UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Visión Electrónica

https://revistas.udistrital.edu.co/index.php/visele

VISIÓN ELECTRÓNICA

A CASE-STUDY VISION

Computer Forensics Software Tools

Herramientas Informáticas en Computación Forense

Lilia Edith Aparicio Pico [1], *Jonathan Morrison Tarquino* [2]

INFORMACIÓN DEL ARTÍCULO

Historia del artículo:
Enviado: 13/06/2024
Recibido: 01/07/2024
Aceptado: 04/10/2024

Keywords:
Forensic Tools
Computer Forensics
Forensic Analysis
Digital Incident Investigation
Forensic Methodologies

Palabras clave:
Análisis Forense
Computación Forense
Herramientas Forenses
Informática Forense
Investigación de Incidentes Digitales
Metodologías Forenses

ABSTRACT

This research explores the process associated with analysis in forensic computing, structuring it into four widely recognized phases: collection, examination, analysis and reporting. In addition, it relates this process to specific procedures used in the study of digital incidents, with the purpose of identifying specific needs linked to the main computer tools used in forensic computing.

In the development of the study, two tools are identified for each forensic procedure, selected according to their characteristics, uses and functionalities. These tools are described and subsequently compared to evaluate their main advantages and disadvantages. The results of this comparison are presented in a table that allows a quick and accessible analysis of the items evaluated.

Additionally, the work emphasizes the use of multifunctional tools that allow the execution of more than one forensic procedure. These tools, although not directly compared in the study, stand out for their growing adoption due to their versatility and integration with the diverse needs of forensic computing researchers.

RESUMEN

La presente investigación explora el proceso asociado al análisis en informática forense, estructurándolo en cuatro fases ampliamente reconocidas: recolección, examinación, análisis y reporte. Además, relaciona este proceso con procedimientos específicos empleados en el estudio de incidentes digitales, con el propósito de identificar necesidades concretas vinculadas a las principales herramientas informáticas utilizadas en la computación forense.

En el desarrollo del estudio, se identifican dos herramientas para cada procedimiento forense, seleccionadas según sus características, usos y funcionalidades. Estas herramientas son descritas y posteriormente comparadas

[1] Licenciada en Ciencias de la Educación - Física, Universidad Distrital Francisco José de Caldas, Colombia. Magister en Teleinformática, Universidad Distrital Francisco José de Caldas, Colombia y Doctor en Ciencias Técnicas, Universidad Central "Marta Abreu" de las Villas, Cuba. Current position: Director del grupo Gitem++, Profesor Titular / Universidad Distrital Francisco José de Caldas, Colombia. E-mail: medicina@udistrital.edu.co

[2] Matemático, Fundación Universitaria Konrat Lorenz, Colombia. Magister en Modelado y Simulación, Universidad Jorge Tadeo Lozano, Colombia. Current position: Miembro del grupo Gitem++, Profesor, Universidad Distrital Francisco José de Caldas, Colombia. E-mail: jmtarquinoa@udistrital.edu.co

Citar este artículo como: L. E. Aparicio, W. Barragan "Integration and Interoperability of Electronic Health Records", Visión electrónica, vol. 18, no. 2, pp. 309-322, july-december 2024, https://doi.org/10.14483/22484728.23380

con el fin de evaluar sus principales ventajas y desventajas. Los resultados de esta comparación se presentan en un cuadro que permite un análisis rápido y accesible de los ítems evaluados.

Adicionalmente, el trabajo enfatiza el uso de herramientas multifuncionales que permiten la ejecución de más de un procedimiento forense. Estas herramientas, aunque no son objeto directo de comparación en el estudio, destacan por su creciente adopción debido a su versatilidad e integración con las diversas necesidades de los investigadores en informática forense.

1. Introduction

Forensic informatics has appeared in response to the need for well-defined methodologies and concrete practices for investigating digital incidents. In a precise definition, Broun C. [1] states that it is "the science and art of applying computer science to aid the legal process, allowing the reconstruction of what has happened in a computer system after a digital incident". In this context, it is essential to find that this digital process consists of three main factors.

The first factor is the forensic investigator, who must apply a set of knowledge and skills to solve the puzzles presented, which often deviate from any predefined structure in technical and scientific methodologies. The second factor is the analyzed technological infrastructure, which can include a variety of equipment (servers, laptops, desktops, routers, firewalls, IDS, and IPS, among others). The third factor is a set of computer tools, whether in software or hardware form, which may vary depending on the analyzed incident, the investigator's preferences, or simply their awareness of the existence of these tools.

These three elements form the foundation for presenting forensic analysis, defined by Thomas [2] as "a set of techniques, protocols, and knowledge aimed at identifying, analyzing, preserving, and providing digital evidence in a manner that is valid within a legal framework."

To ensure the success of this approach in practice, it is necessary to have a set of tools that can be used in each of the procedures outlined by the forensic investigator, aiding in the execution of each phase. Some of these procedures, proposed by Kent, Chevalier, Grance, and Dang [3], consist of Collection, Examination, Analysis, and Presentation. It is important to understand that each of these phases is further composed of steps or actions outlined in the forensic process method and must be executed with best practices to avoid analytical errors that could lead to legal pitfalls. As Allende [4] aptly points out, "It is crucial to consider the legal framework; otherwise, digital evidence may not be admissible, or even legal violations may occur, such as the breach of privacy regarding specific data and communications."

Considering the, this study delves into the thematic exploration of the uses and primary functionalities of computer tools employed in forensic analysis.

2. Forensic Process

The forensic process encompasses the interaction of the forensic analyst, their tools, and the computer environment under investigation, governed by a series of phases (see Figure 1), which include:

2.1. Identification of Possible Data Sources

These sources can be of any type and have various locations. In the best-case scenario, evidence is found locally, such as on hard drives, CDs, DVDs, USB flash drives, or devices like routers, IDS, IPS, and Firewalls within the same technological infrastructure. However, sometimes, evidence is in environments to which we have no direct access. This is a common scenario with ISPs, where a court order is needed to obtain the desired information.

Data Acquisition: This process involves a set of actions, including designing a data acquisition plan, collecting the data, and verifying its integrity. In the planning phase, data is assessed to estimate its potential value in the investigation. Next, the volatility or permanence of the data in the medium is decided, influenced by the type of digital incident being analyzed. Finally, the effort needed to obtain the data is evaluated. Data extraction encompasses both volatile and non-volatile data, which may originate from hard drives, servers, databases, system recordings, or electronic devices. Volatile data extraction involves collecting information stored in locations like the computer's RAM or slack space. Its key characteristic is its potential to disappear or change when abnormal procedures are executed on the computer, such as shutting it down or putting it into standby or hibernation mode. Non-volatile data extraction aims to collect all types of information that can persist in the analyzed medium despite various actions taken on the computer. Data integrity is verified, a term defined by Vanstone [5] as the "property that confirms that digital data has not been altered in an unauthorized manner since it was created, transmitted, or stored by an authorized source."

2.2. Examination

Examination is the second step in forensic analysis, with its primary goals being to clarify what is being sought, find types of files and extensions that may be useful in the investigation, and explore more characteristics of the operating system and its components. This includes investigating hidden data, encrypted data, password-protected files, or those with some form of compression. This process enables data filtering and preparation for later analysis.

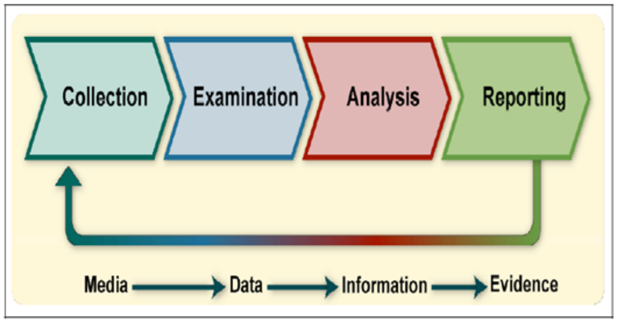
2.3. Data Analysis

The data analysis stage, the third step, serves to link the collected information with the ongoing investigation. This allows for the identification of the key characteristics of the digital incident and generates input for the analyst to supply concrete conclusions about the actual events that happened.

2.4. Reporting or Presentation

The reporting or presentation process involves preparing, documenting, and presenting the findings of the forensic analysis. It characterizes the entire executed process and presents the necessary conclusions to ensure the investigation yields best results and gains approval in the relevant professional or legal contexts.

Figure 1. Forensic Process. [3]



3. Needs of the forensic process

As shown in the earlier index, the forensic process consists of several stages, all of which must be meticulously executed with the best practices and methods to prevent any incidents in the conducted investigation. It is essential to be aware that each forensic analysis is unique, and no digital incident is ever entirely the same, even if they appear similar. With this understanding in mind, the most common and fundamental procedures of digital forensics are found, which are encompassed within the process proposed by Kent, Chevalier, Grance, and Dang [3], and are outlined below:

Table 1. Stages of the Forensic Process Based on Procedure.

Phases of the Forensic Process	Procedure
Collection	Volatile data collection
	Duplication of non-volatile data sources
	Non-volatile data integrity verification
Examination	Data visualization
	Data recovery
	Opening applications and/or files
Analysis	Metadata extraction
Reporting	Documentation and recording of performed procedures.
	Documentation and recording of found data.

Source: own.

So, in each of the mentioned procedures, there are characteristic factors that generate requirements for the forensic analyst, and so, for the tools used by them. In this regard, each of the processes is described below, along with some of their characteristics:

3.1. Collection of Volatile Data

This process prioritizes the extraction of this type of data because, as Broun C. suggests, volatile data are "those that can be in an active state or change and are found in physical memory devices like RAM and would disappear with the loss of power to the device." However, collection follows an order outlined in RFC 3227 [6], which is:

- Registry, cache
- Routing tables, ARP cache, process tables, kernel statistics, and memory
- System temporary files
- Disk
- Remote logs and relevant system data in analysis
- Physical configuration and network topology
- Archiving media

3.2. Duplication of Non-Volatile Data Sources

This process should enable the generation of exact copies of data sources and often becomes the creation of disk images, which are defined by Lyle [7] as "the duplication of an entire hard drive or partition" by "copying disk sectors from a source to a destination that is identical or nearly identical to the original."

3.3. Data Integrity Verification

To ensure that the duplication process did not alter any collected data, integrity verification is performed, which involves comparing the message digest between the source and destination media. This message digest is a unique hash value that changes with even a single bit change between the source and destination and is calculated using algorithms like SHA-1 and MD5.

3.4. Data Visualization

After collecting data from various sources, it is necessary to visualize it in a way that finds its key characteristics such as file types, formats, extensions, and sizes, while also ensuring that the accessed data will not be changed or altered. Furthermore, it is pertinent to have features for visualizing hidden files or system-protected files because, as Allen [8] points out, "digital data is more persistent in storage than the average user expects."

3.5. Data Recovery

When collecting data, it must be ensured that the ability to recover files that have been cut from the original system but may still be stored in accessible disk sectors is available for reconstruction.

3.6. Opening Applications and/or Files

This process involves the identification of applications and/or files protected by passwords and applying various techniques or procedures to access their content.

3.7. Extraction of Metadata

This process allows for obtaining additional data from the files being analyzed, which are defined as follows [9] in document headers: "title," "author," "affiliation," "address," "note," "email," "data," "abstract," "telephone," "keyword," "web," "degree," and "pubnum." "Note" refers to phrases about recognition, copyright, notices, and citations; "degree" refers to the language associated with the thesis; "pubnum" means the publication number. Bibliographic fields include "author," "book title," "date," "publisher," "institution," "journal," "location," "note," "pages," "editorial," "technology," "title," and "volume."

3.8. Documentation and Recording of Procedures Performed

Each step of the forensic analysis must be documented to prove the implementation of best practices. This will serve to present the evidence in a legal setting, ensuring that the collected data is not rejected due to mishandling.

3.9. Documentation and Recording of Found Data

The data found in the conducted process must be recorded, with their key characteristics such as file sizes, volumes, partitions, file types, and applications clearly documented to clarify the content of the collected information from the outset.

4. Identification of tools to be used based on the forensic process

For the execution of each of the procedures (Table 1) within the forensic process (Fig. 1), there are a series of tools used, each having characteristics such as ease of use, correct data copying and management, licensing, and compatibility with operating systems and analyzed platforms. These characteristics were carefully considered and presented as a means of selecting and proposing two tools per forensic procedure, which are used in some of the procedures or, in some cases, can also serve as utilities that fulfill various functions. Therefore, they contribute to one or more of the procedures performed in forensic analysis.

Table 2. Phases and Procedures Based on Forensic Tools.

Phases of the Forensic Process	Procedure	Forensic Tool
Collection	Volatile Data Collection	Volatile Framework FatKit
	Duplication of Non-volatile Data Sources	FIRST Diskimager AIR (Automated Image and Restore) (Automated Image and Restore) Automated Image and Restore
	Data Integrity Verification	HashTab GtKHash
Examination	Data Visualization	File
	Data Recovery	Diskintemals NTFS, FAT y RAID recovery Data Recovery de WinHex
	Opening Applications and/or Files	Cain y Abel John the Ripper
Analysis	Metadata Extraction	Metadata Extraction Metadata
Report	Documentation and Recording of Found Data	Encase Forensic Ediscovery
	Multiple procedures	Sysintemals Suite Encase Caine Forensic Toolkit FTK

Source: own.

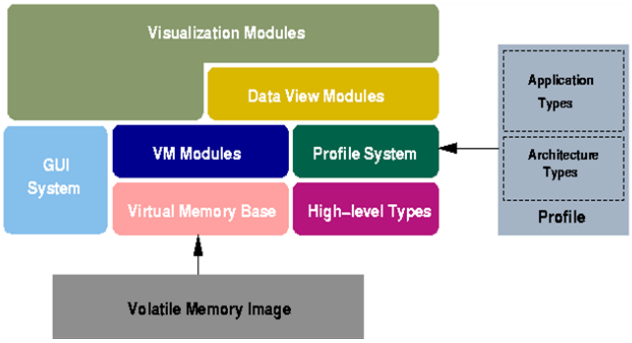
After finding the two tools per procedure, a brief description of the functionality and key features of each is supplied as follows:

4.1. Tools Used for the Collection of Volatile Data

- Volatility Framework

It is a collection of open-source tools, implemented under GNU, that enables the extraction of volatile data from RAM memory. It can perform tasks such as copying system date and time stamps, running processes, open network sockets, established network connections, DLLs loaded by each process, files opened by each process, registry entries per process, memory-mapped processes, Kernel modules of the operating system, mapping of physical offsets to virtual addresses (process chains), virtual addressing descriptions, process thread scans, sockets and connections, extracting samples of executables in memory, and automated format conversion. Additionally, it is a tool compatible with Linux, Cygwin, Windows, and OSX 10.5 platforms.

Figure 2. FatKit Software Architecture. [11]



- FatKit (Forensic Analysis Toolkit)

Automates the extraction and visualization of digital objects in physical memory [11]. It also allows for the reconstruction of virtual address space, translation of virtual addresses to physical ones, and supports profiles for automating low-level object formats when source code is available. It can count tasks, processes, and find malicious code living in memory. It is compatible with X-86 based virtual address spaces and Windows and Linux kernels. It has two modules: the Object Browser, which can interpret binary objects in memory with a code-level abstraction and high-level languages, and the Address Space Viewer, which allows analysts to visualize data appearing in a physical or virtual memory space. Other tools used for this volatile data collection procedure include WMFT (Windows Memory Forensic Tool), Procenum, Idetect, MemDump, KnTlist from KnTTools, and Vad Tools (Virtual Address Descriptor).

4.2. Tools for duplicating non-volatile data sources

- FIRST Diskimager

Described as a project designed specifically to help forensic investigators implement a clean data acquisition procedure without changing the data. Its main aims include supplying a user-friendly Graphical User Interface (GUI) solution to ensure proper handling of source and destination data, unlike commonly used command-line-based imaging tools.

- AIR (Automated Image and Restore)

An application with a GUI that allows for bit-by-bit imaging from the source device to dd/dcfldd images. It features auto-detection of drives with IDE, SCSI, CD-ROM, and tape interfaces, creating dd or dcfldd images, data verification by comparing source and destination using MD5 or SHA 1/256/384/512 hashing, image compression and decompression in gzip or bzip2 formats, image creation using TCP/IP networks via netcat or crypcat, image splitting into multiple segments, and logging of activity times and hours.

Other software-based tools used for data cloning, duplication, or imaging, sometimes referred to as "imaging" tools, include guymager, aimage, dd, iLook IXimager, iLook, dcfldd.

4.3. Tools for duplicating non-volatile data sources.

- HashTab

Allows for calculating the hash values of a specific file. It supports MD5, SHA1, SHA2, RipeMD, HAVAL, and Whirlpool algorithms. It is supported by the Windows console, ensuring easy verification of data integrity and authenticity when comparing data with the original. In Windows, it is as simple as right clicking the file and selecting the "observe new hash" option. Additionally, users can customize the type of hash they want to calculate.

- GtkHash

A utility that calculates the message digest of files using the mhash library, which supports hash functions like MD5, SHA1, SHA256, SHA512, RIPEMD, HAVAL, TIGER, and WHIRLPOOL. This application is covered by the GNU GPL from the Free Software Foundation.

Other integrity-checking tools include MD5sum, SHA1Sum, Md5 for Windows, and sha1 for Windows.

4.4. Tools for data visualization

- File

A command-line-based tool that finds files based on their associated data types and formats. The application uses three test methods: file system, magic number, and language, executed in that order. The file system examination checks the file type and filename, looking for execution flags and common file names. The magic number review examines the first few bytes of the file to find its type, while the language review checks the text to decide the encoding used.

- Handyfiletool

A free-to-use tool that allows users to search and manage files and folders, create lists of favorite files, copy, move, create, or cut files and folders, rename files in multiple directories simultaneously, replace text blocks in lists of files and folders simultaneously, and rename files and folders in the same directory. It also supports file searches by extension (*.doc; *xls; *html), creation or usage periods, creation date intervals, file sizes, and attributes.

4.5. Tools for data recovery

- DiskInternals NTFS, FAT, and RAID Recovery

Licensed tools that allow for file recovery from physically damaged hard drives, damage caused by viruses or malware on a specific disk volume, failures in disk access by the operating system, partition or disk formatting, inaccessible files or folders, corrupted or damaged partitions, or simply cut files from the system. These tools are compatible with NTFS, FAT, FAT32, JBOD RAID arrays, 0, 1, 1E, 4, 5, 0+1, and 1+0, and function on Windows operating systems.

- Data Recovery from the Winhex Suite

This functionality within the WinHex suite allows for the recovery of damaged, lost, or deleted files from drives with FAT12, FAT16, FAT32, and NTFS formats. It supports file recovery by file name and handles file types with extensions such as jpg, png, gif, tif, bmp, dwg, psd, rtf, xml, html, eml, dbx, xls/doc, mdb, wpd, eps/ps, pdf, qdf, pwl, zip, rar, wav, avi, ram, rm, mpg, mov, asf, mid.

Other tools available for data recovery include Easy recovery, Active Uneraser, Active Partition Recovery, Winternals Disk Commander, Active Undelete, GetDataBack for FAT, GetDataBack for NTFS, and Partition Find and Mount.

4.6. Tools for opening applications and/or files

- Cain and Abel

A password recovery tool for Windows operating systems, which allows for password identification through techniques such as network sniffing, encrypted password cracking using brute force, dictionary attacks, or cryptanalysis. It can also decode passwords stored in known locations within the operating system and reveal password boxes. This software does not exploit vulnerabilities but rather uses weaknesses in security protocols, authentication methods, and cache storage mechanisms.

- John the Ripper

A cross-platform tool for password cracking, primarily relying on exploiting weak passwords through brute-force and dictionary attacks. It allows users to define the range of characters used to construct words and lengths, and it can pause and resume the cracking process at the user's discretion. It supports the most used hash types in Unix flavors, Kerberos AFS, and Windows NT/2000/2003/XP.

Other tools used for password identification include THC Hydra, Aircrack, L0phtcrack, Airsnort, SolarWinds, Pwdump, RainbowCrack, and Brutus.

4.7. Tools for metadata extraction

- Metadata Extraction

A tool that allows for extracting metadata preserved in various document formats, including PDF files, image files (BMP, GIF, JPEG, and TIFF), sound files (WAV and MP3), and general office suite documents from Windows (DOC, XLS, PPT). In addition to extracting this information stored in digital files, it can export the collected data in an XML format. The extracted information is based on basic data such as file size, file name, creation date, and the originating computer. This tool works on Windows with a graphical interface and on Unix with a command-line interface. Furthermore, it allows the opening of files in read-only mode, preserving their integrity.

- Metadata Assistant

It is an application compatible with Microsoft Word, Excel, PowerPoint 2000-2007, PDF, and incorporable with Outlook 2000-2007 and Lotus Notes. It allows for the extraction of basic information from documents that is normally hidden or not easily visible. All extracted information is exported to XML or RTF files for use and analysis in different environments.

4.8. Tools for documentation and data recording

- Encase Forensic

It is a specialized suite for forensic analysis, and although it has more functionalities, its reporting module allows for creating lists of all analyzed files and folders, identifying all URLs with dates and visit times found on the analyzed system, storing logs of found information, generating detailed information about the hard disk, its physical and logical partitions, and viewing data acquisition details such as drive geometry, folder and file structures, exporting any report in RTF or HTML formats. It also allows for highlighting analyzed data and creating attached notes.

- Ediscovery

It is an application that guides the investigator through each step of the search and data collection process from shared folders, mail servers, databases, and common repositories. It has a user-friendly graphical interface that allows monitoring and reporting on data processing in real-time for audit purposes like chain of custody verification. It features an easy-to-understand graphical interface for checking the status of exploration and management of analyzed information. For data storage, it has deduplication properties, which save on storage costs.

4.9. Tools used for multiple forensic procedures

- Sysinternals Suite

It is a collection of tools for Windows operating systems, most of which allow for the identification and collection of volatile data. Among the main utilities in this suite are AccessChk (shows account privileges), AccessEnum (identifies who accessed which directories, files, and registry keys), AdExplorer (edits and views Active Directory), AdInsight (real-time LDAP monitor), Autologon (bypasses logon screen during startup), Autoruns (displays programs configured to automatically start), BgInfo (creates desktop backgrounds showing basic system information), BlueScreen (simulates machine restarts), CacheSet (manages system cache), ClockRes (displays system resolution), Contig (file storage optimizer used when defragmentation is not desired), Coreinfo (command-line tool showing mapping between logical and physical processors), DebugView (intercepts calls to DbgPrint by devices, drivers, and Win32 programs), Desktops (allows use of 4 virtual desktops), Disk2vhd (converts physical systems to virtual), DiskExt (displays volumes recognized by the system), Diskmon (captures all disk activity), DiskView (graphs disk sectors), Disk Usage (DU) (displays disk usage by directory), EFSDump (displays encrypted file information), Handle (shows which files are open by which system processes), Junction (creates symbolic links to system locations in NTFS), ListDLLs (lists all loaded DLLs), LoadOrder (displays hardware load order), LogonSessions (shows established sessions in the system), PendMoves (lists files to be deleted or renamed on the next system reboot), PortMon (monitors serial and parallel port activity), Process Explorer (identifies registry keys and files associated with a process), PsFile (displays files opened remotely), PsPasswd (changes account passwords), RootkitRevealer (scans for rootkits), ShareEnum (searches for shared files on the network), Sync (flushes cache to disk). In addition to these, Sysinternals has several other highly functional applications for diverse types of analysis.

- Encase

It is a suite specifically designed for forensic processes, offering various functionalities, including acquisition, forensic process automation, visualization, search, and reporting. In terms of acquisition, it can find the number of defective or error-prone disk sectors, define block size on a disk, stop and resume data acquisition, perform data collection in total or in part, verify image integrity using CRC and MD5, use LinEn for disk image acquisition through booting and WinEn for buying volatile data stored in RAM. In terms of forensic process automation, it allows for the creation of algorithms using logical propositions like AND and OR, automation of Active Directory information, partition recovery, and recovery of deleted or damaged files. For analysis, it includes a system log analyzer, unallocated disk space search, file signature analysis. In terms of visualization and search, it provides registry visualization, timeline analysis with date and time, binary file search, web and email usage file search, and search for files in slack space and unallocated space. For reporting, it can display all files reviewed in a specific analysis, review and save logs, document the incident response process, supply detailed information about physical and logical partitions, exporting all the aforementioned information to RTF or HTML files. Lastly, it includes detailed analysis of web history and cache, HTML reconstruction, and review of specific files associated with email databases like Outlook, Microsoft Exchange EDB parser, Lotus Notes, Yahoo, Hotmail, and Netscape. This application supports software RAID and is compatible with Windows 2000/XP/2003 Server.

- Caine

It is a collection of applications compiled on a live CD, allowing various forensic procedures. Some run-in console mode, while others use a graphical interface. The included utilities are AIR (creates and splits images), Guymager (generates images in standard formats and Encase files), DC3DD (evolution of the

DD and DCFLDD image creation tools), Autopsy Forensic Browser (a graphical interface for SleuthKit for analyzing Unix and Windows operating systems with NTFS, FAT, UFS1/2, Ext2/3 file systems), Foremost (data recovery based on headers, footers, and internal data structures), Scalpel (recovers data by scanning image files created with dd, safeback, and Encase), SFDumper (recovers files by extension type and includes additional features like duplicate file deletion based on hash comparison), StegDetect (a tool for steganography used to discover hidden information in image files), OphCrack (password cracker using brute force), Fundl (used to recover all deleted files from a disk or image). In addition to the mentioned tools, this live CD includes LRRP, photorec, stegdetect, smartmontools, testdisk, afflib, cryptcat, libewf, md5sum, sha256sum, sha512sum, MD5deep, Tigerdeep, Whirpooldeep, reglookup, ddrescue, Xhfs, HFSutils, reglookup, Pasco, Rifiuti, Rifiuti2, Galleta, Fatback, Wipe, Shred, Tableau-Parm, readpst, AtomicParsley, Exif, bkhive, lnk_parse, mork.pl, dos2unix, Steghide, chntpw, tkdiff, xdeview, and lnk.sh.

- Forensic Toolkit FTK

It is a specialized solution that integrates various functionalities, allowing procedures such as image creation, registry analysis, file description, password cracking, identification of files with hidden data using steganography techniques, recovery of passwords in over 100 known applications, and dictionary-based password cracking. Additionally, it enables case generation, associating all processes with a specific case, as well as creating backups and archiving bought information. Advanced data exploration allows for analysis of data in slack space, unallocated space, and volatile data in general, including RAM dumps, listing all running processes, loaded DLLs in memory, set up sockets, and system analysis for the identification of threats like Rootkits. Its advanced search function allows for finding files and contents using keywords, extensions, and file formats. It features a graphical interface that eases easy tool manipulation, with the capability to export files to CSV formats and integrate with databases. Reports are generated in HTML, PDF, XML, and RTF formats for light reading and analysis.

Other tools that are included in live CD versions are: SafeBoot Disk, Helix 3, SMART Linux, DEFT Linux, SPADA, BackTrack, and Plain Sight.

5. Comparison of computer tools used for digital forensics.

Each of the studied tools is summarized below in Table 3, which allows for the identification of their key features, licensing, usage mode, and compatibility in terms of operating systems. In this way, commonalities and differences between each of the tools can be found.

Table 3. Forensic tools are based on key features, usage modes, and compatibility.

Tool	Procedure	Features	License	Usage mode	Compatibility
Volatile Framework	Volatile data collection	Image copy of timestamp, network connections and sockets, registry entries, and executable DLLs in memory.	GPL	Console	Linux, Cygwin, Windows y OSX 10.5.

Tool	Procedure	Features	License	Usage mode	Compatibility
FatKit	Volatile data collection	Automation of extraction and visualization of digital objects in physical memory, virtual address space reconstruction, virtual-to-physical address translation	Open source	Graphical	Windows y Linux
Volatile Framework	Volatile data collection	Image copy of timestamp, network connections and sockets, registry entries, and executable DLLs in memory.	GPL	Console	Linux, Cygwin, Windows y OSX 10.5.
FatKit	Volatile data collection	Automation of extraction and visualization of digital objects in physical memory, virtual address space reconstruction, virtual-to-physical address translation	Open source	Graphical	Windows y Linux
FIRST Diskimager	Duplication of non-volatile data sources	Duplication of non-volatile data sources	GPL	Graphical	Not applicable
AIR Automated Image and Restore	Duplication of non-volatile data sources	Creation of bit-by-bit images from a source device to dd/defldd images	GPL	Graphical	Windows MAC (Multiplexed Analog Components) OS
HashTab	Data integrity verification	Calculation of hash values for a specific file	Open source	Graphical	Windows MAC OS
GtkHash	Data integrity verification	Allows for file message digest calculation.	GPL	Graphical	Linux
File	Data visualization	Enables the identification of files based on data types and formats.	GPL	Console	Linux
Handyfile tool	Data visualization	Allows for searching and managing files and folders, creating list of favorite files, copying, moving, creating, or cutting files and folders.	Open source	Graphical	Windows
Diskinternals NTFS, FAT y RAID	Data recovery	Allows for Dile from states such as physically damaged hard drives, damage caused by viruses or malware on a specific disk volume.	SI	Windows MAC OS	Windows MAC OS
Data Recovery de WinHex	Data Recovery	Allows for the recovery of damaged, lost, or deleted files from drives with FAT12, FAT16, FAT32, and NTFS formats.	Open source	Graphical	Windows
Cain y Abel	Application and/or File Opening	Password recovery for Windows operating systems	Open source	Graphical	Windows
Metadata Extraction	Metadata Extraction	Allows for the extraction of preserved metadata in documents of various formats.	GPL	Graphical / Console	Windows, Linux

Source: own.

6. Conclusions

The use of tools in the field of digital forensics enables the optimization of various procedures associated with forensic analysis. However, it demands constant updating of knowledge about the management and usage of these tools; otherwise, there is a risk of misusing resources, which can hinder the entire forensic process.

It is pertinent to design methodologies for evaluating the functionality of different computer tools used in the forensic process, aligning them with the regulations and specific technological, social, and legal aspects of the Colombian environment.

Open-source tools prove to be a practical solution, particularly when it comes to developing country-specific technological solutions for digital forensics. They allow for the modification of source code to meet specific needs, adoption and translation of tools into native languages, and the creation of results presentations that are understandable within the Colombian context.

While tools with graphical interfaces work perfectly for analysis and reporting procedures, in most cases, the best tools for data collection and examination run through command-line interfaces. This demands forensic analysts to have a wide range of skills to perform each procedure optimally and without issues.

Due to the substantial number of forensic tools available in the technological landscape, current standards and testing methodologies have become insufficient for evaluating, comparing, and recommending computer tools. This creates technological gaps that can affect the implementation of best practices in forensic analyses.

References

[1] By C. L. T. Brown "Computer Evidence: Collection & Preservation", in Journal of Digital Forensic Practice, vol. 1, pp. 71–72. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/15567280500541397

[2] W. A. Bhat, A. Alzahrani, and M. A. Wani, "Can computer forensic tools be trusted in digital investigations?" Science and Justice, vol. 61, no. 2, pp. 198–203, Mar. 2021. https://doi.org/10.1016/j.scijus.2020.10.002

[3] B. K. Akcam, "Forensic Science International we should give special mention to the observance of secrecy in the automotive industry in case of security relevant systems Digitizing Forensic Laboratories: The Turkish Criminal Police Laboratories Case."

[4] L. Xu, B. Wang, L. Wang, D. Zhao, X. Han, and S. Yang, "PLC-SEIFF: A programmable logic controller security incident forensics framework based on automatic construction of security constraints," *Computers and Security*, vol. 92, May 2020. https://doi.org/10.1016/j.cose.2020.101749

[5] M. I. Cohen, D. Bilby, and G. Caronni, "Distributed forensics and incident response in the enterprise," in *Digital Investigation*, 2011, vol. 8. https://doi.org/10.1016/j.diin.2011.05.012

[6] C. J. Courtney Mustaphi *et al.*, "Guidelines for reporting and archiving 210Pb sediment chronologies to improve fidelity and extend data lifecycle," *Quaternary Geochronology*, vol. 52, pp. 77–87, Jun. 2019. https://doi.org/10.1016/j.quageo.2019.04.003

[7] P. Lutta, M. Sedky, M. Hassan, U. Jayawickrama, and B. Bakhtiari Bastaki, "The complexity of internet of things forensics: A state-of-the-art review," *Forensic Science International: Digital Investigation*, vol. 38. Elsevier Ltd, Sep. 01, 2021. https://doi.org/10.1016/j.fsidi.2021.301210

[8] W. Halboob, R. Mahmod, N. I. Udzir, and M. D. T. Abdullah, "Privacy levels for computer forensics: Toward a more efficient privacy-preserving investigation," in Procedia Computer Science, 2015, vol. 56, no. 1, pp. 370–375. https://doi.org/10.1016/j.procs.2015.07.222

[9] G. Ma, Z. Wang, L. Zou, and Q. Zhang, "Computer forensics model based on evidence ring and evidence chain," in *Procedia Engineering*, 2011, vol. 15, pp. 3663–3667. https://doi.org/10.1016/j.proeng.2011.08.686

[10] M. Saadoon, S. H. Siti, H. Sofian, H. H. M. Altarturi, Z. H. Azizul, and N. Nasuha, "Fault tolerance in big data storage and processing systems: A review on challenges and solutions," *Ain Shams Engineering Journal*, vol. 13, no. 2. Ain Shams University, Mar. 01, 2022. https://doi.org/10.1016/j.asej.2021.06.024

[11] D. Closser and E. Bou-Harb, "A live digital forensics approach for quantum mechanical computers," *Forensic Science International: Digital Investigation*, vol. 40, p. 301341, Apr. 2022. https://doi.org/10.1016/j.fsidi.2022.301341

[12] G. Koorey, S. McMillan, and A. Nicholson, "Incident Management and Network Performance," in *Transportation Research Procedia*, 2015, vol. 6, pp. 3–16. https://doi.org/10.1016/j.trpro.2015.03.002

[13] K. Barik, S. Das, K. Konar, B. Chakrabarti Banik, and A. Banerjee, "Exploring user requirements of network forensic tools," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 350–354, Nov. 2021. https://doi.org/10.1016/j.gltp.2021.08.043

[14] A. M. Marshall, "Digital forensic tool verification: An evaluation of options for establishing trustworthiness," *Forensic Science International: Digital Investigation*, vol. 38, Sep. 2021. https://doi.org/10.1016/j.fsidi.2021.301181

[15] T. Wu, F. Breitinger, and S. O'Shaughnessy, "Digital forensic tools: Recent advances and enhancing the status quo," *Forensic Science International: Digital Investigation*, vol. 34, Sep. 2020. https://doi.org/10.1016/j.fsidi.2020.300999

[16] W. A. Bhat, A. AlZahrani, and M. A. Wani, "Can computer forensic tools be trusted in digital investigations?" *Science and Justice*, vol. 61, no. 2, pp. 198–203, Mar. 2021. https://doi.org/10.1016/j.scijus.2020.10.002

[17] A. Daniel D and S. E. Roslin, "Data validation and integrity verification for trust-based data aggregation protocol in WSN," *Microprocessors and Microsystems*, vol. 80. Elsevier B.V., Feb. 01, 2021. https://doi.org/10.1016/j.micpro.2020.103354

[18] J. Tian and X. Jing, "Cloud data integrity verification scheme for associated tags," *Computers and Security*, vol. 95, Aug. 2020. https://doi.org/10.1016/j.cose.2020.101847

[19] C. Yang, F. Zhao, X. Tao, and Y. Wang, "Publicly verifiable outsourced data migration scheme supporting efficient integrity checking," *Journal of Network and Computer Applications*, vol. 192, Oct. 2021. https://doi.org/10.1016/j.jnca.2021.103184

[20] Q. Zhao, S. Chen, Z. Liu, T. Baker, and Y. Zhang, "Blockchain-based privacy-preserving remote

data integrity checking scheme for IoT information systems," *Information Processing and Management*, vol. 57, no. 6, Nov. 2020. https://doi.org/10.1016/j.ipm.2020.102355

[21] K. Porter, R. Nordvik, F. Toolan, and S. Axelsson, "Timestamp prefix carving for filesystem metadata extraction," *Forensic Science International: Digital Investigation*, vol. 38, Sep. 2021. https://doi.org/10.1016/j.fsidi.2021.301266

[22] R. Nordvik, K. Porter, F. Toolan, S. Axelsson, and K. Franke, "Generic Metadata Time Carving," *Forensic Science International: Digital Investigation*, vol. 33, Jul. 2020. https://doi.org/10.1016/j.fsidi.2020.301005

[23] M. Kiweler, M. Looso, and J. Graumann, "MARMoSET – Extracting Publication-ready Mass Spectrometry Metadata from RAW Files," *Molecular and Cellular Proteomics*, vol. 18, no. 8, pp. 1700–1702, 2019. https://doi.org/10.1074/mcp.TIR119.001505

[24] N. K. Booker, P. Knights, J. D. Gates, and R. E. Clegg, "Applying principal component analysis (PCA) to the selection of forensic analysis methodologies," *Engineering Failure Analysis*, vol. 132, Feb. 2022. https://doi.org/10.1016/j.engfailanal.2021.105937

[25] J. W. Ma, T. Czerniawski, and F. Leite, "An application of metadata-based image retrieval system for facility management," *Advanced Engineering Informatics*, vol. 50, Oct. 2021. https://doi.org/10.1016/j.aei.2021.101417