley Publishing, Indianapolis, Indiana), Second edition. Ed., pp xx, 316 pages.

[46] Stephenson P (2000) *Investigating computer-related crime* (CRC Press, Boca Raton, Fla), p 304 p.

[47] Stephenson P , Gilbert K (2013) *Investigating computer-related crime* (Taylor & Francis, Boca Raton), 2nd Ed., pp xxiii, 380 p.

[48] Bar M (2000) *Linux internals* (McGraw-Hill, New York ; London), pp xv, 351 p.

[49] Carvey HA (2005) *Windows forensics and incident recovery* (Addison-Wesley, Boston), pp xvi, 460 p.

[50] Carvey HA (2014) *Windows forensic analysis toolkit : advanced analysis techniques for Windows 8* (Syngress, Amsterdam ; Boston), Fourth edition. Ed., pp xxi, 321 pages.

[51] Carvey HA (2016) *Windows registry forensics : advanced digital forensic analysis of the Windows registry* (Elsevier, Amsterdam), Second edition. Ed., pp xvi, 198 pages.

[52] Carvey HA , Casey E (2009) *Windows forensic analysis : DVD toolkit* (Syngress Pub., Burlington, MA), 2nd Ed., pp xxiv, 482 p.

[53] Honeycutt J (1996) *Using the Windows 95 registry* (Que, Indianapolis, IN), Special Ed., pp xx, 782 p.

[54] Honeycutt J (1998) *Using the Windows 98 registry* (Que, Indianapolis, Ind.), pp ix, 590 p.

[55] Honeycutt J (1998) *Windows 98 registry handbook* (Que, Indianapolis, Ind.), pp xx, 392 p.

[56] Honeycutt J (2000) *Microsoft Windows 2000 registry handbook* (Que, Indianapolis, Ind.), p 366 p.

[57] Honeycutt J (2003) *Microsoft Windows XP registry guide* (Microsoft Press, Redmond, Wash.), pp xxv, 497 p.

[58] Bunting S (2012) *EnCase computer forensics : the official EnCE : EnCase certified examiner study guide* (Wiley, Indianapolis, Ind.), 3rd Ed., pp xxxiv, 709 p.

[59] Casey E (2001) *Handbook of computer crime investigation : forensic tools and technology* (Academic Press, San Diego, Calif.), pp xiv, 448 p.

[60] Aquilina JM, Casey E, Malin CH (2008) *Malware forensics : investigating and analyzing malicious code* (Syngress Pub., Burlington, MA), pp xxxvi, 674 p.

[61] Bejtlich R (2006) *Extrusion detection : security monitoring for internal intrusions* (Addison-Wesley, Upper Saddle River, NJ), pp xxviii, 385 p.

[62] Caloyannides MA (2001) *Computer forensics and privacy* (Artech House, Boston, MA), pp xvii, 392 p.

[63] Caloyannides MA (2004) *Privacy protection and computer forensics* (Artech House, Boston), 2nd Ed., pp xix, 345 p.

[64] Casey E (2011) *Digital evidence and computer crime : forensic science, computers and the Internet* (Academic Press, Waltham, MA), 3rd Ed., pp xxvii, 807 p.

[65] Ferraro MM, Casey E, McGrath M (2005) *Investigating child exploitation and pornography : the Internet, the law and forensic science* (Elsevier/Academic Press, Amsterdam ; Boston, Mass.), pp xvi, 304 p.

[66] Jones KJ, Bejtlich R, Rose CW (2005) *Real digital forensics : computer security and incident response* (Addison-Wesley, Upper Saddle River, NJ), pp xxx, 650 p.

[67] Kipper G (2004) *Investigator's guide to steganography* (Auerbach Publications, Boca Raton, FL), pp xix, 220 p.

[68] Malin CH, Casey E, Aquilina JM (2012) *Malware forensics field guide for Windows systems : digital forensics field guides* (Syngress, Waltham, MA), pp xxxviii, 518 p.

[69] Malin CH, Casey E, Aquilina JM, Rose CW (2014) *Malware forensics field guide for Linux systems* (Elsevier, Amsterdam ;), pp xxxix, 574 pages.

[70] Mohay GM (2003) *Computer and intrusion forensics* (Artech House, Boston), pp xxi, 395 p.

[71] Reiber L (2019) *Mobile forensic investigations : a guide to evidence collection, analysis, and presentation* (McGraw-Hill Education, New York), Second edition. Ed., pp xviii, 542 pages.

[72] Sammes AJ , Jenkinson B (2000) *Forensic computing : a practitioner's guide* (Springer, London ; New York), pp xi, 295 p.

[73] Steel C (2006) *Windows forensics : the field guide for conducting corporate computer investigations* (Wiley Pub., Indianapolis, IN), pp xvii, 382 p.

[74] Williams HE (2006) *Investigating white-collar crime : embezzlement and financial fraud* (Charles C. Thomas, Springfield, Ill.), 2nd Ed., pp xiv, 347 p.

[75] Lyle J, Mead S, Rider K (2007) Disk drive I/O commands and write blocking. *Int Fed Info Proc* 242:163-+.

[76] Kim YR, S. (2012) Digital Forensics Formats: Seeking a Digital PreservationStorage Container Format for Web Archiving. *The International Journal of Digital Curation* 7(2):21-39. https://doi.org/10.2218/ijdc.v7i2.227

[77] Vandeven S (2014) Forensic images: For your viewing pleasure [White Paper]. (SANS Institute), 19 September 2014.

[78] Adelstein F, Carrier B, Casey E, Garfinkel SL, Hosmer C, Kornblum J, Lyle J, Rogers M, Turner P, Grp C (2006) Standardizing digital evidence storage. *Commun Acm* 49(2):67-68.

[79] Garfinkel S, Malan DJ, Dubec K-A, Stevens CC, Pham C (2006) Advanced forensic format: An open, extensible format for disk imaging. *Advances in Digital Forensics II: FIP International Conferences on Digital Forensics*, eds Olivier M & Shenoi S (Springer, New York), pp 17-31.

[80] Cohen M, Garfinkel S, Schatz B (2009) Extending the advanced forensic format to accommodate multiple data sources, logical evidence, arbitrary information and forensic workflow. *Digit Invest* 6:S57-S68. https://doi.org/10.1016/j.diin.2009.06.010

[81] Cohen M , Schatz B (2010) Hash based disk imaging using AFF4. *Digit Invest* 7:S121-S128. https://doi.org/10.1016/j.diin.2010.05.015

[82] Schatz BL (2015) Wirespeed: Extending the AFF4 forensic container format for scalable acquisition and live analysis. *Digit Invest* 14:S45-S54. https://doi.org/10.1016/j.diin.2015.05.016

[83] Pollitt M (2010) A History of Digital Forensics. *IFIP Advances in Information and Communications Technology* 337:3-15. https://doi.org/10.1007/978-3-642-15506-2_1

[84] Turner P (2006) Selective and intelligent imaging using digital evidence bags. *Digit Invest* 3:S59-S64. https://doi.org/10.1016/j.diin.2006.06.003

[85] Novak M, Grier J, Gonzalez D (2019) New approaches to digital evidence aquisition and analysis. *NIJ Journal* 280(January 2019):1-8.

[86] NIST (2015)– *FIPS 180-4 Secure Hash Standard*). https://doi.org/http://dx.doi.org/10.6028/NIST.FIPS.180-4

[87] NIST (2015)– *FIPS 202 SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*).

[88] Fellows GH (2005) The joys of complexity and the deleted file. *Digit Invest* 2(2):89-93. https://doi.org/10.1016/j.diin.2005.04.001

[89] Richard G, Roussev V, Marziale L (2007) In-place file carving. *Int Fed Info Proc* 242:217-+.

[90] Sanderson P (2018) *SQLite Forensics* (Independently Published).

[91] Nordvik R, Stoykova R, Franke K, Axelsson S, Toolan F (2021) Reliability validation for file system interpretation. *Forens Sci Int-Digit* 37. https://doi.org/ARTN 301174
10.1016/j.fsidi.2021.301174

[92] Nordvik R, Georges H, Toolan F, Axelsson S (2019) Reverse engineering of ReFS. *Digit Invest* 30:127-147. https://doi.org/10.1016/j.diin.2019.07.004

[93] Magnet Forensics (2021) AXIOM Artifact Reference 50.0.

[94] Magnet Forensics (2021) IEF Artifact Reference 6.48.0.

[95] Bjelland PC, Franke K, Arnes A (2014) Practical use of Approximate Hash Based Matching in digital investigations. *Digit Invest* 11:S18-S26. https://doi.org/10.1016/j.diin.2014.03.003

[96] Cifuentes J, Orozco ALS, Villalba LJG (2021) A survey of artificial intelligence strategies for automatic detection of sexually explicit videos. *Multimed Tools Appl*. https://doi.org/10.1007/s11042-021-10628-2

[97] Franqueira VNL, Bryce J, Al Mutawa N, Marrington A (2018) Investigation of Indecent Images of Children cases: Challenges and suggestions collected from the trenches. *Digit Invest* 24:95-105. https://doi.org/10.1016/j.diin.2017.11.002

[98] Schwartz R, Vassiev A, Greene K, Perine Lo, Burt A, Patrick HallEsperance S (2022) Towards a Standard for Identifying and Managing Bias in Artificial Intelligence - NIST SP 1270. (NIST), 1270, March 15, 2022. https://doi.org/https://doi.org/10.6028/NIST.SP.1270

[99] Grother P, Ngan M, Hanaoka K (2019) Face Recognition Vendor Test (FRVT) Part 3: Demographic Effects. NISTIR 8280, December 2019. https://doi.org/https://doi.org/10.6028/NIST.IR.8280

[100] Harris R (2006) Arriving at an anti-forensics consensus: Examining how to define and control the anti-forensics problem. *Digit Invest* 3:S44-S49. https://doi.org/10.1016/j.diin.2006.06.005

[101] Huebner E, Bem D, Wee CK (2006) Data hiding in the NTFS file system. *Digit Invest* 3(4):211-226. https://doi.org/10.1016/j.diin.2006.10.005

[102] Piper S, Davis M, Manes G, Shenoi S (2006) Detecting hidden data in Ext2/Ext3 file systems. *Advances in Digital Forensics* 194:245-+.

[103] Gershteyn P, Davis M, Manes G, Shenoi S (2006) Extracting concealed data from BIOS chips. *Advances in Digital Forensics* 194:217-+.

[104] Rodriguez B , Peterson G (2007) Detecting steganography using multi-class classification. *Int Fed Info Proc* 242:193-+.

[105] IEEE (2017)– *IEEE Standard for Software Verification and Validation - IEEE Std 1012* (IEEE).

[106] Peterson WW , Brown DT (1961) Cyclic Codes for Error Detection. *P Ire* 49(1):228-&. https://doi.org/Doi 10.1109/Jrproc.1961.287814

[107] Wang X , Yu H (2005) How to break MD5 and other hash functions. (Springer Berlin Heidelberg), pp 19-35.

[108] Wang X, Yin YL, Yu H (2005) Finding Collisions in the Full SHA-1. *CRYPTO 2005*, ed Shoup V (Springer, Cham), pp 17-36. https://doi.org/https://doi.org/10.1007/11535218_2

[109] Thompson E (2005) MD5 collisions and the impact on computer forensics. *Digit Invest* 2(1):36-40. https://doi.org/10.1016/j.diin.2005.01.004

[110] Regulator FS (2020) *Guidance: Method validation in digital forensics*.

[111] Arshad H, Jantan AB, Abiodun OI (2018) Digital forensics: Review of issues in scientific validation of digital evidence. *Journal of Information Processing Systems* 14(2):346-376.

[112] Beckett J , Slay J (2007) Digital forensics: Validation and verification in a dynamic work environment. *40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, (Waikoloa, HI), pp 266-276.

[113] Brunty J (2011) Validation of Forensic Tools and Software: A Quick Guide for the Digital Forensic Examiner. *Forensic Magazine*.

[114] Casey E (2011) The increasing need for automation and validation in digital forensics. *Digit Invest* 7(3-4):103-104. https://doi.org/10.1016/j.diin.2011.02.002

[115] Craiger P, Swauger J, Marberry C, Hendricks C (2006) Validation of Digital Forensic Tools. *Digital Crime and Forensic Science in Cyberspace,* (IGI Global, Hershey, PA, USA), pp 91-105.

[116] Guo YH, Slay J, Beckett J (2009) Validation and verification of computer forensic software tools-Searching Function. *Digit Invest* 6:S12-S22. https://doi.org/10.1016/j.diin.2009.06.015

[117] Horsman G (2018) "I couldn't find it your honour, it mustn't be there!" – Tool errors, tool limitations and user error in digital forensics. *Sci Justice* 58(6):433-440. https://doi.org/https://doi.org/10.1016/j.scijus.2018.04.001

[118] Horsman G (2019) Tool testing and reliability issues in the field of digital forensics. *Digit Invest* 28:163-175. https://doi.org/10.1016/j.diin.2019.01.009

[119] Marshall AM , Paige R (2018) Requirements in digital forensics method definition: Observations from a UK study. *Digit Invest* 27:23-29. https://doi.org/10.1016/j.diin.2018.09.004

[120] Risinger D (2018) The five functions of forensic science and the validation issues they raise: A piece to incite discussion on validation. *Seton Hall Law Review* 48:719-732.

[121] Wilsdon T , Slay J (2006) Validation of forensic computing software utilizing black box testing techniques. *4th Australian Digital Forensics Conference*, (Security Research Institute (SRI), Edith Cowan University, Edith Cowan University).

[122] NIST (2015)– *Secure Hash Standard*Washington, D.C.). https://doi.org/http://dx.doi.org/10.6028/NIST.FIPS.180-4

[123] Lyle JR (2010) If error rate is such a simple concept, why don't I have one for my forensic tool yet? *Digit Invest* 7:S135-S139. https://doi.org/10.1016/j.diin.2010.05.017

[124] Popper KR (1959) *The logic of scientific discovery* (Basic Books, New York,), p 479 p.

[125] Anobah M, Saleem S, Popov O (2014) Testing framework for mobile device forensics tools. *Journal of Digital Forensics, Security and Law* 9(2):221-234.

[126] Cusack B , Homewood A (2013) Identifying bugs in digital forensic tools. *Australian Digital Forensics Conference*. https://doi.org/10.4225/75/57b3c3befb86c

[127] Cusack B , Liang J (2011) Comparing the performance of three digital forensic tools. *Journal of Applied Computing and Information Technology* 15(1).

[128] Flandrin F, Buchanan WJ, Macfarlane R, Ramsay B, Smales A (2014) Evaluating Digital Forensic Tools (DFTs). *7th International Conference: Cybercrime Forensics Education and Training*.

[129] Garfinkel S (2012) Lessons learned writing digital forensics tools and managing a 30TB digital evidence corpus. *Digit Invest* 9:S80-S89. https://doi.org/10.1016/j.diin.2012.05.002

[130] Garfinkel S, Farrell P, Roussev V, Dinolt G (2009) Bringing science to digital forensics with standardized forensic corpora. *Digit Invest* 6:S2-S11. https://doi.org/10.1016/j.diin.2009.06.016

[131] Glisson WB, Storer T, Buchanan-Wollaston J (2013) An empirical comparison of data recovered from mobile forensic toolkits. *Digit Invest* 10(1):44-55. https://doi.org/10.1016/j.diin.2013.03.004

[132] Grajeda C, Breitinger F, Baggili I (2017) Availability of datasets for digital forensics - And what is missing. *Digit Invest* 22:S94-S105. https://doi.org/10.1016/j.diin.2017.06.004

[133] Guttman B, Lyle JR, Ayers R (2011) Ten Years of Computer Forensic Tool Testing. *Digital Evidence and Electronic Signature Law Review*:139-147.

[134] Hibshi H, Vidas T, Cranor L (2011) Usability of Forensics Tools: A User Study. *2011 Sixth International Conference on IT Security Incident Management and IT Forensics*, pp 81-91. https://doi.org/10.1109/IMF.2011.19

[135] James JI, Lopez-Fernandez A, Gladyhsev P (2014) Measuring accuracy of automated parsing and categorization tools and processes in digital investigations. *Digital Forensics and Cyber Crime ICDF2C 2013 Lecture Notes of the Instute for Computer Sciences, Social Informatics, and Telecommunications Engineering*, eds Gladyhsev P, Marrington A, & Baggili I (Springer, Cham), Vol. 132.

[136] McKemmish R (2008) When is Digital Evidence Forensically Sound? *Advances in Digital Forensics IV Digital Forensics 2008 IFIP - The International Federation for Information Processing*, eds Ray I & Shenoi S (Springer, Boston, MA), Vol. 285, pp 3-15.

[137] Yates M , Chi H (2011) A framework for designing benchmarks of investigating digital forensics tools for mobile devices. *Proceedings of the 49th Annual Southeast Regional Conference*, (Association for Computing Machinery, Kennesaw, Georgia), pp 179–184. https://doi.org/10.1145/2016039.2016088

[138] National Institute of Standards and Technology (2019) *Computer Forensics Tool Testing Program (CFTT)*. Available at https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt.

[139] National Institute of Standards and Technology (2020) *CFTT Federatied Testing Project*. Available at https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt/federated-testing.

[140] National Institute of Standards and Technology (2005) Digital data acquisition tool test assertions and test plan. Draft 1 of Version 1.0 Ed., November 10, 2005.

[141] National Institute of Standards and Technology (2004) Digital data aquisition tool specification. Draft 1 of Version 4.0 Ed., October 4, 2004.

[142] National Institute of Standards and Technology (2005) Hardware write blocker (HWB) assertions and test plan. Draft 1 for public comment of Version 1.0 Ed., March 24, 2005.

[143] National Institute of Standards and Technology (2004) Hardware write blocker device (HWB) specification Version 2.0. Version 2.0 Ed., May 19, 2004.

[144] National Institute of Standards and Technology (2003) Software write block tool specification & test plan. Version 3.0 Ed., September 1, 2003.

[145] National Institute of Standards and Technology (2009) Forensic media preparation tool test assertions and test plan. Draft 1 for Public Comment of Version 1.0 Ed., January 9, 2009.

[146] National Institute of Standards and Technology (2009) Forensic Storage Media Preparation Tool Specification. Draft 1 for Public Comment of Version 1.0 Ed., January 9, 2009.

[147] National Institute of Standards and Technology (2014) Forensic file carving tool specification Version 1.0. April 2014.

[148] National Institute of Justice , National Institute of Standards and Technology (2014) Forensic file carving tool test assertions and test plan v 1.0. April 2014.

[149] National Institute of Standards and Technology (2009) Active file identification & deleted file recovery tool specification. Draft 1 of Version 1.1 Ed., March 24, 2009.

[150] National Institute of Standards and Technology (2018) Windows registry forensic tool specification. Dract 2 of Version 1.0 for Public Comment Ed., June 2018.

[151] National Institute of Standards and Technology (2018) Windows registry forensic tool test assertions and test plan. Draft 2 of Version 1.0 for Public Comment Ed., June 2018.

[152] National Institute of Standards and Technology (2008) Forensic string searching tool requirements specification. Public Draft 1 of Version 1.0 Ed., April 24, 2008.

[153] National Institute of Standards and Technology (2018) Forensic string searching tool test assertions and test plan. Public Draft 1 of Version 1.0 Ed., 3/14/2018.

[154] National Institute of Standards and Technology (2021)– *SQLite data recovery specification, test assertions, and test cases*).

[155] National Institute of Standards and Technology (2019) Mobile Device Forensic Tool Test Specification, Test Assertions and Test Cases V3.0. May 2019.

[156] Lyle JR (2006) A strategy for testing hardware write block devices. *Digit Invest* 3:S3-S9. https://doi.org/10.1016/j.diin.2006.06.001

[157] Lyle J (2002) Testing Disk Imaging Tools. *DFRWS*, (Syracuse, NY).

[158] Lyle JR , Wozar M (2007) Issues with imaging drives containing faulty sectors. *Digit Invest* 4:S13-S15. https://doi.org/10.1016/j.diin.2007.06.002

[159] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital acquisition tool: Dc3dd v7.2.61. October 2016.

[160] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for digital daa acquisition tool - Paladin v2.06. March 2013.

[161] National Institute of Justice , National Institute of Standards and Technology (2009) Test results for digital data acquisition tool - BlackBag MacQuisition v2.2. September 200.

[162] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for digital data acquisition tool - DCCIdd (Version 2.0). January 2008.

[163] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for digital data acquisition tool - DCFLDD v1.3.4-1. December 2013.

[164] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for digital data acquisition tool - EnCase Linen v5.05f. January 2008.

[165] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for digital data acquisition tool - EnCase LinEn v6.01.

[166] National Institute of Justice , National Institute of Standards and Technology (2009) Test results for digital data acquisition tool - EnCase v6.5.

[167] National Institute of Justice , National Institute of Standards and Technology (2013) Test results for digital data acquisition tool - FTK Imager CLI 2.9.0 Debian. May 2013.

[168] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for digital data acquisition tool - FTK Imager v2.5.3.14. June 2008.

[169] National Institute of Justice , National Institute of Standards and Technology (2011) Test results for digital data acquisition tool - Image MASSter Solo-3 Forensics; Software Version 2.0.10.23f. December 2011.

[170] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for digital data acquisition tool - Image MASSter Solo-4 Forensic. November 2013.

[171] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for digital data acquisition tool - IXImager v3.0.nov.12.12. November 2013.

[172] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for digital data acquisition tool - MacQuisition v2013R2. July 2014.

[173] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for digital data acquisition tool - Paladin v4.0. May 2014.

[174] National Institute of Justice , National Institute of Standards and Technology (2011) Test results for digital data acquisition tool - Tableau TD1 Forensic Duplicator; Firmware v2.34 2/17/2011. December 2011.

[175] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for digital data acquisition tool - Tableau TD3 Forensic Imager v1.3.0. July 2014.

[176] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for digital data acquisition tool - X-Ways Forensics 16.2 SR-5. November 2013.

[177] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Logicube Forensic Falcon v3.OU1RC13. October 2016.

[178] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Guymager v0.8.1. October 2016.

[179] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Logicube Forensic Falcon v2.4u1. October 2016.

[180] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Paladin v6.08. October 2016.

[181] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Paladin v6.09. October 2016.

[182] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: Tableau TD2u Firmware v1.1.2.3948-4270f9c. October 2016.

[183] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: WiebeTech Ditto Forensic FieldStation v2016Mar01a. October 2016.

[184] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for digital data acquisition tool: X-Ways Forensics v18.8. October 2016.

[185] Department of Homeland Security , National Institute of Standards and Technology (2020) Test results (federated testing) for hardware write block device - CRU Forensic UltraDock FUDv5.5 Firmware Version f3.01.0011. March 2020.

[186] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - Digital intelligence Firefly 800 IDE (FireWire Interface). April 2006.

[187] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - Digital Intelligence UltraBlock SATA (FireWire interface). May 2006.

[188] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - Digital Intelligence UltraBlock SATA (USB Interface). April 2006.

[189] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - FastBloc FE (FireWire Interface). June 2007.

[190] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - FastBloc FE (USB Interface). June 2007.

[191] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - FastBloc IDE (Firmware Version 16). April 2006.

[192] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Forensic ComboDock FCDv5.5. October 2018.

[193] National Institute of Justice , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Forensic ComboDock v5. October 2018.

[194] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Forensic LabDock U5. October 2018.

[195] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Forensic UltraDock FUDv5.5. October 2018.

[196] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - ICS ImageMasster DriveLock IDE (Firmware Version 17). April 2006.

[197] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Media WriteBlocker. October 2018.

[198] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - MyKey NoWrite (Firmware Version 1.05). April 2006.

[199] National Institute of Justice , National Institute of Standards and Technology (2009) Test results for hardware write block tool - T4 Forensic SCSI Bridge (FireWire Interface). September 2009.

[200] Department of Homeland Security , National Institute of Standards and Technology (2009) Test results for hardware write block tool - T4 Forensics SCSI Bridge (USB Interface). September 2009.

[201] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau eSATA Forensic Bridge T35es-R2. October 2018.

[202] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic FireWire Bridge T9. October 2018.

[203] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - Tableau Forensic IDE Pocket Bridge T14 (FireWire Interface). January 2007.

[204] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic PCle Bridge T7u. October 2018.

[205] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic SAS Bridge T6es-B. October 2018.

[206] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic SAS Bridge T6u. October 2018.

[207] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - Tableau Forensic SATA Bridge T3u (Firewire Interface). January 2007.

[208] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - Tableau Forensic SATA Bridge T3u (USB Interface). January 2007.

[209] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic SATA/IDE Bridge T35u. October 2018.

[210] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic Universal Bridge T356789IU. October 2018.

[211] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic USB 3.0 Bridge T8u. October 2018.

[212] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - Tableau Forensic USB Bridge T8-R2. October 2018.

[213] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - Tableau T5 Forensic IDE Bridge (FireWire Interface). June 2007.

[214] National Institute of Justice , National Institute of Standards and Technology (2007) Test results for hardware write block tool - Tableau T5 Forensic IDE Bridge (USB Interface). June 2007.

[215] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for hardware write block tool - Tableau T8 Forensic USB Bridge (FireWire Interface). August 2008.

[216] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for hardware write block tool - Tableau T8 Forensic USB Bridge (USB Interface). August 2008.

[217] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - UltraBlock USB 3.0 Forensic Card Reader. October 2018.

[218] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for hardware write block tool - USB WriteBlocker. October 2018.

[219] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - WiebeTech FireWire DriveDock Combo (FireWire Interface). April 2006.

[220] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - WiebeTech Forensic ComboDock (USB Interface). May 2006.

[221] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - WiebeTech Forensic SATADock (FireWire Interface). December 2006.

[222] National Institute of Justice , National Institute of Standards and Technology (2006) Test results for hardware write block tool - WiebeTech Forensic SATADock (USB Interface). December 2006.

[223] National Institute of Justice , National Institute of Standards and Technology (2005) Test results for software write block tools - PDBLOCK v1.02 (PDB LITE). June 2005.

[224] National Institute of Justice , National Institute of Standards and Technology (2005) Test results for software write block tools - PDBLOCK Version 2.00. June 2005.

[225] National Institute of Justice , National Institute of Standards and Technology (2005) Test results for software write block tools - PDBLOCK Version 2.10. June 2005.

[226] National Institute of Justice , National Institute of Standards and Technology (2004) Test results for software write block tools - RCMP HDL VO.4. August 2004.

[227] National Institute of Justice , National Institute of Standards and Technology (2004) Test results for software write block tools - RCMP HDL VO.5. August 2004.

[228] National Institute of Justice , National Institute of Standards and Technology (2004) Test results for software write block tools - RCMP HDL VO.7. August 2004.

[229] National Institute of Justice , National Institute of Standards and Technology (2004) Test results for software write block tools - RCMP HDL VO.8. February 2004.

[230] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for software write block tools - Writeblocker Windows 2000 V5.02.00. January 2008.

[231] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for software write block tools - Writeblocker Windows XP V6.10.0. January 2008.

[232] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - Androit Photo Forensics 2013 v3.1d. July 16, 2014.

[233] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - EnCase Forensic v6.18.0.59. July 16, 2014.

[234] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - EnCase Forensic v7.09.05. July 16, 2014.

[235] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - FTK v4.1. July 16, 2014.

[236] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - iLook v2.2.7. July 16, 2014.

[237] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - PhotoRec v7.0-WIP. July 16, 2014.

[238] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - R-Studio v6.2. July 16, 2014.

[239] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - Recover My Files v5.2.1. July 16, 2014.

[240] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - Scalpel v2.0. July 16, 2014.

[241] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for graphic file carving tool - X-Ways Forensics v17.6. July 16, 2014.

[242] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for video file carving tool - EnCase v7.09.05. October 22, 2014.

[243] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - iLook v2.2.7. October 22, 2014.

[244] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - PhotoRec v7.0-WIP. October 22, 2014.

[245] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - R-Studio v6.2. October 22, 2014.

[246] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - Recover my Files v5.2.1. October 22, 2014.

[247] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - Scalpel v2.0. October 22, 2014.

[248] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for video file carving tools - X-Ways v17.6. October 22, 2014.

[249] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools - FTK v3.3.0.33124. June 23, 2014.

[250] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools - ILooKIX v2.2.3.151. September 22, 2014.

[251] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools - SMART for Linux v2011-02-02. June 30, 2014.

[252] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools - the Sleuth Kit (TSK)/Autopsy v3.2.2/2.24. July 2, 2014.

[253] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools - X-Ways Forensics v16.0 SR-4. July 2, 2014 Ed.

[254] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for deleted file recovery and active file listing tools (revised) - EnCase Forensic v6.18.0.59. June 23, 2014.

[255] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results for Windows registry forensic tool - EnCase forensic 8.07.00.93 (x64). April 2019.

[256] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results for Windows registry forensic tool - forensic Toolkit (FTK) 7.0.0.163 (x64). April 2019.

[257] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results mobile device acquisition tool - E3-DS v2.2.118.12.15844. October 2019.

[258] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acqusition tool - Blacklight 2018 Release 1.1. November 2018.

[259] National Institute of Justice , National Institute of Standards and Technology (2010) Test results for mobile device acquition tool - Secure View 2.1.0. November 2010.

[260] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for mobile device acquistion tool - Paraben Device Seizure 2.1. October 2008.

[261] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for mobile device acquisition tools - viaExtract v2.5. December 2014.

[262] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for mobile device acquisition tools - UFED Physical Analyzer v3.9.6.7. October 2014.

[263] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tools - Oxygen Forensic Suite 2015 - Analyst v7.0.0.408. March 2015.

[264] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for mobile device acquisition tools - Mobile Phone Examiner Plus v5.5.3.73. December 2014.

[265] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tools - Lantern v4.5.6. June 2015.

[266] Department of Homeland Security , National Institute of Standards and Technology (2014) Test results for mobile device acquisition tools - iOS Crime Lab v1.0.1. December 2014.

[267] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tools - EnCase Smartphone Examiner v7.10.00.103. April 2015.

[268] Department of Homeland Security , National Institute of Standards and Technology (2013) Test results for mobile device acquisition tools - EnCase Smartphone Examiner v7.0.3. April 2013.

[269] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tools - Device Seizure v6.8. June 2015.

[270] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool -UFED Touch/ Physical Analyzer v6/2/1/17/ v6.3.0.284. January 2018.

[271] National Institute of Justice , National Institute of Standards and Technology (2010) Test results for mobile device acquisition tool - Zdziarski's Method. December 2010.

[272] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - XRY v7.8.0. November 2018.

[273] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - XRY v7.3.1. August 2017.

[274] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - XRY v7.0.1.37853. November 2016.

[275] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - XRY Kiosk v7.8.0. November 2018.

[276] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - MD-NEXT v1.75.20171226.28830D4,MD-RED v2.3.20171226.28828D6. April 2018.

[277] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - XRY Kiosk v7.0.0.36568. January 2017.

[278] National Institute of Justice , National Institute of Standards and Technology (2010) Test results for mobile device acquisition tool - XRY 5.0.2. November 2010.

[279] Department of Homeland Security , National Institute of Standards and Technology (2017) Test resutls for mobile device acquisition tool - Lantern v4.6.8. July 2017.

[280] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - UFED Touch v4.4.0.1 Internal Build 4.2.8.36. July 11, 2016.

[281] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results for mobile device acquisition tool - UFED InField Kiosk v7.50.0875. October 2019.

[282] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results for mobile device acquisition tool - UFED 4PC v7.8.0.942/Physical Analyzer v7.9.0.223. April 2019.

[283] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - UFED 4PC v4.2.6.5 - Physical Analyzer v4.2.6.4. January 6, 2016.

[284] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - Secure View v4.3.1. November 2017.

[285] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - Secure View v4.1.9. July 2016.

[286] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tool - Secure View v3.16.4. February 2015.

[287] National Institute of Justice , National Institute of Standards and Technology (2013) Test results for mobile device acquisition tool - Secure View 3v3.3.8.0. February 2013.

[288] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tool - Phone Forensics Express v2.1.2.2761.

[289] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - Oxygen Forensics v10.0.0.81. April 2018.

[290] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - Oxygen Forensics v8.3.1.105. August 2016.

[291] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - Mobilyze v2018.1. November 2018.

[292] National Institute of Justice , National Institute of Standards and Technology (2011) Test results for mobile device acquisition tool - Mobilyze v1.1. January 2011.

[293] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - MOBILedit forensics v9.1.0.22420. January 2018.

[294] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - MOBILedit Forensics Express v4.2.1.11207. November 2017.

[295] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - MOBILedit Forensic v8.6.0.20354 November. November 2016.

[296] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - MOBILedit Forensic v8.6.0.20354. December 2016.

[297] Department of Homeland Security , National Institute of Standards and Technology (2015) Test results for mobile device acquisition tool - MOBILedit Forensic v7.8.3.6085. December 2015.

[298] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - MOBILedit Forensic Express v3.5.2.7047. March 2017.

[299] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - Mobile Phone Examiner Plus v5.6.0. March 2017.

[300] National Institute of Justice , National Institute of Standards and Technology (2012) Test results for mobile device acquisition tool - Mobile Phone Examiner Plus (MPE+) 4.6.0.2.

[301] National Institute of Justice , National Institute of Standards and Technology (2013) Test results for mobile device acquisition tool - Micro Systemation XRY v6.3.1. February 2013.

[302] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for mobile device acquisition tool - Micro Systemation .XRY 3.6. October 2008.

[303] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - Magnet AXIOM v1.2.1.6994. October 2018.

[304] National Institute of Justice , National Institute of Standards and Technology (2013) Test results for mobile device acquisition tool - Lantern v2.3. February 2013.

[305] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - Katana forensics Triage v1.1802.220. May 2018.

[306] National Institute of Justice , National Institute of Standards and Technology (2010) Test results for mobile device acquisition tool - iXAM Version 1.5.6. December 2010.

[307] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for mobile device acquisition tool - guidance software Neutrino 1.4.14. October 2018.

[308] Department of Homeland Security , National Institute of standards and Technology (2019) Test results for mobile device acquisition tool - GrayKey OS Version 1.4.2 App Bundle 1.11.2.5. June 2019.

[309] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - Final mobile forensics v2018.02.07. July 2018.

[310] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - Electronic Evidence Examiner Device Seizure v1.0.9466.18457. April 2017.

[311] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for mobile device acquisition tool - electronic evidence examiner - device seizure (E3:DS) v1.7. June 2018.

[312] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - Device Seizure v7.4 build 5921.15166. May 2016.

[313] National Institute of Justice , National Institute of Standards and Technology (2013) Test results for mobile device acquisition tool - Device Seizure v5.0 build 4582.15907. February 2013.

[314] National Institute of Justice , National Institute of Standards and Technology (2012) Test results for mobile device acquisition tool - CellBrite UFED 1.1.8.6-Report Mgr 1.8.3/UFED Physical Analyzer 2.3.0. September 2012.

[315] Department of Homeland Security , National Institute of Standards and Technology (2017) Test results for mobile device acquisition tool - Blacklight v2016.3.1. December 2017.

[316] Department of Homeland Security , National Institute of Standards and Technology (2016) Test results for mobile device acquisition tool - BlackLight v2016.1. May 2016.

[317] National Institute of Justice , National Institute of Standards and Technology (2010) Test results for mobile device acquisition tool - BitPim - 1.0.6. January 2010.

[318] Department of Homeland Security , National Institute of Standards and Technology (2019) Test results for binary image tool - final mobile forensics v2019.07.05. July 2019.

[319] Department of Homeland Security , National Institute of Standards and Technology (2018) Test results for string search tool - Autopsy version 4.6.0. November 2018.

[320] Department of Homeland Security , National Institute of Standards and Technology (2019) Test resutls for string search tool - X-Ways Forensics Version 19.6-SR-4 x64. March 2019.

[321] Department of Homeland Security , National Institute of Standards and Technology (2020) Test results for string search tool - BlackLight Version 2018-R4. March 2020.

[322] Department of Homeland Security , National Institute of Standards and Technology (2020) Test results for string search tool - EnCase Version 8.09.00.192.

[323] Department of Homeland Security , National Institute of Standards and Technology (2020) Test results for string search tool - Magnet Axiom Version 4.1.1.20153.

[324] Department of Homeland Security , National Institute of Standards and Technology (2020) Test results for string search tool - Access Data Forensic Toolkit (FTK) Version 7.0.0.163.

[325] National Institute of Justice , National Institute of Standards and Technology (2002) Test results for disk imaging tools: dd GNU fileutils 4.0.36, Provided with Red Hat Linux 7.1. August 2002.

[326] National Institute of Justice , National Institute of Standards and Technology (2002) Test results for disk imaging tools: SafeBack 2.18. June 2003.

[327] National Institute of Justice , National Institute of Standards and Technology (2008) Test results for digital data acquisition tool: EnCase 4.22a. January 2008.

[328] National Institute of Justice , National Institute of Standards and Technology (2003) Partial results from prototype testing effors for disk imaging tools: SafeBack 2.0. April 2003.

[329] Lyle JR (2006) A strategy for testing hardware write block devices. *Digit Invest*:S3-S9. https://doi.org/10.1016/j.diin.2006.06.001

[330] Tobin P, Le-Khac NA, Kechadi MT (2016) A Lightweight Software Write-blocker for Virtual Machine Forensics. *2016 Sixth International Conference on Innovative Computing Technology (Intech)*:730-735.

[331] Gavrila S , Fong E (2004) Forensic Software Testing Support Tools Test Summary Report. (NIST), NISTIR 7103-B, April 2004.

Appendix A. Glossary, List of Symbols, Abbreviations, and Acronyms

AAFS
American Academy of Forensic Sciences.

Advanced Format
Revised storage device design created to address technical limitations with the 512-byte sector size used on some storage devices by changing the sector size from 512-bytes to a multiple of 512-bytes such as 4096-bytes. In other words, storage devices with a sector size larger than 512-bytes.

Algorithm
A sequence of steps for solving a problem or accomplishing a task.

Anti-forensics
Active measures and techniques taken by a computer user to mislead or obstruct a future examiner. Common methods include deleting relevant files, creating bogus artifacts, modifying time stamps, altering log files, and creating file system artifacts that can disrupt the operation of common forensic tools.

APFS
Apple File System. One of the file systems supported on Macintosh Computers. APFS was introduced in 2017.

Artifact
A singular unit of interpretable data that can be extracted from a given data source that is useful for addressing questions in forensic investigations.

ASCII
American Standard Code for Information Interchange. A character encoding standard for electronic communication.

ATA
AT Attachment, also known as PATA (Parallel ATA) or IDE (Integrated Drive Electronics). ATA is a protocol for connecting storage devices to a host computer. Note: AT is an IBM PC model name, not an acronym.

Binary
A base-2 representation for numbers that uses a sequence of 1s and 0s to write a number. See *Place Value Notation*.

BIOS
Basic Input Output System. PC firmware to perform hardware initialization during the PC power-on startup process and provide other services to the operating system.

CFReDS
Computer Forensics Reference Data Sets. A repository at NIST of community created test data sets for testing digital forensic tools, including CFTT test data sets.

CFTT
Computer Forensic Tool Testing. A project at NIST for testing digital forensic tools.

Chip-Off
A destructive method of acquiring digital data from a device by removing memory chips from a printed circuit board and then directly copying the data from the chip.

CRC
Cyclic Redundancy Check. An error-detecting code commonly used to detect some accidental changes to transmitted data.

Data Acquisition

The general process of making a copy of digital data. This can be an entire digital device, just a partition from a storage device, or selected files from a file system. See also Disk Imaging.

DC3

Defense Cyber Crime Center.

DCO

Device Configuration Overlay. Used to change the features offered by a storage device, to present a subset of the available features, and to change the apparent storage capacity of a storage device to a smaller size.

DE

Digital Evidence. Also known as electronic evidence. Any probative information stored or transmitted in digital form that a party to a court case might use at trial.

DFRWS

Digital Forensics Research Workshop. A nonprofit organization dedicated to digital forensics. Also, the name of their annual conference.

DFRWS-EU

Digital Forensics Research Workshop Europe. Europe-based conference run by DFRWS.

DHS

U.S. Department of Homeland Security.

Disk Imaging

The process of acquiring the digital contents of a storage device (fixed disk, removable disk, flash drive, etc.) accessible by a user. This acquires all the data on a device including files, metadata, and contents of unallocated areas of the device. Device metadata and other data that are not user accessible are not usually acquired. See also Data Acquisition.

DOJ

U.S. Department of Justice.

EBCDIC

Extended Binary Coded Decimal Interchange Code. A character encoding used on older IBM mainframe computers.

ECC

Error Correcting Code. A method to ensure accurate detection and correction of transmission errors when data are moved from one place to another, e.g., memory to memory transfers, storage device to memory transfers.

Encode

In computing, to represent information as numbers. For example, text can be encoded by assigning each letter a unique number.

Encrypt

To encode information in a way that prevents unauthorized access. For example, decryption with a key is required to access the information.

ExFAT

Extensible File Allocation Table. A revised implementation of the FAT file system introduced that addresses some shortcomings in the FAT file system, e.g., allows files larger than 4GB and faster performance.

Exif

A metadata format employed in specific digital still camera file formats, e.g., JPEG. While EXIF is a specific type of metadata, the term is used colloquially in reference to a variety of metadata embedded in audio and image files describing the file content. Audio files may have metadata such as artist, copyright, creation date and more. An image file may have camera make, model, exposure settings, geolocation and more.

Ext4

Fourth Extended File System. The default file system for many Linux distributions as of 2022. Ext4 was introduced in 2008 as a replacement for the earlier ext2 and ext3 Linux file systems.

FAT

File Allocation Table. A file system developed for Microsoft computers introduced in 1977 and revised and extended over the years. Versions include FAT12 (12-bit addresses), FAT16 (16-bit addresses) and FAT32 (32-bit addresses).

File System

A method for organizing files on a storage device. Common file systems on Windows systems are NTFS, ExFAT and FAT. LINUX systems use ext4 and FAT. Apple Macs use HFS+, APFS, FAT and ExFAT.

Fixed media

A storage device that is physically installed inside a computer.

Hash

A mathematical technique that computes a short, fixed length value from a usually much longer set of data. Also, the value resulting from such a technique. Hashes can exhibit several useful properties depending on the intended application. In digital forensics, *cryptographic hashes* are usually used that have the following properties:

The same input always produces the same hash.

The original input data cannot be reconstructed from the hash.

Hash values of similar files have large differences.

The chance of two different files selected at random having the same hash value is so infinitesimal that it is essentially zero.

Hashes can be used to verify that a file, e.g., an acquisition from a device, has not changed, or to find copies of known contraband. Some cryptograph hashes in current use include Message Digest 5 (MD5), Secure Hash Algorithm (SHA-1, SHA-2 & SHA-3). SHA-2 and SHA-3 come in several variants.

Hexadecimal

A base 16 number system than uses 6 letters (A through F) in addition to the digits 0-9, in contrast with the traditional base 10 decimal system. See *Place Value Notation*.

HFS Plus

Hierarchical File System Extended. It is an apple file system introduced in 1998 and was replaced by APFS in 2017.

HPA

Host Protected Area. An area normally hidden from the user that can be configured on a storage device.

HTCIA

High Technology Crime Investigation Association.

IDEMA

International Disk Drive Equipment and Materials Association. A trade organization that represents the disk drive industry.

IEC

International Electrotechnical Commission. A nonprofit organization that develops and publishes standards concerning electrical technologies.

ISO

International Organization for Standardization. A nonprofit organization that develops and publishes international standards.

JTAG

Joint Test Action Group. An industry standard for verifying designs and testing printed circuit boards after manufacture.

LBA

Logical Block Address. A scheme to locate data on a storage device. An LBA of 0 is the first block of data on the storage device, LBA of 1 is the next block of data and so on.

MAC Times

Time stamp metadata maintained by a file system to track events in the life cycle of a file. The exact events recorded depends on the operating system and the file system. The usual meanings are Modify, Access and Create with slight differences in meaning for each type of file system and differences in meaning for files and directories.

MD5

Message Digest 5. A commonly used cryptographic hash algorithm.

Metadata

Metadata is a description of stored data. Categories of metadata include: (1) application metadata (in a document this could be author, organization, etc. For example, a database has metadata to describe the layout of the stored data), (2) file system metadata (placement of the file within the file system, owner, permissions, MAC times, etc.), (3) partition metadata that identifies the type of file system and global file system parameters, and (4) device metadata describes the layout of partitions on a device.

NTFS

New Technology File System. Microsoft Windows file system introduced in 1993, revised multiple times over the years.

NW3C

National White Collar Crime Center. A nonprofit nationwide support system for law enforcement and regulatory agencies involved in the prevention, investigation, and prosecution of economic and high-tech crime.

Operating System

The software that creates the digital environment for running software on a computer or other digital device. While most operating systems are variants of either MS Windows (95, 98, 2000, Vista, XP, 10, 11, etc.) or UNIX (BSD, Linux, Mac OS, iOS, etc.) other operating systems are sometimes encountered, e.g., Azure RTOS ThreadX and MOCOR. Various hardware devices often use custom built operating systems e.g., IoT devices, embedded systems, and feature phones.

OSAC

Organization of Scientific Area Committees for Forensic Science. The OSAC was created in 2014 to strengthen the nation's use of forensic science by facilitating the development and promoting the use of high-quality, technically sound standards.

Partition

A contiguous area of a storage device often used to contain a formatted file system but is sometimes used for special functions such as a swap area or a private storage area for an operating system.

Partition Table

A table describing the layout of a physical storage device that has been divided into partitions.

PhotoDNA

A hash technique that creates similar hash values for similar image files. The calculation of the hash is based on image content and not the binary representation of the image file. It also addresses reformatting of an image from one format to another, e.g., JPG to PNG, since the image content stays the same even though the binary representation changes significantly.

Place Value Notation

A method for representing numbers using a sequence of symbols selected from a fixed set of symbols that are assigned values based on the relative position within the sequence.

Removable media

A storage device that is either (1) a data container that is inserted and removed from a data reader or (2) a storage device that can be connected to or disconnected from a computer while the computer is running.

SATA

Serial ATA. A protocol for connecting storage devices to a host computer.

SCSI

Small Computer System Interface. A protocol for connecting storage devices to a host computer.

SHA

Secure Hash Algorithm. A family of cryptographic hash algorithms approved by NIST for security applications. Includes SHA-1, SHA-2 and SHA-3.

SIM card

Subscriber Identity Module. An older generation of a UICC.

Storage Device

An electronic or optical device that can store data for later retrieval. Older technologies such as magnetic tape (often used for system backup) are sometimes encountered. A storage device usually has some type of *file system* to organize the stored data as files. There are several types:

Fixed media physically installed in a computer. The computer must be powered off to install or remove the storage device.

Removable media. Can be installed or removed while the computer is running. Small storage devices are called *flash drives* or *thumb drives* (they are about the size of a human thumb). These devices are usually connected via a USB interface.

Memory card. One of several digital storage media types that can be inserted into a compatible card reader, e.g., secure digital (SD) card.

Optical disk. A compact disk (CD) or digital video disc (DVD) in one of several formats.

SWGDE

Scientific Working Group on Digital Evidence. A volunteer organization actively engaged in the field of digital and multimedia evidence to foster communication and cooperation as well as to ensure quality and consistency within the forensic community.

UICC

Universal Integrated Circuit Card. A new generation of a SIM card, which contains phone number and account information for mobile devices.

Unicode

A character encoding standard that can represent most languages in current use. There are several encoding formats that may be used. These Universal Text Interchange Formats (UTF) are available in several versions, e.g., UTF-8 (variable length), UTF-16 (16-bit) and others.

Volatile

Data stored on a device that is lost when power is removed from the device. For example, computer memory is lost when power to the computer is turned off.

Write Blocking

Techniques designed to prevent any modification to a storage device during acquisition or browsing. However, using a write blocker does not mean that no changes occur to a storage device. Sometimes changes are triggered by the storage device itself. These changes are usually transparent to the user of the device and have no impact on user created active data.