REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider – BISKRA
Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie
Département d'informatique

N° d'ordre : RTIC8/M2/2019

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : RTIC

Conception et réalisation d'un système d'aide aux investigations numériques

-application des techniques de computer forensics-

Par :

KHEIREDDINE REMADNA

Soutenu le 06 juillet 2019, devant le jury composé de :

Dr. TOUIL Keltoum	M A A	Président
Dr. Hamza Saouli	M A A	Rapporteur
Dr. Nadia BEN SEGHIER	M A B	Examinateur

Remerciements

Avant tout, notre sincère louange à ALLAH le tout puissant qui m'a donné la fois, la volonté, la santé et la patience pour accomplir ce mémoire.

Un grand merci à mes parents qui m'ont apporté support et soutient tout au long de mes études.

Mes vifs remerciements vont à mon encadreur ***Dr. SAOULI Hamza****, pour sa disponibilité, son suivi, ses critiques constructives, ses explications et suggestions.*

Je veux également remercier tous les membres de jury :

Dr. TOUIL Keltoum*, Maître de conférences à l'Université de Biskra,*

Dr. BEN SEGHIER Nadia*, Maître de conférences à l'Université de Biskra, pour m'avoir fait l'honneur d'accepter de juger ce travail.*

Mes remerciements vont également à tous mes enseignants à travers mon parcours universitaire et enfin toutes les personnes qui m'ont, de près ou de loin, aidé pour la réalisation de ce travail.

Kheireddine Remadna

Résumé

L'utilisation massive des technologies de l'information et le développement de la cybercriminalité a conduit à la création d'une nouvelle branche dans le domaine de l'informatique : l'investigation numérique. Les techniques et méthodes d'investigation ne cessent d'évoluer et la diversité des supports numériques implique de suivre des règles et des procédures strictes, nécessaires à la recherche, la conservation et l'interprétation des données extraites. Cette branche initialement exercée par des experts judiciaires de la police nationale ou de la gendarmerie à des fins de perquisition numérique, s'étend et se développe au secteur privé dans le cadre d'audit de sécurité et de réponse à incident. Encore relativement nouvelle en Algérie, l'investigation numérique est de plus en plus demandée par des sociétés privées telles que les banques et les multinationales. Aujourd'hui, les criminels sont de plus en plus conscients des aspects judiciaires et sont maintenant capables de compromettre des ordinateurs sans accéder au disque dur de l'ordinateur cible. Ainsi de nombreuses sociétés spécialisées en sécurité informatique ont développé cette activité pour répondre à la demande croissante des entreprises.

L'objectif de ce travail de guider l'expert en charge de l'investigation au moyen de règles dites de « bonnes pratiques » mais également de faire découvrir les méthodes de forensic à toutes les personnes désirant développer leurs compétences dans ce domaine. Enfin, on développe et réalise un système qui facilite l'investigation numérique de la mémoire physique sur les systèmes Windows.

Mots-clés : L'investigation numérique, cybercriminalité, forensic, l'investigation numérique de la mémoire physique, judiciaires, réponse à incident

Table des matières

III. Investigations digitales sur la mémoire

IV. Conception du système

V. Implémentation

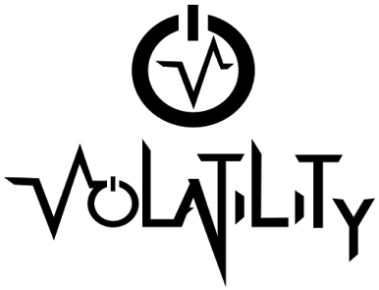
VI. Conclusion Générale

Abstract

The massive use of information technologies and the development of cybercrime in the field of computer science has led to the creation of a new branch: Digital forensics. The techniques and methods of investigation are evolving continually, the diversity of digital media requires strict rules, and procedures to be followed, which are needed to search, preserve and interpret the extracted data. This branch, which was first exercised by judicial experts from the national police or the gendarmerie for the purpose of digital searches, is now being extended and developed to the private sector as part of a security audit and incident response. Digital investigation is still relatively new in Algeria, and is increasingly being requested by private companies such as banks and multinationals.More recently, criminals are becoming more forensically aware and are now able to compromise computers without accessing the hard disk of the target computer. As such, many companies specialized in IT, security have developed this field to meet the growing demand of companies.

The objective of this work is to guide the expert in charge of investigation by means of so-called "good practice" rules but also to introduce forensic methods to all persons wishing to develop their skills in this field. Finally, a system is developed and implemented to facilitate the digital investigation of physical memory on Windows systems.

Keywords: Digital forensics, Digital evidence, Incident Response, Physical memory forensic and analysis, Volatile memory, Memory analysis Tools, Cybercrime.



Development build and wiki:
github.com/volatilityfoundation

Download a stable release:
volatilityfoundation.org

Read the book:
artofmemoryforensics.com

Development Team Blog:
http://volatility-labs.blogspot.com

(Official) Training Contact:
voltraining@memoryanalysis.net

Follow: @volatility
Learn: www.memoryanalysis.net

Basic Usage

Typical command components:
vol.py -f [image] --profile=[profile] [plugin]

Display profiles, address spaces, plugins:
vol.py --info

Display global command-line options:
vol.py --help

Display plugin-specific arguments:
vol.py [plugin] --help

Load plugins from an external directory:
vol.py --plugins=[path] [plugin]

Specify a DTB or KDBG address:
vol.py --dtb=[addr] --kdbg=[addr]

Specify an output file:
vol.py --output-file=[file]

Image Identification

Get profile suggestions (OS and architecture):
imageinfo

Find and parse the debugger data block:
kdbgscan

Processes Listings

Basic active process listing:
pslist

Scan for hidden or terminated processes:
psscan

Cross reference processes with various lists:
psxview

Show processes in parent/child tree:
pstree

Process Information

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Display DLLs:
dlllist

Show command line arguments:
cmdline

Display details on VAD allocations:
vadinfo [--addr]

Dump allocations to individual files:
vaddump --dump-dir=PATH [--base]

Dump all valid pages to a single file:
memdump --dump-dir=PATH

Display open handles:
handles
-t/--object-type=TYPE Mutant, File, Key, etc…
-s/--silent Hide unnamed handles

Display privileges:
privs
-r/--regex=REGEX Regex privilege name
-s/--silent Explicitly enabled only

Display SIDs:
getsids

Display environment variables:
envars

PE File Extraction

Specify -D/--dump-dir to any of these plugins to identify your desired output directory.

Dump a kernel module:
moddump
-r/--regex=REGEX Regex module name
-b/--base=BASE Module base address

Dump a process:
procdump
-m/--memory Include memory slack

Dump DLLs in process memory:
dlldump
-r/--regex=REGEX Regex module name
-b/--base=BASE Module base address

Injected Code

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Find and extract injected code blocks:
malfind
-D/--dump-dir=PATH Dump findings here

Cross-reference DLLs with memory mapped files:
ldrmodules

Scan a block of code in process or kernel memory for imported APIs:
impscan
-p/--pid=PID Process ID
-b/--base=BASE Base address to scan
-s/--size=SIZE Size to scan from start of base

Logs / Histories

Recover event logs (XP/2003):
evtlogs
-S/--save-evt Save raw event logs
-D/--dump-dir=PATH Write to this directory

Recover command history:
cmdscan and consoles

Recover IE cache/Internet history:
iehistory

Show running services:
svcscan
-v/--verbose Show ServiceDll from registry

Networking Information

Active info (XP/2003):
connections and sockets

Scan for residual info (XP/2003):
connscan and sockscan

Network info for Vista, 2008, and 7:
netscan

Kernel Memory

Display loaded kernel modules:
modules

Scan for hidden or residual modules:
modscan

Display recently unloaded modules:
unloadedmodules

Display timers and associated DPCs:
timers

Display kernel callbacks, notification routines:
callbacks

Audit the SSDT
ssdt
-v/--verbose Check for inline API hooks

Audit the IDT and GDT:
idt (x86 only)
gdt (x86 only)

Audit driver dispatch (IRP) tables:
driverirp
-r/--regex=REGEX Regex driver name

Display device tree (find stacked drivers):
devicetree

Print kernel pool tag usage stats:
pooltracker
-t/--tags=TAGS List of tags to analyze
-T/--tagfile=FILE pooltag.txt for labels

Kernel Objects

Scan for driver objects:
driverscan

Scan for mutexes:
mutantscan
  -s/--silent Hide unnamed mutants

Scan for used/historical file objects:
filescan

Scan for symbolic link objects (shows drive mappings):
symlinkscan

Registry

Display cached hives:
hivelist

Print a key's values and data:
printkey
  -o/--hive_offset=OFFSET Hive address (virtual)
  -K/--key=KEY Key path

Dump userassist data:
userassist

Dump shellbags information:
shellbags

Dump the shimcache:
shimcache

Timelines

To create a timeline, create output in body file format. Combine the data and run sleuthkit's mactime to create a CSV file.

timeliner --output=body > time.txt
shellbags --output=body >> time.txt
mftparser --output=body >> time.txt

mactime –b [time.txt] [-d] > csv.txt

Volshell

List processes:
>>> ps()

Switch contexts by pid, offset, or name:
>>> cc(pid = 3028)
>>> cc(offset = 0x3eb31340, physical=True)
>>> cc(name = "explorer.exe")

Acquire a process address space after using cc:
>>> process_space = proc().get_process_address_space()

Disassemble data in an address space
>>> dis(address, length, *space*)

Dump bytes, dwords or qwords:
>>> db(address, length, *space*)
>>> dd(address, length, *space*)
>>> dq(address, length, *space*)

Display a type/structure:
>>> dt("_EPROCESS", recursive = True)

Display a type/structure instance:
>>> dt("_EPROCESS", 0x820c92a0)

Create an object in kernel space:
>>> thread = obj.Object("_ETHREAD", offset = 0x820c92a0, vm = addrspace())

Dump Conversion

Create a raw memory dump from a hibernation, crash dump, firewire acquisition, virtualbox, vmware snapshot, hpak, or EWF file:
imagecopy –O/--output-image=FILE

Convert any of the aforementioned file types to a Windows crash dump compatible with Windbg:
raw2dmp –O/--output-image=FILE

API Hooks

Scan for API hooks:
apihooks
  -R/--skip-kernel Don't check kernel modules
  -P/--skip-process Don't check processes
  -Q/--quick Scan faster

Yara Scanning

Scan for Yara signatures:
yarascan
  -p/--pid=PID Process IDs to scan
  -K/--kernel Scan kernel memory
  -Y/--yara-rules=RULES String, regex, bytes, etc.
  -y/--yara-file=FILE Yara rules file
  -W/--wide Match Unicode strings
  -s/--size Size of preview bytes

File System Resources

Scan for MFT records:
mftparser
  --output=body Output body format
  -D/--dump-dir Dump MFT-resident data

Extract cached files (registry hives, executables):
dumpfiles
  -D/--dump-dir=PATH Output directory
  -r/--regex=REGEX Regex filename

Parse USN journal records:
usnparser (github.com/tomspencer)

GUI Memory

Sessions (shows RDP logins):
sessions

Window stations (shows clipboard owners):
wndscan

Desktops (find ransomware):
Deskscan

Display global and session atom tables:
atoms and atomscan

Dump the contents of the clipboard:
clipboard

Detect message hooks (keyloggers):
messagehooks

Take a screen shot from the memory dump:
screenshot --dump-dir=PATH

Display visible and hidden windows:
windows and wintree

Strings

Use GNU strings or Sysinternals strings.exe:
strings -a -td FILE > strings.txt
strings -a -td -el FILE >> strings.txt (Unicode)

strings.exe -q -o > strings.txt (Windows)

Translate the string addresses:
strings
  -s/--string-file=FILE Input strings.txt file
  -S/--scan

Password Recovery

Dump LSA secrets:
lsadump

Dump cached domain hashes:
cachedump

Dump LM and NTLM hashes:
hashdump (x86 only)

Extract OpenVPN credentials:
openvpn (github.com/Phaeilo)

Extract RSA private keys and certificates:
dumpcerts
  -s/--ssl Parse certificates with openssl

Disk Encryption

Recover cached TrueCrypt passphrases:
truecryptpassphrase

Triage TrueCrypt artifacts:
truecryptsummary

Extract TrueCrypt master keys
truecryptmaster

Malware Specific

Dump Zeus/Citadel RC4 keys:
zeusscan and citadelscan

Find and decode Poison Ivy configs:
poisonivyconfig

Decode Java RAT config:
javaratscan (github.com/Rurik)

General Investigations	
Dump the system's raw registry hive files	dumpfiles -p 4 --regex='(config|ntuser)' --ignore-case --name -D ./
Create a Graphviz diagram of processes	psscan --output=dot --output-file=graph.dot
Create a color coded diagram of processes memory	vadtree -p PID --output=dot --output-file=graph.dot
Translate an account SID to user name	printkey -K "Microsoft\\Windows NT\\CurrentVersion\\ProfileList\\[SID]" | grep ProfileImagePath
List run keys for HKLM and all users	printkey -K "Microsoft\\Windows\\CurrentVersion\\Run" printkey -K "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
Find Unicode hostnames or URLs	yarascan -Y "/(www|http).+\.(com|net|org)/" --wide [--kernel]
Find null-terminated ASCII dot quad IP addresses	yarascan -Y "/([0-9]{1,3}\.){3}[0-9]{1,3}\x00/" --wide [--kernel]
Locate and extract the HOSTS file to local directory	filescan | egrep hosts$ | awk '{print $1}' 0x0000000005e3c6d8 dumpfiles -Q 0x0000000005e3c6d8 --name -D ./
Extract the admin password hash	hashdump | grep Administrator > admin.txt
Malicious Code	
Check if a process has domain or enterprise admin	getsids | egrep '(Domain|Enterprise)'
Identify processes with raw sockets	handles -t File | grep "\\Device\\RawIp\\0"
Look for explicit enabled debug privilege	privs --silent --regex=debug
Identify alternate data streams	mftparser | grep "DATA ADS"
Dump MFT-resident batch scripts	mftparser -D output/ file output/* | grep "DOS batch file"
Determine what is spying on the clipboard	wndscan | grep ClipViewer
Dump injected code and focus on executables	malfind -D output/ file output/* | grep PE
Trace API hooks through memory	apihooks -p PID --quick | grep 'Hook address' 0x1da654f echo "dis(0x1da654f, length = 512)" | volshell -p PID
Scan for a specific mutex on the system	mutantscan | grep [-i] [MUTANT NAME]
Dump injected DLL, fix image base + IDA import labels	dlldump --base=ADDR -p PID -D./ --fix –memory impscan --base=ADDR -p PID --output=idc > labels.idc
Find binaries loaded from temporary directories	envars -p PID | grep TEMP | awk '{print $5}' C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp Filter dlllist and modules output for the specified path
User Activity	
Detect remote mapped shares	handles -t File | egrep "\\Device\\(LanmanRedirector|Mup)"
Files on Truecrypt volumes	filescan | grep TrueCryptVolume
Extract ASCII and Unicode clipboard content	clipboard | grep TEXT
Brute force search for command history	yarascan -Y "/C:\\\\.+>/" --wide [--kernel]
Recently clicked applications and shortcuts	userassist | grep REG_BINARY
Find prefetch files (recently executed programs)	mftparser | grep \.pf$ | awk '{print $NF}'
Kernel Memory	
Identify hooked driver dispatch tables	driverirp --regex=tcpip | grep IRP | egrep -vi '(tcpip|ntos)'
Look for hooked SSDT functions	ssdt | egrep –vi '(ntos|win32k) '
Malicious kernel callbacks and timers	callbacks | grep UNKNOWN (same with timers)
Locate hidden thread-based kernel rootkits	threads -F OrphanThread | grep StartAddress
Speed Enhancements	
Find and set the kernel DTB	psscan | grep System | awk '{print $5}' 0x00319000 (Now use --dtb=0x00319000)
Find and set the KDBG on XP-7 and 32-bit 8	kdbgscan | grep Offset | grep V | uniq Offset (V) : 0xf80002803070 (add to --kdbg)
Find and set the KDBG on 64-bit 8 and 2012	kdbgscan --profile=[PROFILE] | grep KdCopyDataBlock KdCopyDataBlock (V) : 0xf80281ff5ea0 (add to --kdbg)
Volshell Scripting	
Create a process ID lookup table	by_pid = dict((p.UniqueProcessId, p) for p in getprocs()) parent_name = by_pid[PID].ImageFileName
Scan process memory and print a hex dump	needles = ["abc123", "def456"] for hit in proc().search_process_memory(needles): db(hit)
Extract a chunk of kernel memory to disk	data = addrspace().zread(ADDR, SIZE) with open("output.bin", "wb") as handle: handle.write(data)
Translate a kernel address and seek to it (raw dumps only)	echo "addrspace().vtop(0x98dfd9c8)" | volshell -f [MEMDUMP] 597989832 xxd -s 597989832 [MEMDUMP]
Kernel modules with embedded PE signatures	signed = [mod for mod in getmods() if mod.sec_dir()]

Linux Commands

Processes Listings

Basic active process listing:
linux_pslist

List processes and threads:
linux_pidhashtable

Cross reference processes with various lists:
linux_psxview

Show processes in parent/child tree:
linux_pstree

Process Information

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Display shared libraries:
linux_library_list

List threads:
linux_threads

Show command line arguments:
linux_psaux

Display details on memory ranges:
linux_proc_maps

Dump allocations to individual files:
linux_dump_map
-D/--dump-dir=PATH
--vma=ADDR Range to dump

Display open handles:
linux_lsof

Display environment variables:
linux_psenv and linux_bash_env

ELF File Extraction

Specify -D/--dump-dir to any of these plugins to identify your desired output directory.

Dump a kernel module:
linux_moddump
-r/--regex=REGEX Regex module name
-b/--base=BASE Module base address

Dump a process:
linux_procdump

Dump shared libraries in process memory:
linux_librarydump
-r/--regex=REGEX Regex module name
-b/--base=BASE Module base address

Injected Code

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Find and extract injected code blocks:
linux_malfind

Cross-reference shared libraries with memory-mapped files:
linux_ldrmodules

Check for process hollowing:
linux_process_hollow
-b/--base Base address of ELF file in memory
-P/--path Path of known good file on disk

Command History

Recover command history:
linux_bash

Recover executed binaries:
linux_bash_hash

Networking Information

Active info:
linux_netstat

Interface information:
linux_ifconfig

Raw sockets:
linux_list_raw

Routing cache:
linux_route_cache
-R/--resolve DNS resolve destination IPs

Netfilter entries:
linux_netfilter

ARP cache:
linux_arp

Kernel Memory

Display loaded kernel modules:
linux_lsmod

Check for system call hooks:
linux_check_syscall

Check for network stack hooks:
linux_check_afinfo

Check for credential copying:
linux_check_creds

Check for file operations hooking:
linux_check_fop

Check for inline kernel hooks:
linux_check_inline_kernel

Check for hidden modules:
linux_check_modules
linux_hidden_modules

Check for TTY hooks:
linux_check_tty

Check for malicious keyboard callbacks:
linux_keyboard_notifiers

Print the kernel debug buffer:
linux_dmesg

Audit the IDT:
linux_idt (x86 only)

Userland API Hooks

Scan for API hooks:
linux_apihooks
-a/--all Check hooked PLT entries

Scan for GOT/PLT hooks:
linux_plthook
-a/--all List all PLT entries
-i/--ignore Libraries to ignore in processing

Yara Scanning

Scan for Yara signatures:
linux_yarascan
-p/--pid=PID Process IDs to scan
-K/--kernel Scan kernel memory
-Y/--yara-rules=RULES String, regex, bytes, etc.
-y/--yara-file=FILE Yara rules file
-W/--wide Match Unicode strings
-s/--size Size of preview bytes

File System Resources

List mount points:
linux_mount

Enumerate files:
linux_enumerate_files

Extract cached files:
linux_find_file
-F/--find=FILE Path of file to find
-i/--inode=INODE Address of inode to dump
-L/--listfiles Lists files in cache
-O/--outputfile File path to write

Disk Encryption

Recover cached Truecrypt passphrases:
linux_truecryptpassphrase

Strings

Translate extracted strings:
linux_strings
-s/--string-file=FILE Input strings.txt file

Mac OS X Commands

Processes Listings

Basic active process listing:
mac_pslist

List PID hash table:
mac_pid_hash_table

List tasks:
mac_tasks

Cross reference processes with various lists:
mac_psxview

Show processes in parent/child tree:
mac_pstree

Process Information

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Display shared libraries:
mac_dyld_maps

Show command line arguments:
mac_psaux

Display details on memory ranges:
mac_proc_maps

Dump allocations to individual files:
mac_dump_map
-D/--dump-dir=PATH
--map_address=ADDR

Display open handles:
mac_lsof

Display environment variables:
mac_psenv and mac_bash_env

Display login sessions:
mac_list_sessions

Mach-O File Extraction

Specify -D/--dump-dir to any of these plugins to identify your desired output directory.

Dump a kernel module:
mac_moddump
-r/--regex=REGEX Regex module name
-b/--base=BASE Module base address

Dump a process:
mac_procdump

Dump shared libraries in process memory:
mac_librarydump
-b/--base=BASE Module base address

Injected Code

Specify –o/--offset=OFFSET or -p/--pid=1,2,3

Find and extract injected code blocks:
mac_malfind

Cross-reference shared libraries with memory-mapped files:
mac_ldrmodules

Command History

Recover command history:
mac_bash

Recover executed binaries:
mac_bash_hash

Networking Information

Active info:
mac_netstat

Active info from network stack:
mac_network_conns

Interface Information:
mac_ifconfig

ARP cache:
mac_arp

Route table:
mac_route

Socket filters:
mac_socket_filters

IP filters:
mac_ip_filters

Kernel Memory

Display loaded kernel modules:
mac_lsmod

Check for kernel API hooks:
mac_apihooks_kernel

Check for system call hooks:
mac_check_syscalls

Check for shadow system call table:
mac_check_syscall_shadow

Check sysctl handlers:
mac_check_sysctl

Check the trap table:
mac_check_trap_table

Check the mig table:
mac_check_mig_table

Check for file operations hooking:
mac_check_fop

Check for inline kernel hooks:
mac_check_inline_kernel

Check for hidden modules:
mac_lsmod_iokit
mac_lsmod_kext_map

Check for TrustedBSD hooks:
mac_trustedbsd

Print the kernel debug buffer:
mac_dmesg

API Hooks

Scan for API hooks:
mac_apihooks
-R/--skip-kernel Don't check kernel modules
-P/--skip-process Don't check processes
-Q/--quick Scan faster

Check for process hollowing:
mac_process_hollow
-b/--base Base address of ELF file in memory
-P/--path Path of known good file on disk

Scan for GOT/PLT hooks:
mac_plthook
-a/--all List all PLT entries
-i/--ignore Libraries to ignore in processing

Yara Scanning

Scan for Yara signatures:
mac_yarascan
-p/--pid=PID Process IDs to scan
-K/--kernel Scan kernel memory
-Y/--yara-rules=RULES String, regex, bytes, etc.
-y/--yara-file=FILE Yara rules file
-W/--wide Match Unicode strings
-s/--size Size of preview bytes

Disk Encryption

Recover possible Keychain keys:
mac_keychaindump

File System Resources

List mount points:
mac_mount

List cached files and their vnode addresses:
mac_list_files

Extract cached files:
mac_dump_file
-q/--file_offset Offset of vnode to dump
-O/--outputfile File path to write

Strings

Translate extracted string:
mac_strings
-s/--string-file=FILE Input strings.txt file

User Activity

Recover Adium messages, including OTR chat:
mac_adium

Recover Calendar entries:
mac_calendar

Recover contacts:
mac_contacts

MANUEL D'UTILISATION DE L'APPLICATION

REMADNA KHEIREDDINE

[ZODIAC FORENSICS] [kheireddinejava1992@gmail.com]

USER MANUAL FOR APPLYING "FORENSIC AND TECHNICAL POLICE"

Copyright

Disclaimer

This document is provided to you for informational purposes only and is believed to be accurate as of the date of its publication, and is subject to change without notice. The author assumes no responsibility for any errors or omissions in this document and shall have no obligation to you as a result of having made this document available to you or based upon the information it contains.

Version Information

Zodiac version 1.0.5

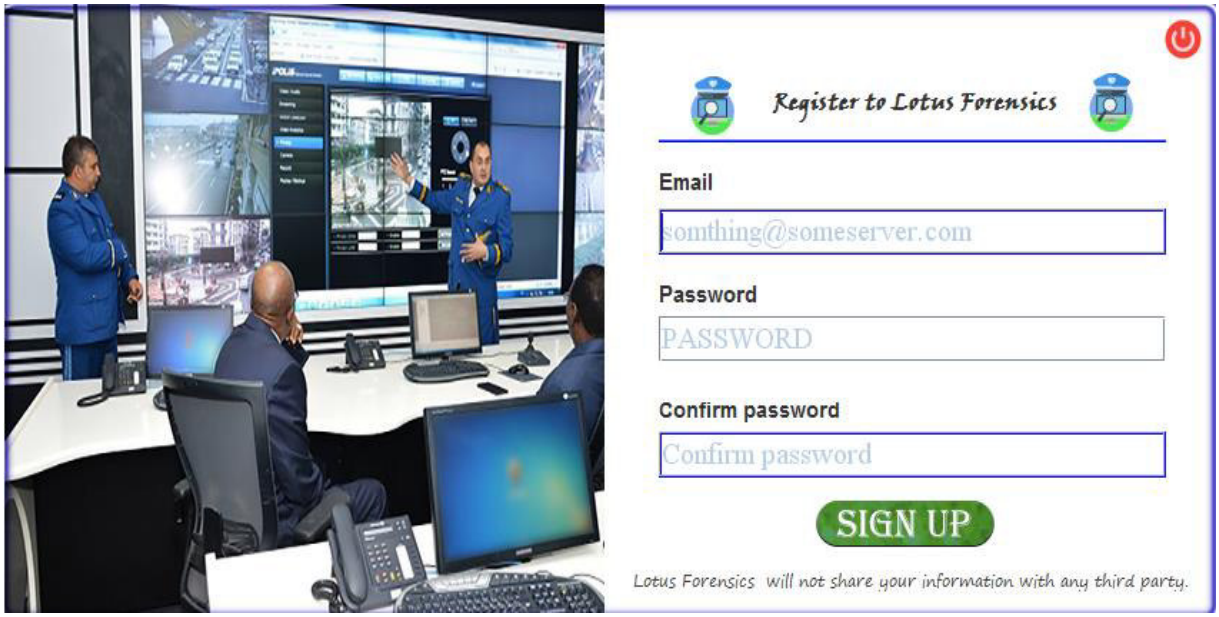
Document version 1.1

SECTION I: FOR INDIVIDUAL

Step 1. Start the Zodiac Forensic Browser tool

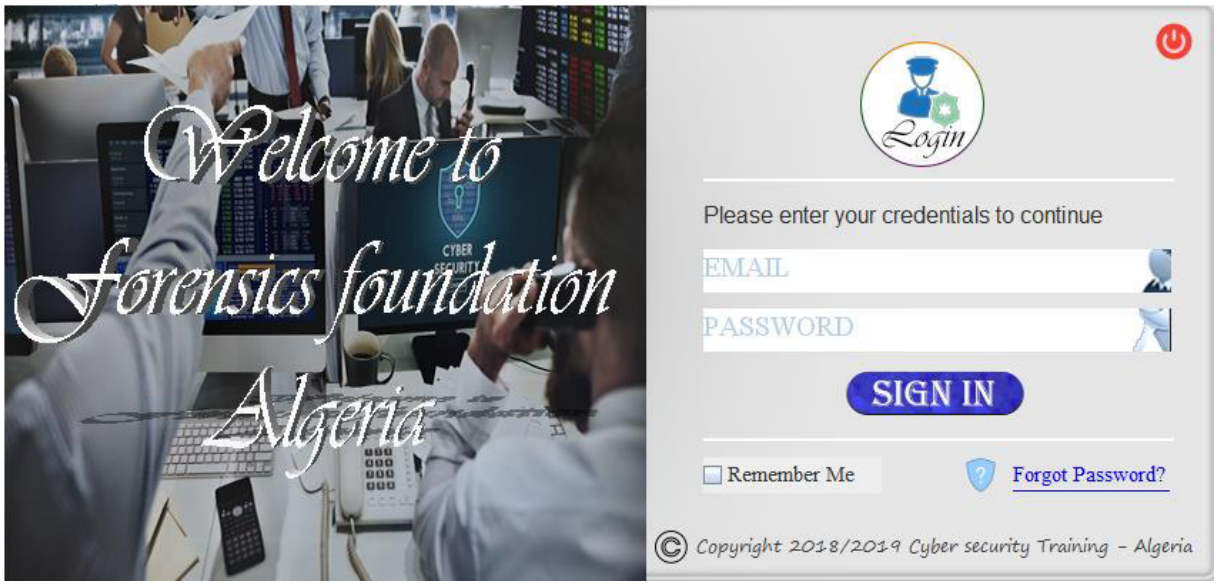
Step 2. Click on “Register” button given on the home interface.

Step 3. One Registration in forensic tool



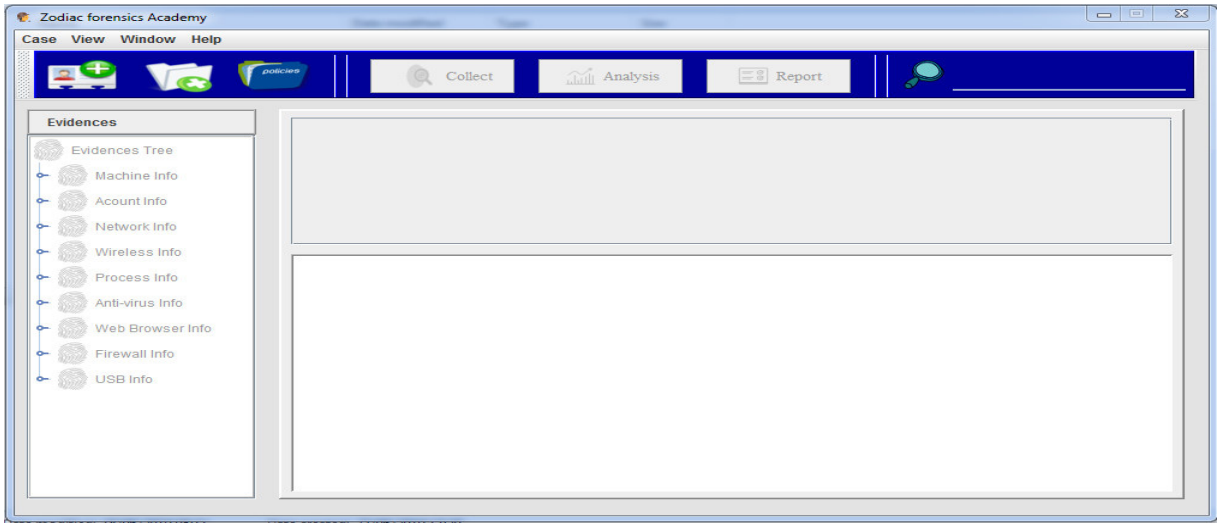
Step 4. Fill all given details for e.g. your name, email, password and click on “sign up” button.

Step 5. Once you are registered. Go to the login interface whenever you want to open your private account.

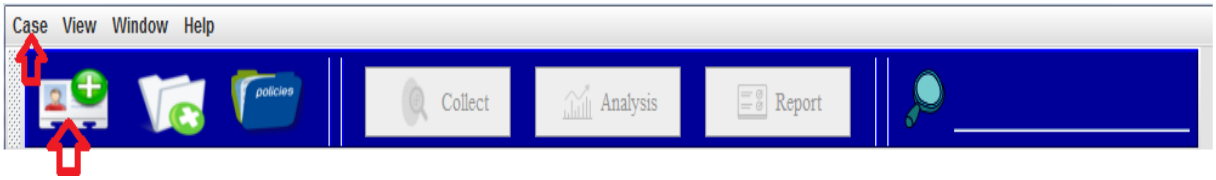


- Login with your email id and your password. One application form will be opened.

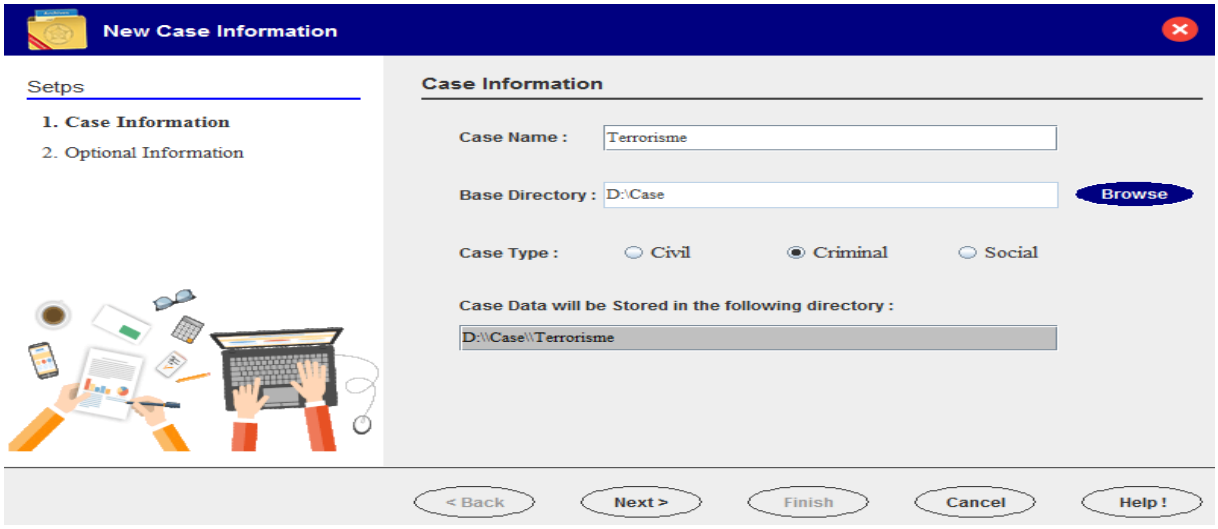
Step 6. If a person is authorized, the tool will guide you to the main interface.



Step 7. To **create a case**, use the "**Create New Case**" option either on the Welcome screen or from the "**File**" menu.

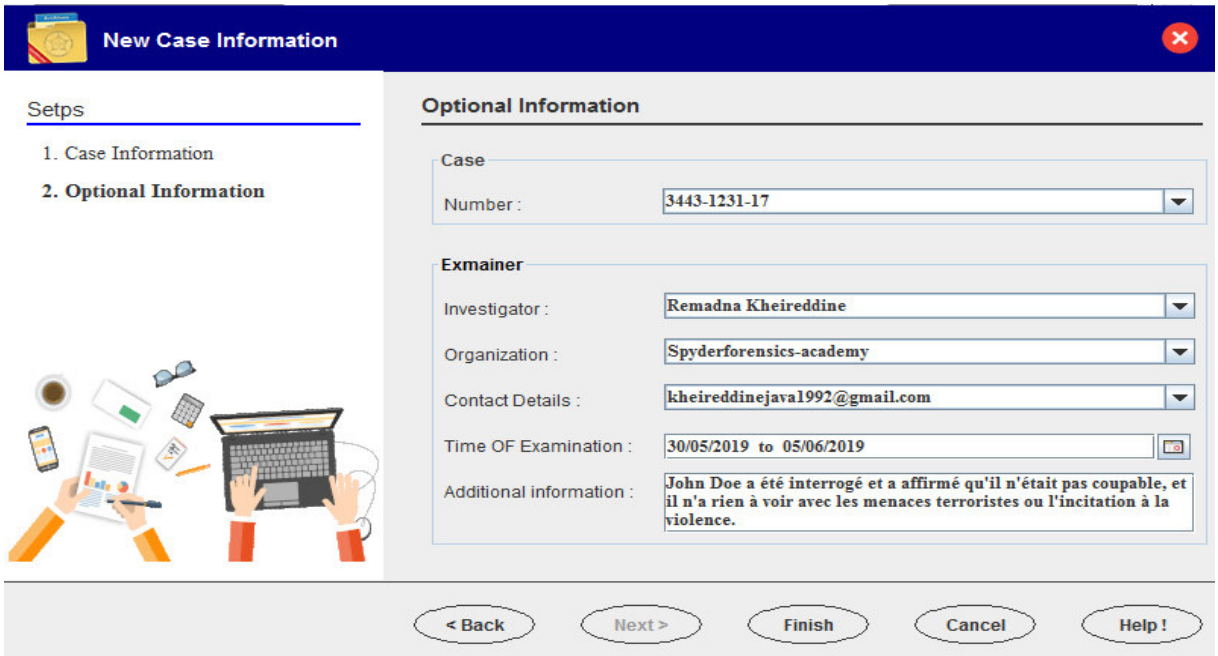


Step 8. The New Case Information window appears. Enter the Case Details



1. In the **Case Name** box, type a descriptive name for the case.
2. Click **Browse** and change to the directory where you want to store the case files and click **Select**.
3. In the **New Case Information** window, click **Next**.

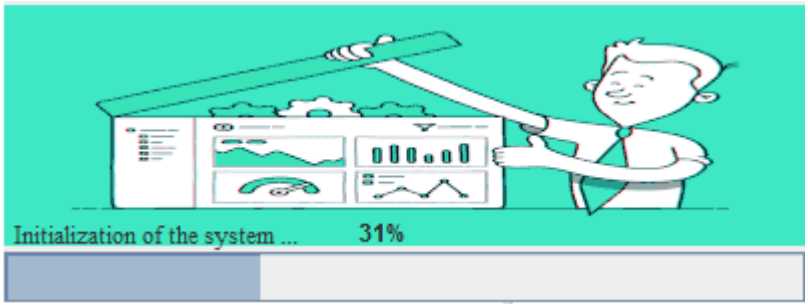
Step 9. The **New Case Information** window updates to show **Additional Information**. You will also be prompted for **optional information**, such as investigator name and case number.



Note

If you do not add the case number and examiner when you create the case, you cannot add it to the case later.

1. Click **Finish**. The tool will automatically generate a directory for the case in the "base directory of the legal USB key". The directory will contain configuration files, a database, and other files generated by the add-ons.

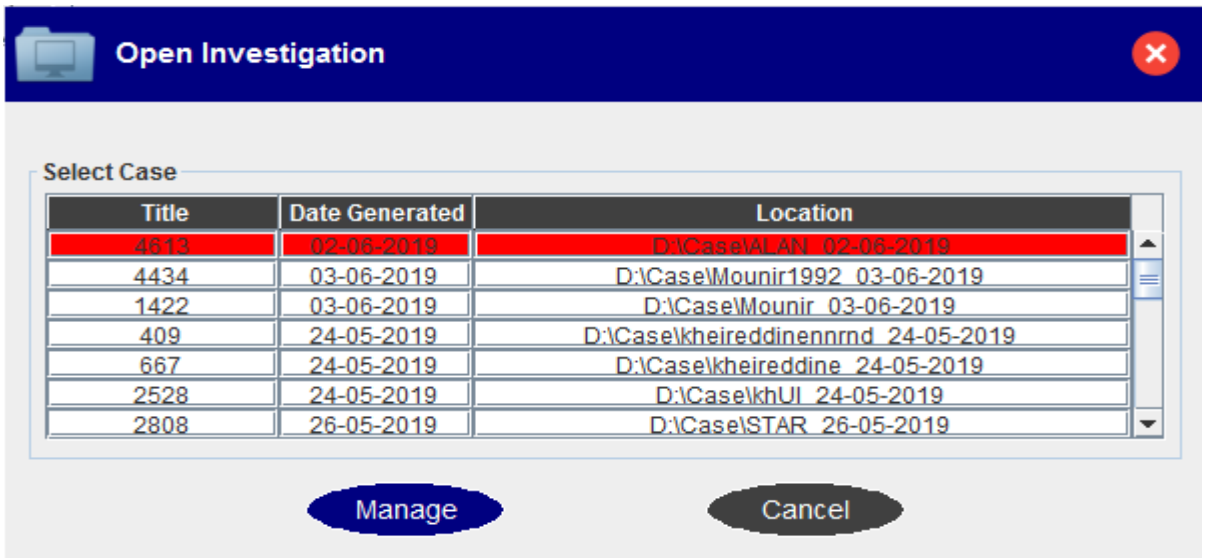


- **Opening a Case**

1. To open a case, either:

- Choose "**Open Existing Case**" from the opening splash screen.

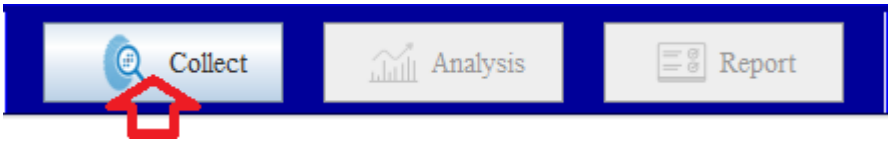
2. **Navigate** to the case directory and select the **available file**.



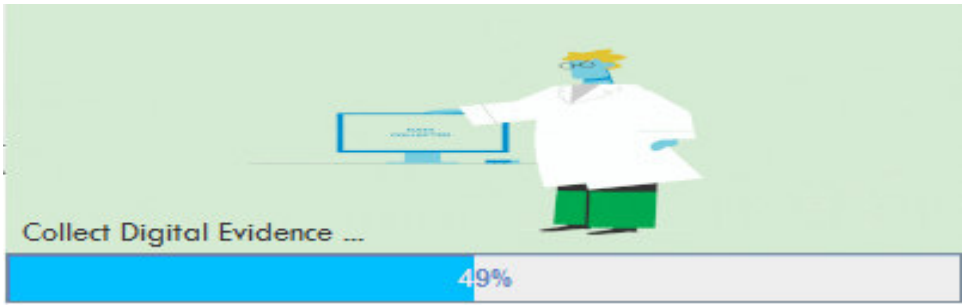
- Click **Mange**. The tool **automatically** adds an image of the cases you would like to process.

Step 10. The following steps are digital investigation steps on the physical memory of the computer:

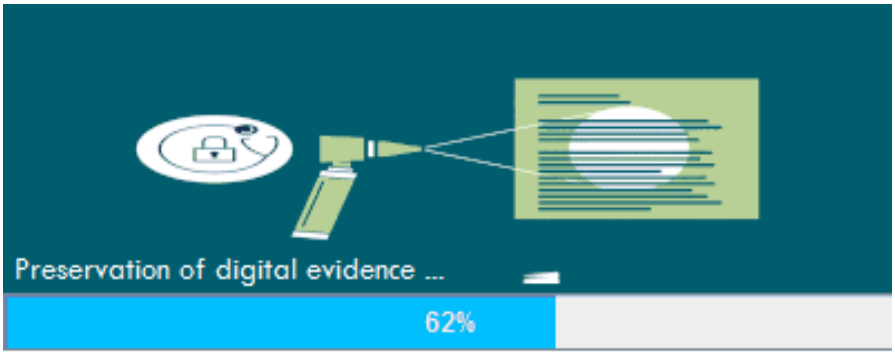
I. The tools in this category allow the analyst to make a **faithful** copy of a digital support on RAM memory. They also allow you to copy the files protected by the system, you click on the **Collect** button (See Figure).



- The tool **automatically** generates a **reliable** copy of the digital media on **RAM**.



- After completing the process of acquiring evidence, the tool will **automatically calculate** its **fingerprint** (**MD5** and/or **SHA1**) to ensure the integrity of the file during acquisition.



- Before starting the forensic analysis process, the tool confirms automatically that the copied data are not modified during the investigation.

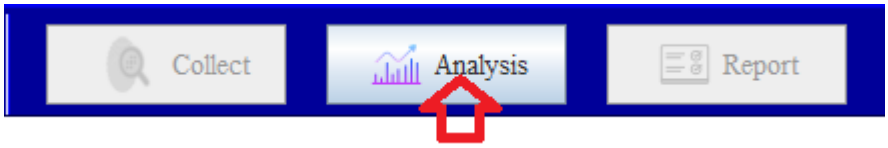
Note

If there are any changes in the case during the investigation:

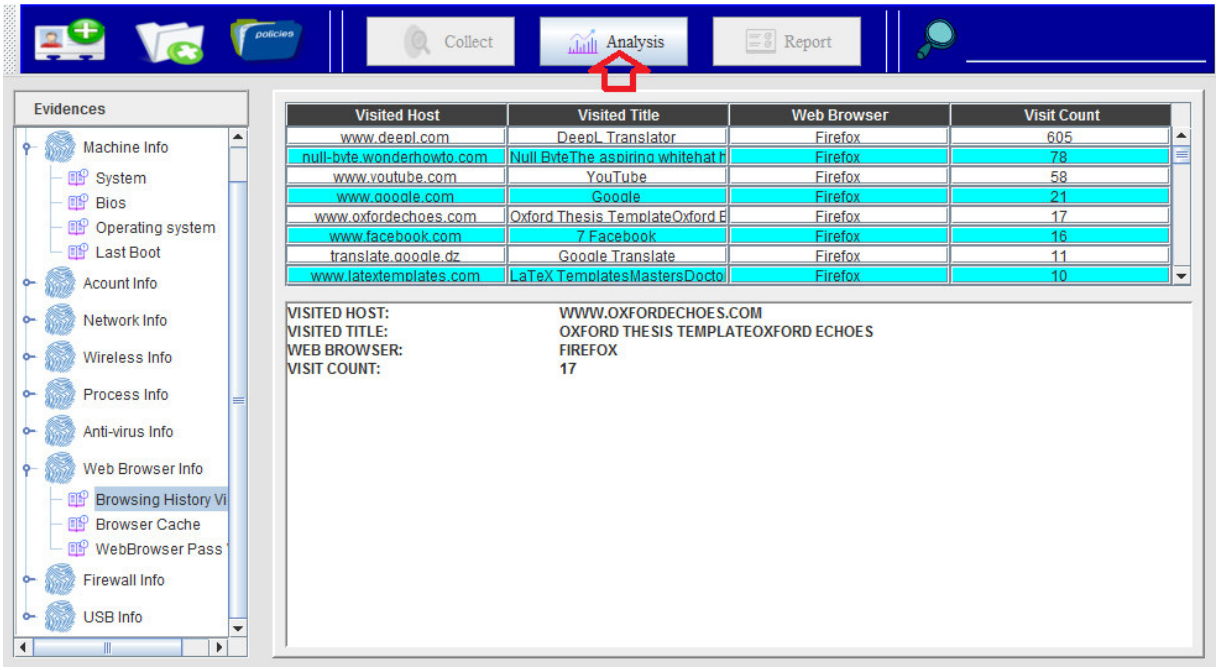
To solve the problem, you need to recreate a new image of the case.

II. The tools available in this category allow the investigator to analyze the content of a web browser, the file system, active processes, system logs and events.

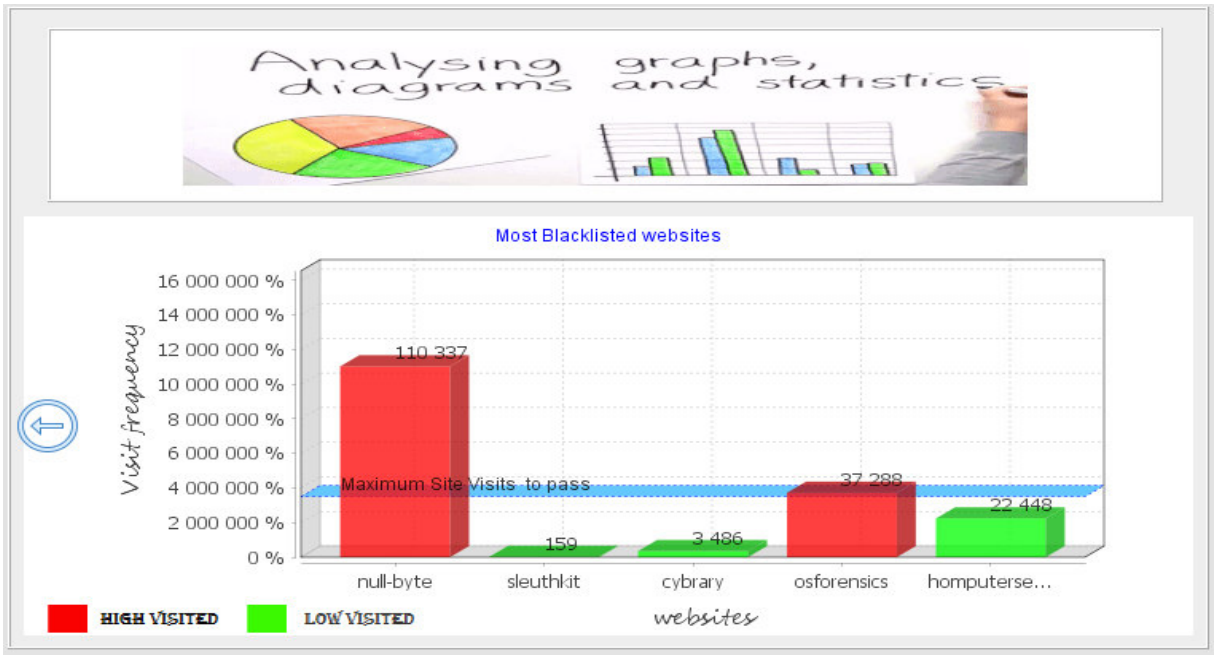
Click on the **Analysis** button. (See Figure).



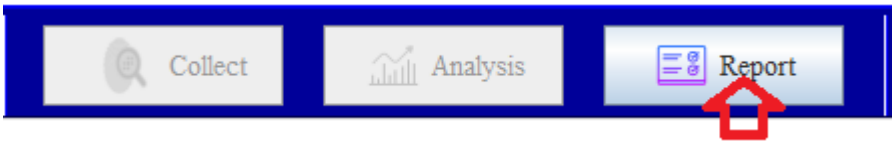
- Example for the analysis of web browser artifacts.



- Display of the result in the form of a graphical chart diagram.

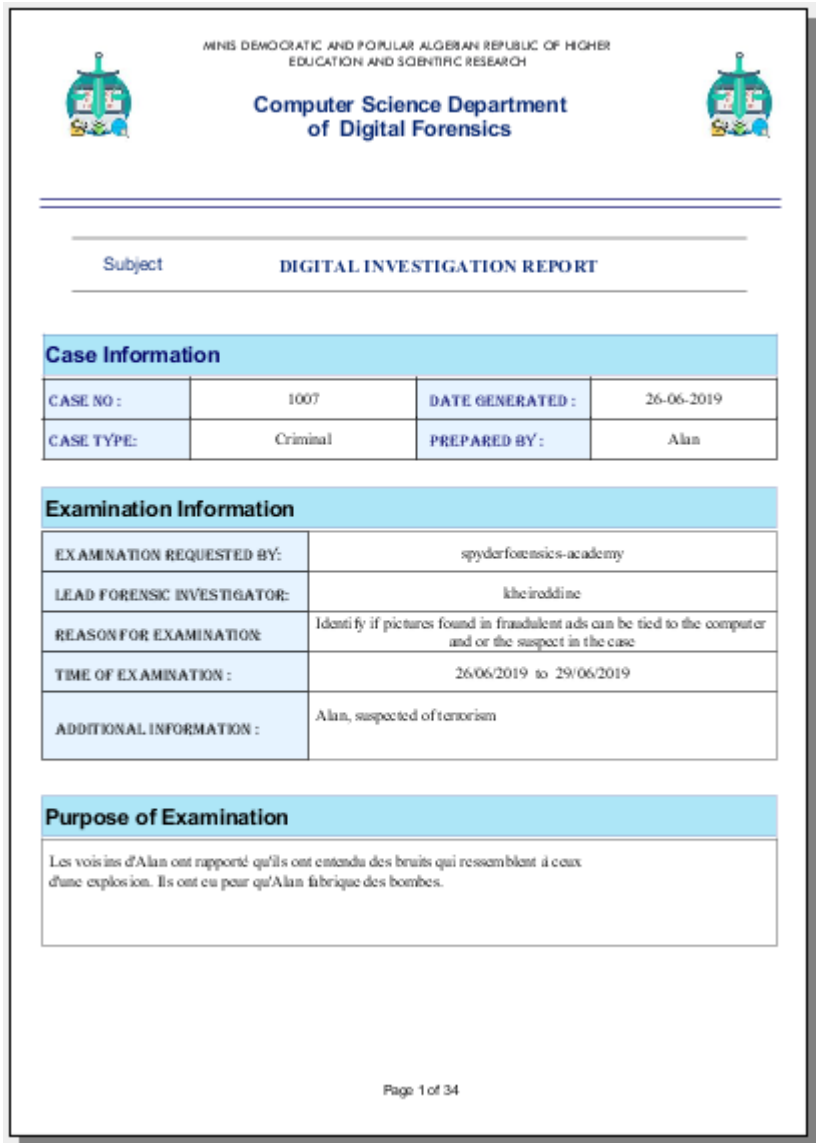


III. The tools in this category allow the investigator to document all previous phases and generate reports and conclusions for presentation to the respective jury or management units. Click on the **Report** button. (See Figure).



- The tool gives a simple report of the findings and evidence obtained during a criminal investigation in order to make them accessible to all. Click on the **Show** button. (See figure).

- Sample Investigation Report



MINIS DEMOCRATIC AND POPULAR ALGERIAN REPUBLIC OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Computer Science Department of Digital Forensics

Subject **DIGITAL INVESTIGATION REPORT**

Case Information

CASE NO :	1007	DATE GENERATED :	26-06-2019
CASE TYPE:	Criminal	PREPARED BY :	Alan

Examination Information

EXAMINATION REQUESTED BY:	spyderforensics-academy
LEAD FORENSIC INVESTIGATOR:	kheireddine
REASON FOR EXAMINATION:	Identify if pictures found in fraudulent ads can be tied to the computer and or the suspect in the case
TIME OF EXAMINATION :	26/06/2019 to 29/06/2019
ADDITIONAL INFORMATION :	Alan, suspected of terrorism

Purpose of Examination

Les voisins d'Alan ont rapporté qu'ils ont entendu des bruits qui ressemblent à ceux d'une explosion. Ils ont eu peur qu'Alan fabrique des bombes.

Page 1 of 34

MINIS DEMOCRATIC AND POPULAR ALGERIAN REPUBLIC OF
HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Computer Science Department of Digital Forensics

Subject DIGITAL INVESTIGATION REPORT

Case Information

Case No :	1918	Date Generated :	06-07-2019
Case Type:	Criminal	Case name:	alan

Examination Information

Examination Requested By:	spyderforensics-academy
Lead Forensic Investigator:	kheireddine
Reason For Examination:	the Person suspected of terrorism case
Time Of Examination :	06/07/2019 to 16/07/2019
Additional Information :	The suspect, Alan, was suspected of being aggressive in politics, he has always browsed foreign websites on terrorist attacks.

Purpose of Examination

The neighbours of Alan reported that they heard noises that sounded like an explosion. They were worried that Alan would make bombs.

Machine Data Systems

INVESTIGATION REPORT | Printed on the July 13, 2019

Machine Information

System Information

ComputerName	Manufacturer	Model	Domain
KHAIRY-PC	Acer	Aspire E1-571G	WORKGROUP

BIOS Information

ComputerName	SMB Version	Manufacturer BOIS	Version
KHAIRY-PC	V215	Insyde Corp	ACRSYS - 1

Date Installation & OS version

ComputerName	OS	ServicePack	InstallationDate
KHAIRY-PC	Microsoft Windows 7	Service Pack 1	11/10/2018 05:56:21

Last Boot Time

ComputerName	Last Boot Time
KHAIRY-PC	06/07/2019 08:31:15

Account Information

User Information

Name user	Type user	Disabled user	Password Changeable	Password Expires	Password Required
Administrator	512	True	True	False	True
Guest	512	True	False	False	False
khairy	512	False	True	False	False

Network Information

Configuration Interface

Interface_Name	Bytes_Out	Bytes_In	Media State	MTU
Loopback Pseudo-Interface	4910	0	1	4294967295
Wireless Network	953505	8152607	1	1500
Wireless Network Connection 2	0	0	5	1500
Local Area Connection	0	0	5	1500
Local Area Connection 2	0	0	5	1500
Npcap Loopback Adapter	30806	0	1	1500

TCP Statistics

Timeout Algorithm	Maximum Connections	Active Opens	Passive Opens	Range Num Ports	Retransmitted Segments	Attempts Failed
Van Jacobsons Algorithm	Dynamic	139	9	64510	70	33

TCP Statistics

TCP Connection Entry

Local Address	Local Port	Remote Address	Remote Port	State
169.254.203.121	139	0.0.0.0	0	Listen
192.168.8.100	139	0.0.0.0	0	Listen
192.168.8.100	1137	77.234.45.61	80	Established
192.168.8.100	1210	104.24.111.250	443	Time Wait

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:52:09	WmiPrvSE.exe	7404	
06/07/2019 08:52:07	conhost.exe	3488	\??\C:\Windows\system32\conhost.exe"-1972796597305382061946224418-879408236-207631413-
06/07/2019 08:52:06	powershell.exe	7952	powershell.exe -executionpolicy bypass -File G:\otuil\bath\Script_1.ps1 -DirG:\Repertoire\alan_06-07-2019
06/07/2019 08:51:48	notepad.exe	6260	"C:\Windows\system32\NOTEPAD.EXE" G:\inforamtion case.txt
06/07/2019 08:47:05	javaw.exe	6976	"C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe" -jar"G:\otuil\lance\ZodiacForensic_2019.exe"

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:46:42	javaw.exe	3560	"C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe" -jar"G:\otuil\lance\ZodiacForensic_2019.exe"
06/07/2019 08:44:20	WUDFHost.exe	340	
06/07/2019 08:39:35	wisptis.exe	7528	
06/07/2019 08:39:19	FoxitProxyServer_Socket_RD.exe	7696	"C:\Program Files (x86)\Foxit Software\FoxitReader\Plugins\Creator\FoxitProxyServer_Socket_RD.exe"]g•efhpmTlc
06/07/2019 08:39:17	POWERPNT.EXE	7616	"C:\Program Files\Microsoft Office\Office16\POWERPNT.EXE ""D:\Memoire\Daipo\presentation.pptx" /ou "u"
06/07/2019 08:38:01	svchost.exe	3128	
06/07/2019 08:36:40	CodeMeter.exe	1784	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:36:39	svchost.exe	5136	
06/07/2019 08:34:32	hsscp.exe	6632	"C:\Program Files (x86)\Hotspot Shield\bin\hsscp.exe"
06/07/2019 08:34:20	SearchIndexer.exe	7108	
06/07/2019 08:34:18	ICCProxy.exe	6824	
06/07/2019 08:34:16	aswidsagent.exe	6684	
06/07/2019 08:32:57	netcut_windows.exe	4512	
06/07/2019 08:32:57	conhost.exe	6016	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:32:52	ZeroConfigService.exe	1864	
06/07/2019 08:32:33	nvcontainer.exe	5328	"C:\Program Files (x86)\NVIDIA Corporation\NvContainer\nvcontainer.exe" -f"C:\ProgramData\NVIDIA\NvContainerUser%d.log" -d "C:\Program
06/07/2019 08:32:32	WmiPrvSE.exe	5232	
06/07/2019 08:32:28	WmiPrvSE.exe	4904	
06/07/2019 08:32:27	unsecapp.exe	4672	
06/07/2019 08:32:22	svchost.exe	4436	
06/07/2019 08:32:22	RegSrvc.exe	4408	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:32:22	NvTelemetryContainer.exe	4364	
06/07/2019 08:32:22	nvcontainer.exe	4300	
06/07/2019 08:32:19	IEMonitor.exe	4132	"C:\Program Files (x86)\Internet Download Manager\IEMonitor.exe"
06/07/2019 08:32:17	mysqld.exe	3740	
06/07/2019 08:32:13	BrMfcMon.exe	2468	"C:\Program Files (x86)\Brother\Brmfcmon\BrMfcmon.exe"
06/07/2019 08:32:12	BrccMCtl.exe	1032	"C:\Program Files (x86)\Brother\ControlCenter3\brccMCtl.exe" /autorun
06/07/2019 08:32:12	jusched.exe	3300	"C:\Program Files (x86)\Common Files\Java\Java Update\jusched.exe"

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:32:10	BrMfcWnd.exe	3960	"C:\Program Files (x86)\Brother\Brmfcmon\BrMfcWnd.exe" /AUTORUN
06/07/2019 08:32:10	AvastUI.exe	3888	AvastUI.exe /nogui
06/07/2019 08:32:10	IDMan.exe	3876	"C:\Program Files (x86)\Internet Download Manager\IDMan.exe" /onboot
06/07/2019 08:32:10	igfxpers.exe	3808	"C:\Windows\System32\igfxpers.exe"
06/07/2019 08:32:09	hkcmd.exe	3760	"C:\Windows\System32\hkcmd.exe"
06/07/2019 08:32:08	igfxtray.exe	3696	"C:\Windows\System32\igfxtray.exe"
06/07/2019 08:32:06	conhost.exe	3608	\??\C:\Windows\system32\conhost.exe"1068678056510558594-482187701-7566763-466779538-187513851-

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:32:06	NVIDIA Web Helper.exe	3600	"C:\Program Files (x86)\NVIDIA Corporation\NvNode\NVIDIA Web Helper.exe" index.js
06/07/2019 08:32:06	AvastBrowserCrashHandler64.exe	3504	
06/07/2019 08:32:06	AvastBrowserCrashHandler.exe	3496	
06/07/2019 08:32:05	GoogleCrashHandler.exe	3392	
06/07/2019 08:32:05	GoogleCrashHandler64.exe	3400	
06/07/2019 08:32:01	explorer.exe	3100	C:\Windows\Explorer.EXE
06/07/2019 08:32:01	dwm.exe	2012	"C:\Windows\system32\Dwm.exe"

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:32:01	taskhost.exe	2668	"taskhost.exe"
06/07/2019 08:32:00	cmw_srv.exe	3056	
06/07/2019 08:31:57	EvtEng.exe	2776	
06/07/2019 08:31:51	svchost.exe	2656	
06/07/2019 08:31:46	set.exe	2452	
06/07/2019 08:31:45	httpd.exe	1220	
06/07/2019 08:31:43	httpd.exe	1956	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:31:43	armsvc.exe	1920	
06/07/2019 08:31:42	svchost.exe	1772	
06/07/2019 08:31:42	spoolsv.exe	1732	
06/07/2019 08:31:41	AvastSvc.exe	1576	
06/07/2019 08:31:40	svchost.exe	1480	
06/07/2019 08:31:38	NVDisplay.Container.exe	1276	
06/07/2019 08:31:38	aips.exe	1232	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:31:37	audiodg.exe	1096	
06/07/2019 08:31:36	svchost.exe	896	
06/07/2019 08:31:36	svchost.exe	696	
06/07/2019 08:31:36	svchost.exe	612	
06/07/2019 08:31:36	svchost.exe	500	
06/07/2019 08:31:36	svchost.exe	308	
06/07/2019 08:31:35	NVDisplay.Container.exe	1004	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:31:35	svchost.exe	920	
06/07/2019 08:31:34	winlogon.exe	828	
06/07/2019 08:31:34	lsass.exe	764	
06/07/2019 08:31:34	lsm.exe	772	
06/07/2019 08:31:34	services.exe	748	
06/07/2019 08:31:34	wininit.exe	684	
06/07/2019 08:31:34	csrss.exe	704	

Last Activity View

Action Time	Process Name	Process Id	Command Line
06/07/2019 08:31:32	csrss.exe	572	
06/07/2019 08:31:27	smss.exe	464	
06/07/2019 08:31:27	System Idle Process	0	
06/07/2019 08:31:27	System	4	

Protecting Your Computer

Firewall Information

Profil_Name	Profile_State	Firewall_Policy	Inbound User Notification	Remote Management	Unicast Response To Multicast
Domain	ON	BlockInbound,Allow Outbound	Enable	Disable	Enable
Private	ON	BlockInbound,Allow Outbound	Enable	Disable	Enable
Public	ON	BlockInbound,Allow Outbound	Enable	Disable	Enable

Anti-virus Information

Computer_Name	AV_Name	Product_Executable	Status	Protection
KHAIRY-PC	Avast Antivirus	C:\Program Files\AVAST Software\Avast\wsc_proxy.	Up to date	Enabled

USB Information

USB Device

device name	device type	connected	serial number	last connected
USB Video Device	Video	No		11/10/2018 05:51:39
USB Input Device	HID (Human Interface Device)	No		05/07/2019 19:15:26
USB Input Device	HID (Human Interface Device)	No		05/07/2019 19:15:26
USB Mass Storage Device	Mass Storage	No		26/06/2019 09:46:40
Brother DCP-195C	Vendor Specific	No		27/06/2019 08:43:07
USB Input Device	HID (Human Interface Device)	No		06/07/2019 08:41:58
Brother DCP-195C	Vendor Specific	No		20/02/2019 01:12:32
USB Mass Storage Device	Mass Storage	No		27/06/2019 08:50:22
USB Printing Support	Printer	No		20/02/2019 01:12:32
USB Printing Support	Printer	No		27/06/2019 08:43:07
USB Input Device	HID (Human Interface Device)	No		06/07/2019 08:42:05
ADATA USB Flash Drive USB Device	Mass Storage	Yes	28B0208250750068	06/07/2019 08:44:18
USB Composite	Unknown	Yes		06/07/2019 08:31:33
Generic Bluetooth Adapter	Bluetooth Device	No		22/03/2019 20:32:40

USB Information

device name	device type	connected	serial number	last connected
USB Composite	Unknown	No		05/07/2019 18:47:11
Lenovo USB Hard Drive USB Device	Mass Storage	No	94CE2A487F664FD888ADF2761F2F6E	09/04/2019 19:26:37
ADATA USB Flash Drive USB Device	Mass Storage	No	11C080255212010	24/06/2019 00:18:34
Generic Flash Disk USB Device	Mass Storage	No	449821DD	04/07/2019 20:46:07
SanDisk Cruzer Blade USB Device	Mass Storage	No	20051535731AFB005D67	23/11/2018 01:11:23
ADATA USB Flash Drive USB Device	Mass Storage	No	28B020829068010B	30/01/2019 07:45:27
USB FLASH DRIVE USB Device	Mass Storage	No	070888DA361C4C9	20/05/2019 21:08:22
USB FLASH DRIVE USB Device	Mass Storage	No	75F3FDEC	17/12/2018 20:43:36
USB Composite	Unknown	No	BROM2F423684	20/02/2019 01:12:32
USB Composite	Unknown	No	BROC3F301492	27/06/2019 08:43:06
USB Composite	Unknown	No		06/07/2019 08:43:04
SanDisk Cruzer Blade USB Device	Mass Storage	No	200443186213F970F232	24/01/2019 19:48:28
Mass Storage Device USB Device	Mass Storage	No	116AC2101219	14/12/2018 00:37:56

Web Browser Forensics

Websites You've Visited

URL	Title	Visit Count	Web browser
www.google.com	Google	11	Firefox
www.deepl.com	DeepL Translator	8	Firefox
contacts.google.com	GoogleContacts	4	Firefox
mail.google.com		2	Firefox
www.facebook.com	Facebook	2	Firefox
www.youtube.com	YouTube	2	Firefox
www.google.com		1	Firefox
www.sospc20.com	Comparatif et test simulateurs rseau	1	Chrome
accounts.google.com	Sign inGoogle Accounts	1	Firefox
feed.sonic-search.com		1	Firefox
forsenergy.com	Understanding Firewall Profiles	1	Firefox
icoconvert.com	ICO ConvertCreate Icons From PNGJPG Images	1	Firefox
myaccount.google.com		1	Firefox
search.yahoo.com	artefacts vs peruveYahoo Search Results Yahoo	1	Firefox

Web Browser Forensics

URL	Title	Visit Count	Web browser
security.stackexchange.c	forensicsWhat is the difference between	1	Firefox
techibee.com	What is domain Public and Private profiles in	1	Firefox
translate.google.dz	Google Translate	1	Firefox
whatis.techtarget.com	What is fulldisk encryption FDEDefinition	1	Firefox
www.europol.europa.eu	Report Cybercrime onlineEuropol	1	Firefox
www.icoconverter.com	Online ICO converter	1	Firefox
www.internet-signalement.gouv.fr		1	Firefox

Data Analysis and Findings

Presenting findings include presenting the pieces of evidence that you found during your examination.

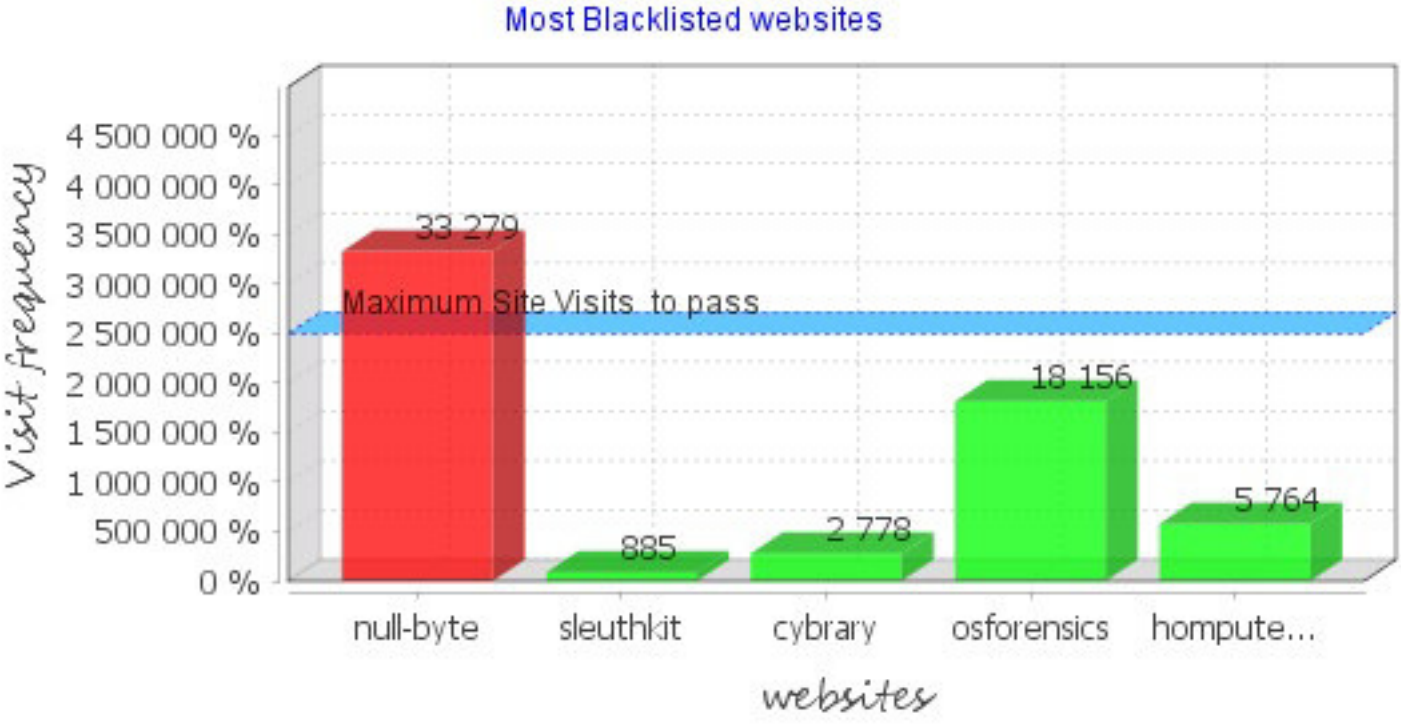
INVESTIGATION REPORT | Printed on the July 13, 2019

Investigating web browsers for forensics artifacts

Web browser

Website	Frequency of Site Visits
null-byte	33279
sleuthkit	885
cybrary	2778
osforensics	18156
homputersecurity	5764

Website Statistics Visitors



Result Of An Investigation

The last piece of information that is commonly present in a forensic report is conclusions drawn by the forensic expert.

INVESTIGATION REPORT Printed on the July 13, 2019

Conclusion

The undersigned inspector certifies:

- Browsed foreign sites on terrorist attacks with high frequency.;

- Retrieve suspect e-mail communications between Alan and John;

Note :

You are advised not to make any decision unless you have clearly understood the observations in this report.

Should you need any further information, please do not hesitate to contact me.

Signature

Références bibliographiques

[Nih19] Nihad A Hassan. Introduction: Understanding digital forensics. In Digital Forensics Basics, pages 1{33. Springer, 2019.

[Käv18] Joakim K¨avrestad. Fundamentals of Digital Forensics: Theory, Methods, and Real-Life Applications. Springer, 2018

[Xia18] Xiaodong Lin. Keyword forensics. In Introductory Computer Forensics, pages 245{255. Springer, 2018

[Mon18] A, M. K., June 2018. *Learning Malware Analysis.Birmingham-Mumbai* : Packt.

[Aym16] Ayman Shaaban and Konstantin Sapronov. *Practical Windows Forensics*. Packt Publishing Ltd, 2016.

[Gra14] Michael W Graves. Digital archaeology: the art and science of digital forensics. Pearson Education, 2013

[Mic14] Michael Hale Ligh, Andrew Case, Jamie Levy, and Aaron Walters. The art of memory forensics: detecting malware and threats in windows, linux, and Mac memory. John Wiley & Sons, 2014

[Lar12] Larry Daniel. *Digital forensics for legal professionals: understanding digital evidence from the warrant to the courtroom*. Elsevier, 2011.

[Esp05] Espen André Fossen and André Årnes. Forensic geolocation of internet addresses using network measurements. Master's thesis, Institutt of Telematics, Norwegian University of Science and Technology, 2005.

[Eog04] Eoghan Casey. Digital evidence and computer crime: Forensic science, computers, and the internet. Academic press, 2011.

[Muk03] Srinivas Mukkamala and Andrew H Sung. Identifying significant features for network forensic analysis using artificial intelligent techniques. International Journal of digital evidence, 1(4):1{17, 2003.

[Ken06] Karen Kent, Suzanne Chevalier, Tim Grance, and Hung Dang. Guide to integrating forensic techniques into incident response. NIST Special Publication, 10(14):800{86, 2006.

[Joh14] Leighton Johnson. Computer incident response and forensics team management: Conducting a successful incident response. Newnes, 2013.

[Sch03] Douglas Schweitzer. Incident response: computer forensics toolkit. Wiley New York, 2003.

[Bri04] Brian Kim Tim Grance, Karen Kent. Computer Security Incident Handling Guide, Recommendations of the National Institute of Standards and Technology. National Institute of standard and technology US department of commerce, 2004.

[Kev03] Chris Prosise, Kevin Mandia, and Matt Pepe. Incident response & computer forensics. 2003.

[Pet11] Peter Hannay and Andrew Woodward. Cold boot memory acquisition: An investigation into memory freezing and data retention claims. In Security and Management, pages 620{622, 2008.

[Pri18] Priya B Gadgil et Sangeeta Nagpure.Hunting advanced volatile threats using memory forensics. International Journal of Advance Research, Ideas and Innovations in Technology,,Volume 4, Issue 4, 2018

[Sam12] Sameer H Mahant and BB Meshram. Ntfs deleted files recovery: Forensics view. IRACST-International Journal of Computer Science and Information Technology & Security (IJCSITS), 2(3), 2012

[Xu and Wang,13] Lijuan Xu and Lianhai Wang. Research on extracting system logged-in password forensically from windows memory image file. In 2013 Ninth International Conference on Computational Intelligence and Security, pages 716{720. IEEE, 2013.

[Marko et al., 14] Steffen Logen, Hans Hofken, and Marko Schuba. Simplifying ram forensics: a gui and extensions for the volatility framework. In 2012 Seventh International Conference on Availability, Reliability and Security, pages 620{624. IEEE, 2012.

[Lei et al.,14] Liu Guangqi, Wang Lianhai, Zhang Shuhui, Xu Shujiang, and Zhang Lei. Memory dump and forensic analysis based on virtual machine. In 2014 IEEE International Conference on Mechatronics and Automation, pages 1773{1777. IEEE, 2014.

[Chang et al., 13] Ya-Ting Chang, Min-Ju Chung, Chin-Feng Lee, Cheng-Ta Huang, and Shiuh-Jeng Wang. Memory forensics for key evidence investigations in case illustrations. In 2013 Eighth Asia Joint Conference on Information Security, pages 96{101. IEEE, 2013.

[Muh18] Muhammad Abulaish and Nur Al Hasan Haldar. Advances in digital forensics frameworks and tools: A comparative insight and ranking. International Journal of Digital Crime and Forensics (IJDCF), 10(2): 95{119, 2018.

[Dan09] Daniel Ayers. A second-generation computer forensic analysis system. Digital investigation, 6:S34 {S42, 2009.

[Vas04] Vassil Roussev and Golden G Richard III. Breaking the performance wall: The case for distributed digital forensics. In Proceedings of the 2004 digital forensics research workshop, volume 94, 2004

[Ted06] Ted Lindsey. Challenges in digital forensics. In The Digital Forensic Research Workshop (DFRWS), New York, 2006.

[Rig06] III Richard and Vassil Roussev. Digital forensic tools: the next generation. In Digital crime and forensic science in cyberspace, pages 75{90. IGI Global, 2006.

[Gab07] Gabriela Limon Garcia. Forensic physical memory analysis: an overview of tools and techniques. In TKK T-110.5290 Seminar on Network Security, volume 207, pages 305{320, 2007.

[Gnr17] Gartner, “Gartner Says Worldwide Information Security Spending Will Grow 7 Percent to Reach $86.4 Billion in 2017,” June 1, 2018. www.gartner.com/newsroom/id/3784965

[Mil04] Matt Miller and Jarkko Turkulainen. \remote library injection.” 2004.

[Suth08] Iain Sutherland, Jon Evans, Theodore Tryfonas, and Andrew Blyth. Acquiring volatile operating system data tools and techniques. *ACM SIGOPS Operating Systems Review*, 42(3):65{73, 2008.

[Rev14] Revue N'tic, juin 2014, Algérie, page 22.

[Knu16] Knutson, T., & Carbone, R. (2016). Filesystem Timestamps: What Makes Them Tick? GIAC GCFA Gold Certification.

[Dha16] Collecting Volatile and Non-volatile Data. 2016. Url: https://www.linkedin.com/pulse/collecting-volatile-non-volatile-data-vuppala-dhanunjaya/. [Accessed 05 /02/ 2019].

[Lord18] What Are Memory Forensics? A Definition of Memory Forensics. 2018. Url: https://www.linkedin.com/pulse/collecting-volatile-non-volatile-data-vuppala-dhanunjaya/ .[Accessed 05 /02/ 2019].

[Zel17] One-Click Windows Memory Acquisition with DumpIt. 2017. Website: https://zeltser.com/memory-acquisition-with-dumpit-for-dfir-2/. [Accessed 05 /02/ 2019].

[Fort17] How to retrieve user's passwords from a Windows memory dump using Volatility. 2017. website : https://www.andreafortuna.org/2017/11/15/how-to-retrieve-users-passwords-from-a-windows-memory-dump-using-volatility/ . [Accessed 05 /02/ 2019].

[Rob16] Msramdmp Project, website: http://www.mcgrewsecurity.com/projects/msramdmp . [Accessed 05 /02/ 2019].

[Vol18] Volatility 2.6 Project. website: https://www.volatilityfoundation.org/26 . [Accessed 05 /02/ 2019].

[Eri09] Eric Zimmermans Tools. website: https://ericzimmerman.github.io/#!index.md . [Accessed 05 /02/ 2019].

[Gek09] What Is a Log File (and How Do I Open One)?, 2018. Url: https:// www.howtogeek.com/359463/what-is-a-log-file/ [Accessed 05 /02/ 2019].

[Mag14] Magnet Forensics (2014), Forensic analysis of LNK files. Available online: https://www.magnetforensics.com/computer-forensic/forensic-analysis-of-lnk-files/, [Accessed 05 /02/ 2019].

[Nir09] Nirsoft project, Url: http://www.nirsoft.net/utils/full_event_log_view.html [Accessed 2019].

[Cso19] Antwanye Ford and LaTia Hutchinson, C. C., January 16, 2018. *Full disk encryption: do we need it?*. [Online] Available at: https://www.csoonline.com/ article/ 3247707/ full-disk-encryption-do-we-need-it.html [Accessed 16/01/2018].

[Apl19] Cookbook, T. D., 2018. The Windows Registry. [En ligne] Available at: https://download.aplteam.com/Cookbook/15-Windows-Registry.html [Accessed 08/2/2019].

[Eye19] Asysko, S., n.d. *videosurveillance.* [Online] Available at: https://www.asysko.com/ en/ solutions-view/videosurveillance/ [Accessed 08/2/2019].

[Cyp19] Tavares, P., 2019. *hashcat-tutorial-for-beginners.* [Online] Available at: https://seguranca-informatica.pt/hashcat-tutorial-for-beginners/#.XRPnwf5S_IV , [Accessed 05/2/ 2019].

[Md5] Blandyuk, 2019. *HashKiller* Project. [Online] Available at: https://hashkiller.co.uk/ , [Accessed 05/02/2019].

[Ran17] Morgan, S., n.d. *Global Ransomware Damage Costs Predicted To Hit $11.5 Billion By 2019.* [Online] Available at: https://cybersecurityventures.com/ransomware-damage-report-2017-part-2/ [Accessed 05/2/ 2019].

[Sek13] sekurlsa, m. @., 2013. *l'essai de récupération des mots de passe des tâches planifiées.* [Online] Available at: http://blog.gentilkiwi.com/securite/mimikatz/sekurlsa-credman [Accessed 05/2/ 2019].

[Microsoft,2019] website: https://docs.microsoft.com/enus/powershell/ scripting/overview?view=powershell-6/, [Accessed 2019].

[Oracle,19] Oracle, C. (2019). Java technology oracle corporation. Website: https://go.java/ index.html?intcmp=gojava-banner-java-com/, [Accessed 05/05/2019].

[Lebigdata,19] Website: https://www.lebigdata.fr/python-langage-definition/ , [Accessed 2019].

[SQLite, 19] Website: http://www.sqlitetutorial.net/what-is-sqlite/ , [Accessed 05/2/ 2019].

[Acc2018] AccessData Corporation. Forensic toolkit Project. [Accessed 22/5/ 2019].

[Cso18] Fruhlinger, J., 2018. *Top cybersecurity facts, figures and statistics for 2018.* [Online] Available at: https://www.csoonline.com/article/3153707/top-cybersecurity-facts-figures-and-statistics.html [Accessed 20 3 2019].

[Der07] Deroko. Ultimate way to hide rootkit. [Accessed 25/2/ 2019].

[Imm19] Canvas Project, website: https://www.immunityinc.com/products/canvas . [Accessed 2019].

[Cor19] Core-impact Project ,website: https://www.coresecurity.com/core-impact. [Accessed 2019].

[Met18] Metasploit Project ,https://www.metasploit.com/,[Accessed 25/2/ 2019].

[Tec19] Techopedia, 2015. *Memory Dump.* [Online] Available at: https:// www. Techopedia .com /definition/20663/memory-dump [Accessed 15/01/2019].

[Ins19] Skills, I., 2018. *File Carving*. [Online] Available at:https://resources.infosecinstitute.com/file-carving/#gref [Accessed 05 05 2019].

[Sof09] Softxpantion. (2009). Metadata in microsoft office and in PDF documents. Available online: https://www.soft-xpansion.eu/files/cc/Metadata.pdf. [Accessed 05/05/2019].

[Eur19] European Union Agency for Law Enforcement Cooperation, 2019, https://www.europol.europa.eu/report-a-crime/report-cybercrime-online, [Accessed 2019].

[Bpd19] Browser Password Dump , http://securityxploded.com/browser-password-dump.php , [Accessed 05 /05/2019].

[Mch15] Web reference of source: http://insidebigdata.com/2015/12 /11/machine-learning-is-cybersecuritys-answer-to-detecting-advancedbreaches/ [Accessed 05 /05/2019].

II.1 Introduction

A l'heure où la sécurisation des données est une des préoccupations principales et où les systèmes informatiques sont de plus en plus piratés, l'informatique légale ou « computer forensic » apparait comme une méthode d'investigation permettant d'identifier les traces laissées par une intrusion et de rassembler les preuves numériques.

C'est dans un contexte de « cybercriminalité » ascendante, que ces méthodes apportent un appui considérable à la lutte contre le cyber-crime.

Ce chapitre fournit une compréhension fondamentale de l'investigation numérique et de la cybercriminalité. Il va définit les termes de base, et couvert le concept de preuve numérique, ses différents types, et où il peut être trouvé dans les dispositifs électroniques.

II.2 Investigation Digital

II.2.1 Définition

L'investigation numérique est une branche de la science médico-légale qui utilise les connaissances scientifiques pour recueillir, analyser, documenter et présenter des preuves numériques liées à la criminalité informatique en vue de leur utilisation devant un tribunal [Nih19]. Le but ultime est de savoir ce qui a été fait, quand et qui l'a fait.

Le terme "digital forensics" est largement utilisé comme synonyme d'ordinateur forensics (également connu sous le nom de cyber-forensics), mais c'est étendu pour couvrir tous les appareils qui sont capables de stocker des données numériques, comme les appareils de réseau, téléphones portables, caméras numériques, Internet des objets, appareils domestiques numériques et autres supports de stockage numériques.

L'investigation numérique est considérée comme une branche relativement nouvelle dans le domaine de la cyber-sécurité, qui prend de plus en plus d'importance avec la prolifération des crimes et des activités illégales dans le cyberespace [Nih19].

II.2.2 Catégories d'investigation numérique

Pour traiter les types d'incidents et pour faciliter la détection et la correction des crimes, le domaine de l'investigation numérique peut être divisé en trois catégories principales :

L'investigation sur internet :

L'investigation sur internet couvre l'enquête sur les activités criminelles qui se produisent sur l'Internet. Elle concerne l'analyse des origines, de contenu, des modèles et des chemins de transmission des e-mails et des pages Web ainsi que l'historique du navigateur, les scripts du serveur Web et les messages d'en-tête [Esp05].

L'investigation sur réseau :

L'investigation sur réseau traite de la capture et de l'inspection des paquets transitant par un nœud sélectionné du réseau afin d'identifier les attaques. Les réseaux contiennent des preuves numériques qui peuvent être utilisées pour établir qu'un crime a été commis, déterminer comment un crime a été commis, fournir des renseignements d'enquête, révéler les liens entre un criminel et une victime et réfuter ou appuyer les déclarations des témoins. Il peut être collecté au niveau de serveur, de proxy ou à partir de plusieurs autres sources telles que routeurs, commutateurs, IDS et pare-feu [Muk03] et [Eog04].

L'investigation sur ordinateur

L'investigation sur ordinateur est le plus ancien type d'enquête numérique, il est concerné par l'investigation des preuves numériques trouvées sur les ordinateurs de bureau, les portables, les périphériques de stockage numérique (comme les disques durs externes, clés USB et cartes SD), et dans la mémoire vive, en plus des systèmes d'exploitation et des applications installées traces et leurs logs associés [Eog04].

L'activité principale de ce type consiste à récupérer les données supprimées de la mémoire du périphérique cible et à les analyser pour rechercher des preuves incriminantes ou exonérâtes.

II.2.3 Objectifs de l'investigation numérique

L'objectif principal de l'investigation numérique est investiguer les crimes commis à partir de dispositifs informatiques comme les ordinateurs, les tablettes, les téléphones cellulaires ou tout autre dispositif qui peut stocker et traiter et en extraire des preuves numériques d'une manière légale et judiciaire pour être présentées devant un tribunal [Nih19].

L'investigation numérique réalise ceci dans les manières suivantes :

- Trouvé des preuves juridiques dans les dispositifs numériques et préserver leur intégrité d'une manière jugée admissible devant un tribunal.
- La préservation et la récupération des preuves suivant les procédures techniques acceptées par le tribunal.
- Identifier les fuites de données dans une organisation.
- Accès aux risques possibles survenant lors d'une violation de données.
- Présenter les résultats dans un rapport officiel pouvant être présenté au tribunal.
- Fournir un guide pour le témoignage d'experts devant les tribunaux.

II.2.4 Le processus investigation numérique

Selon l'Institut national des normes et de la technologie (NIST), le processus de réalisation d'une investigation judiciaire numérique comporte quatre phases [Ken06] : collecte, examen, analyse et documentation, comme illustré dans la Figure 1.

1) La première phase est la collecte des preuves, où l'équipe doit identifier, obtenir, étiqueter et enregistrer les données provenant de toutes les sources possibles, sans négliger des procédures qui préservent leur intégrité.

2) Le deuxième phase concerne l'examen des données, où commence le processus judiciaire des données collectées, en utilisant une combinaison de méthodes manuelles et automatisées, et évalue ainsi que extrait des données présentant un intérêt particulier, tout en préservant leur intégrité.

3) La troisième phase concerne l'analyse des résultats de l'examen avec des méthodes et techniques légalement justifiées, afin d'obtenir des informations sur d'investigation.

4) la dernière phase est la présentation des résultats de l'analyse, qui peut inclure les méthodes utilisées, les outils et les procédures choisis, en déterminant les autres actions qui doivent être menées et les propositions nécessaires pour améliorer chaque aspect du processus forensique.

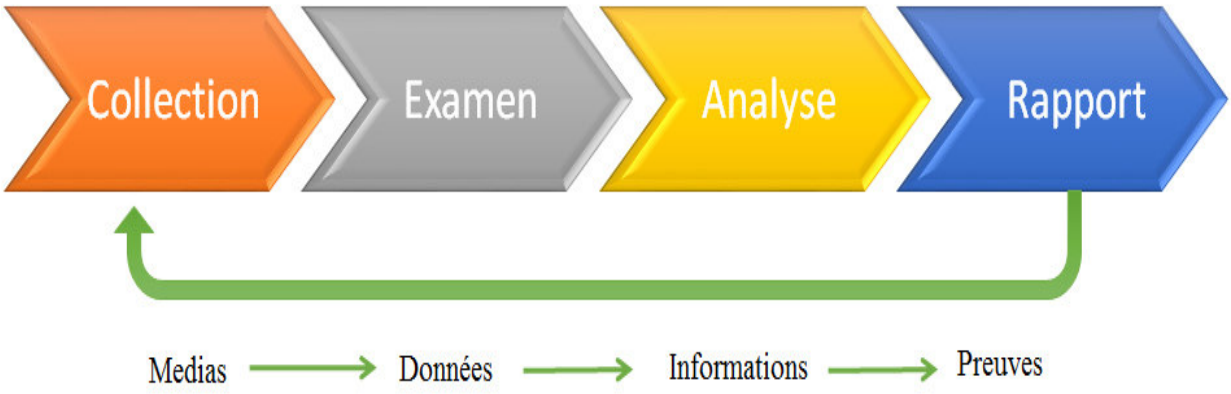


Figure 1. Processus de Investigation numérique.

Si ces procédures sont appliquées de manière incorrecte, on risque d'endommager ou de détruire les preuves numériques, ce qui les rend inadmissibles devant un tribunal. Certains détails de ces étapes peuvent différer, dépendant des besoins spécifiques en criminalistique.

Les quatre principales sources de données, à l'intérieur de tout réseau ou ordinateur où se déroule le processus d'investigation, sont les fichiers, les systèmes d'exploitation, le trafic réseau et les applications. [Joh14]

II.3 Cybercriminalité

En bref, la cybercriminalité comprend toute activité illégale commise en utilisant un type de dispositif ou de réseau informatique tel que l'Internet. Le Département de la Justice des États-Unis (DOJ) définit la cybercriminalité comme « toute infraction criminelle commise contre ou avec l'utilisation d'un ordinateur ou d'un réseau informatique. » [Nih19].

La principale motivation de la cybercriminalité est le gain financier (exemple : la diffusion de logiciels malveillants pour voler les codes d'accès aux comptes bancaires). Cependant, une grande partie de la cybercriminalité a des motivations différentes, comme l'interruption de service (par exemple, les attaques DDoS pour arrêter les services offerts par l'organisation cible), le vol de données confidentielles (par exemple : données consommateurs, informations médicales), l'échange illégal de matériel protégé par le droit d'auteur, et le cyber-espionnage (secrets commerciaux et militaires).

II.3.1 Types d'attaques de cybercriminalité

Attaques internes: c'est le cyber-risque le plus dangereux auquel les organisations sont confrontées aujourd'hui, car il peut durer très longtemps sans qu'elles ne le sachent ,de telles attaques surviennent lorsqu'il y a abus de confiance de la part de l'employé ou d'autres personnes, comme d'anciens employés ou des associés travaillant au sein de l'organisation cible, qui ont un accès légitime à ces systèmes informatiques ou aux informations concernant ses pratiques et moyens de sécurité informatique [Nih19], Par exemple : l'espionnage économique appartient à cette catégorie.

Attaques externes: ce type d'attaque provient de l'extérieur de l'organisation cible, venant généralement de pirates informatiques qualifiés. De telles attaques constituent les plus grandes attaques contre des organisations à travers le monde [Nih19], Un pirate informatique peut essayer de pénétrer dans les réseaux informatiques de l'organisation cible à partir d'un autre pays pour obtenir un accès non autorisé. Parfois, des attaquants externes obtiennent des informations de l'intérieur (employé mécontent) de l'entreprise cible, qui dispose d'informations sur ses systèmes de sécurité pour faciliter leur accès illégal.

II.3.2 Comment les ordinateurs sont-ils utilisés dans les cyber-crimes ?

La cybercriminalité peut être divisée en trois catégories principales en ce qui concerne la manière dont l'ordinateur a été utilisé pour commettre un crime.

Lorsqu'un crime numérique se produit, selon la nature de l'incident, le rôle de l'ordinateur peut être l'un des suivants [Sch03] :

1. **L'ordinateur en tant qu'objet du crime :** Lorsqu'un ordinateur est affecté par l'acte criminel, il fait l'objet du crime. Par exemple, lorsqu'un ordinateur est volé ou détruit.
2. **L'ordinateur comme sujet du crime :** Lorsqu'un ordinateur est l'environnement dans lequel le crime est commis, il est le sujet du crime. Par exemple, lorsqu'un ordinateur est infecté par un virus ou un code malveillant.
3. **L'ordinateur comme assistant dans le crime :** Dans ce cas, l'ordinateur peut être utilisé comme outil pour mener ou planifier un crime. Par exemple, lorsqu'un ordinateur est utilisé pour falsifier des documents ou s'introduire dans d'autres ordinateurs, c'est lui qui contribue au crime.

II.4 Les Objets d'investigation

Ce chapitre est écrit sur la base de la version 1709 de Windows 10 il peut exister des différences de fonctionnement des artefacts dans les versions antérieures et ultérieures de Windows. Il décrit et explique les artefacts juridiques les plus importants, qui doivent être bien connus de tout expert en criminalistique numérique.

II.4.1 Métadonnées

Les métadonnées sont l'une des plus importantes sources d'information criminalistique. On peut utiliser les métadonnées pour suivre les horodatages, l'emplacement des données, le réglage de l'exposition sur une image numérique qui permettent à un utilisateur ou au système de localiser, trier et assembler des données.

Le but des métadonnées est de stocker des informations sur d'autres données. Cela peut aider à l'organisation et à la récupération des données. Dans les mains d'un expert en criminalistique numérique qualifié, les métadonnées peuvent éclairer un problème particulier dans une affaire, ou bien constituer un tournant. Par exemple, on peut éditer les informations de métadonnées dans le cas de documents Office 2019, en cliquant simplement avec le bouton droit de la souris sur un fichier ➤ Info ➤ Properties ➤ Show All Properties (voir Figure 2).

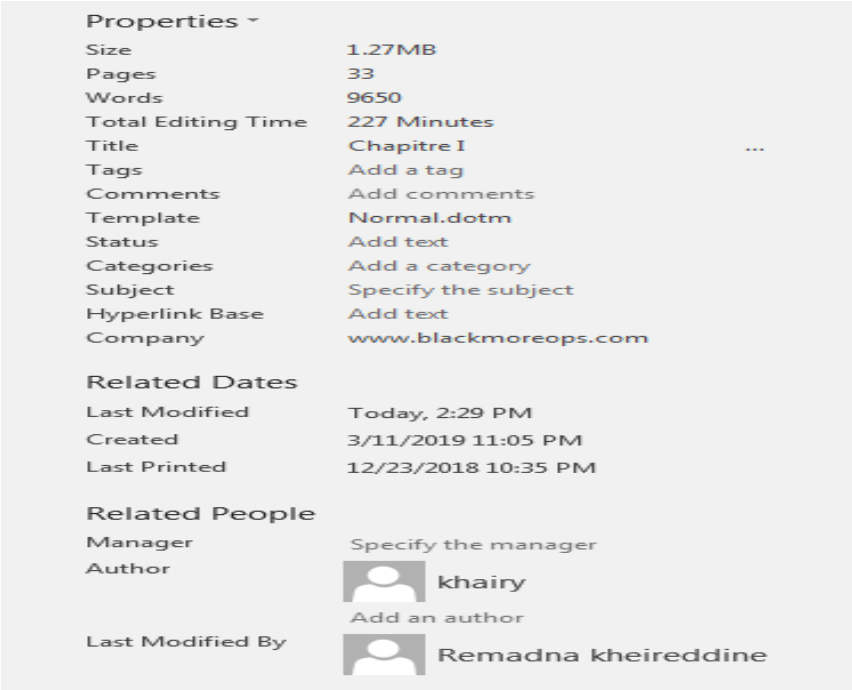


Figure 2. Affichage des propriétés d'un fichier MS Word.

II.4.2 Fichiers. Link

Les fichiers LNK constituent un artefact relativement simple mais précieux pour l'enquêteur en criminalistique. En substance, les fichiers.lnk sont des raccourcis dans Windows [Mag14], Ce qui rend les fichiers.lnk très intéressants pour les experts judiciaires, C'est qu'ils ne sont pas supprimés quand on retire un lecteur distant contenant le fichier cible, ou quand un fichier est supprimé.

Les fichiers LNK ont une valeur légale, parce qu'ils contiennent les informations suivantes :

- L'emplacement du fichier cible.
- Heure de dernière mise à jour du fichier cible
- Informations sur le périphérique où le fichier cible est stocké.
  - S'il s'agit d'un dispositif local, il inclura le numéro de série et le type de volume.
  - Si le périphérique était un dispositif distant, il inclura le nom du partage.

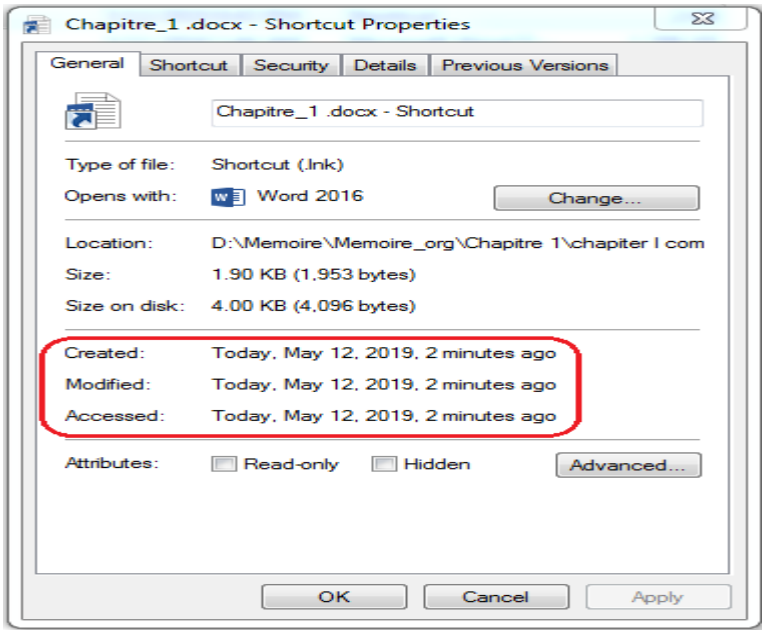


Figure 3. Affichage des propriétés du fichier LNK.

Comme la plupart des données contenues dans les fichiers.lnk sont dans un format difficile à interpréter manuellement, il est pratique d'utiliser un outil pour analyser les fichiers.lnk. Plusieurs logiciels criminalistique prennent en charge l'interprétation et la présentation des données de fichiers.lnk dans un format ordonné. Il existe plusieurs outils spécialement conçus pour l'analyse. Fichiers LK, y compris LECmd de Erik Zimmerman [Eri09].

II.4.3 Fichiers Log

LOG est l'extension de fichier d'un fichier généré automatiquement contenant un enregistrement des événements de certains logiciels et systèmes d'exploitation [Gek09].

Un fichier journal est essentiellement une liste de la façon dont une application s'est comportée et pour aider au dépannage. Cependant, d'un point de vue la criminalistique, les fichiers journaux peuvent fournir une mine d'informations.

L'important d'un fichier journal est de garder une trace de ce qui se passe dans les coulisses et si quelque chose devait arriver dans un système complexe, on avait accès à une liste détaillée des événements survenus avant le dysfonctionnement.

Il existe cinq types d'événements pouvant être enregistrés du log des événements Windows:

1. Erreur: Indique qu'un problème important s'est produit : par exemple, lorsqu'un service ne se charge pas au démarrage.
2. Attention: Ce n'est pas un événement important, mais pourrait entraîner de graves problèmes à l'avenir.
3. Information: Indique le succès du fonctionnement d'un objet comme un service, une application ou un pilote.
4. Audit du succès : Indique un événement de sécurité réussi (par exemple, une connexion réussie est enregistrée comme "Success Audit event").
5. Audit d'échec: Annule un audit réussi (par exemple, lorsqu'un utilisateur ne parvient pas à se connecter à Windows, l'événement est enregistré en tant qu'événement "Failure Audit").

Pour Windows 10/8/8/7/Vista, il existe un outil simple appelé FullEventLogView qui affiche dans un tableau les détails de tous les événements dans les journaux d'événements Windows, y compris la description des événements [Nir09].

II.4.4 Flash Disk

Dans l'utilisation moderne des ordinateurs, il est extrêmement courant d'utiliser des périphériques USB pour stocker des données. Il peut être très important d'être capable de détecter et d'examiner comment des périphériques USB ont été connectés à un ordinateur lors d'un examen légiste.

Heureusement, Windows suivra de nombreux événements liés aux périphériques USB connectés et déconnectés d'un ordinateur. Pour identifier les périphériques USB connectés à un ordinateur sous Windows, il impliquerait de combiner des informations provenant de trois sources: setupapi.upgrade.log, le registre et les logs du système [Käv18].

Setupapi.dev.log est un fichier journal se trouvant dans le répertoire[SystemRoot]\Windows\INF, Il enregistre les événements liés à l'installation et la désinstallation des périphériques et lorsqu'un périphérique USB est inséré, il est installé avant utilisation, ce qui fait de ce fichier journal une source d'information [Käv18].

La figure 4 illustre le début d'une entrée relative à une clé USB insérée.

```
>>>  [Device Install (Hardware initiated) - USBSTOR
\Disk&Ven_Kingston&Prod_DataTraveler_3.0&Rev_PMAP\08606E694934BEA1B7057E9D&0]
>>>  Section start 2018/03/27 11:07:13.879
     ump: Creating Install Process: DrvInst.exe 11:07:13.882
     ndv: Retrieving device info...
     ndv: Setting device parameters...
     ndv: Searching just Driver Store...
     dvi: {Build Driver List} 11:07:13.902
```

Figure 4. Entrée USB dans setupapi.dev.log [Käv18].

Comme le montre la figure 4.10, la première ligne indique qu'un périphérique est en cours d'installation et on pourrait utiliser le terme "USBSTOR\Disk&Ven" pour rechercher des événements relatifs aux périphériques de stockage USB.

Dans le but de suivre le périphérique USB, l'information la plus importante révélée est le numéro de série du périphérique, qui se trouve à la fin de la première ligne. Dans ce cas, il s'agit du 08606E694934BEA1B7057E9D & 0. Les mêmes informations peuvent être trouvées dans les journaux du système Windows, avec l'événement ID 20001, comme illustré à la Fig. 4.1.

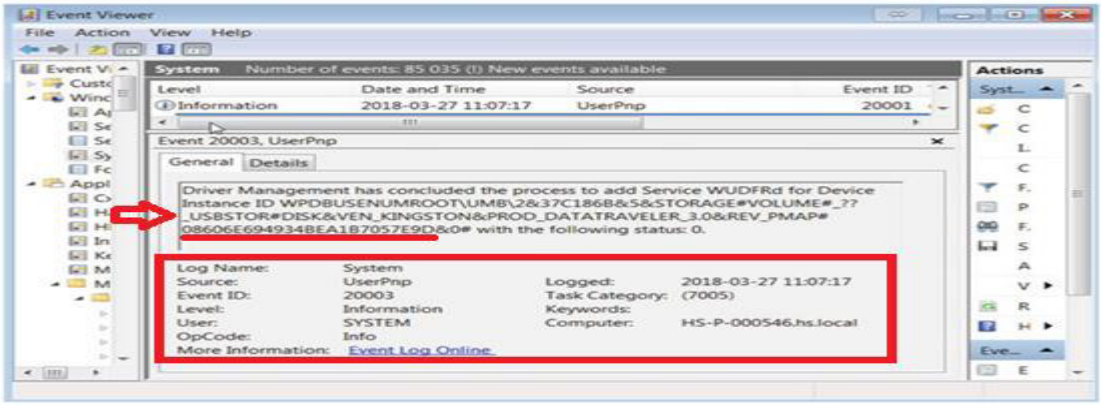


Figure 4.1. Installation USB dans l'observateur d'événements [Käv18].

Après avoir rassemblé les informations des journaux décrits ci-dessus, des informations supplémentaires peuvent être trouvées dans le registre Windows. L'utilisation de clés USB laissera plusieurs traces dans le registre, toutes dans la registre de système, Le premier point d'intérêt concerne les sous-clés de HKEY_LOCAL_MACHINE \ SYSTEM \ CurrentControlSet \ Enum \ USBSTOR et on y trouvera une clé de registre avec le nom "USBSTOR", Cette clé contient des sous-clés pour différentes combinaisons de noms de fournisseurs et de produits, qui à leur tour contiennent des sous-clés pour les différents périphériques USB connectés [Käv18] . Les sous-clés auront un numéro de série pour le périphérique comme nom. Ces informations peuvent servir de chemin de confirmation ou de chemin alternatif pour révéler les informations que nous venons de trouver à l'aide de journaux.

La prochaine étape consiste à déterminer le point de montage utilisé pour le lecteur USB. Le premier endroit à regarder est la clé de registre appelée MountedDevices. Comme le montre la Fig. 4.2, cela montre en fait que le lecteur USB que nous suivons a été monté pour la dernière fois en tant que lettre de lecteur E:

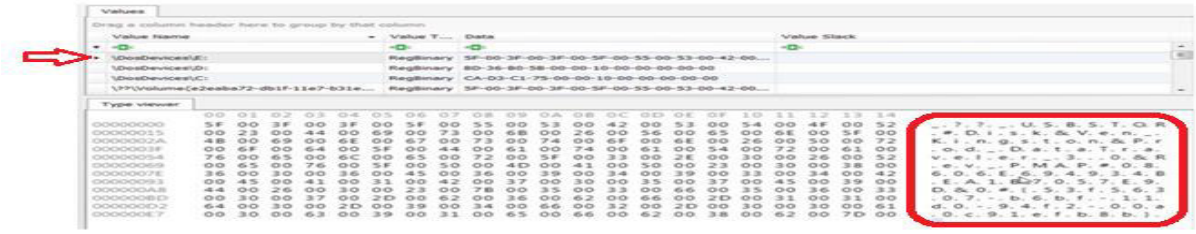


Figure 4.2. Clé de registre MountedDevices [Käv18].

En conclusion, lorsqu'on trouvé une clé USB, il est recommandé d'entrer le numéro de série de cet appareil afin de pouvoir relier cette clé à l'ordinateur examiné.

II.5 Recherche de preuves

La première étape de tout examen criminalistique consiste à collecter des preuves. Pour les débutants, il est nécessaire de discuter de ce qu'est la preuve, ou plus précisément de ce qu'est la preuve numérique.

Une preuve numérique signifie « Les données collectées à partir de tout type de stockage numérique soumis à un examen médico-légal par ordinateur » [Lar12]. Le point essentiel de cette définition est que tout ce qui achemine de l'information numérique peut faire l'objet d'une enquête, et tout support de ce genre qui est ciblé pour examen devrait être traité comme une preuve.

Le processus de collecte de preuves peut être divisé en deux catégories :

II.5.1 Cas l'appareil est éteint

Lorsque l'appareil est éteint, il n'y a pas grand-chose que on puisse faire. On ne peut examiner que les données stockées dans la mémoire statique, telle qu'un disque dur. Lors de la réalisation d'un examen légiste, en particulier dans l'application de la loi, des mesures doivent être prises pour éliminer toute possibilité de modifier les preuves. En fait, chaque action que tu prends va modifier, d'une manière ou d'une autre, les données originales et donc corrompre la preuve. Les preuves corrompues ne seront pas défendables devant les tribunaux.

Pour cette raison, on doit trouver un moyen de faire une copie de la preuve et de procéder à l'examen de la copie. La règle de base est de toujours documenter, en détail, ce que on fait pour vivre des preuves [Käv18] .

II.5.2 Cas l'appareil est allumé

Lors de l'examen d'un ordinateur ou d'un périphérique allumé (examen en direct), l'examinateur a la possibilité de collecter des données volatiles contenant des informations sur l'état actuel du périphérique, Il donne également à l'examinateur l'occasion d'examiner les disques durs actifs pour vérifier s'ils sont cryptés et de collecter des données non cryptées de leur part.

Les applications courantes de chiffrement complet du disque dur (FDE) garantissent que toutes les données sur le disque dur sont chiffrées lorsque l'ordinateur est éteint. Cependant, les données seront déchiffrées lorsque l'ordinateur est allumé [Cso19].

Donc, avant d'éteindre un ordinateur soumis à examen, l'examinateur doit effectuer une recherche approfondie des outils de cryptage. Si un signe de cryptage est présent, L'examinateur doit créer une image logique des disques durs pour s'assurer que les données sont préservées et disponibles pour une analyse ultérieure.

L'objectif ultime d'une enquête en direct est de préserver autant de données volatiles que possible et de s'assurer que les données reposant sur les disques durs sont disponibles pour une analyse ultérieure. On a expliqué un processus pour une investigation directe qui décrit l'ensemble du processus.

Comme le décrit la figure 5 on peut résumer le processus en trois étapes principales:

- Préparation
- Réalisation
- Réflexions sur le passé

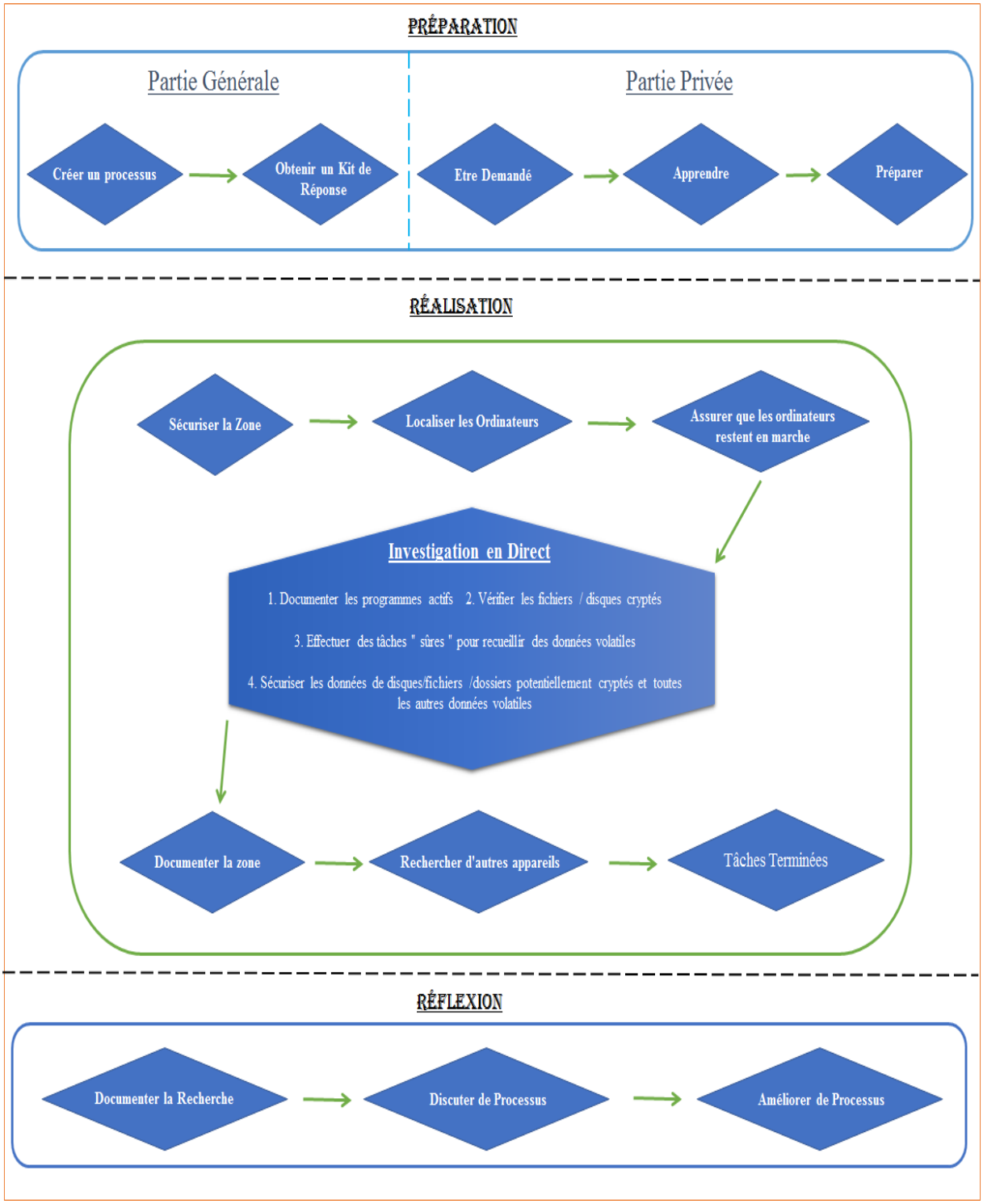


Figure 5. Processus d'investigation en direct [Käv18].

II.5.2.1 L'investigation en direct: Préparation

L'étape de préparation est divisée en deux parties, une partie générale (indiquée par les cases supérieures inclinées) et une partie privée à effectuer pour chaque recherche de maison.

- L'étape générale est divisée en deux phases : la création d'un processus et d'un kit de réponse.

L'idée est que on devrait considérer comment on veut effectuer des investigations en direct de manière générale.

1. La première étape consiste à créer un processus. On peut dire que "créer un processus", c'est mettre des mots et décider comment exécuter le reste des tâches du processus.
2. La prochaine étape consiste à assembler un kit de réponse. Un ensemble de réponse est un kit qui contient les logiciels et le matériel dont on a besoin pour effectuer des enquêtes en direct.

- Selon le cas de la recherche d'une maison, on peut s'attendre à des choses très différentes. Par exemple, si on travaille comme enquêteur privé envoyé pour examiner l'ordinateur d'un assistant économique soupçonné d'avoir volé les registres de clients d'une entreprise, on peut chercher des informations sur les clefs USB.

Lorsque on avait pris connaissance de l'affaire, on doit faire les préparatifs nécessaires pour l'enquête en cours.

II.5.2.2 L'investigation en direct: Réalisation

L'étape de réalisation comprend toutes les tâches qui sont effectuées « sur place ». Dans cette étape, il est important de mentionner deux choses :

1. Selon si on travaille dans la législation ou dans un environnement d'entreprise, on aura des règles et des règlements différents qui restreignent la façon dont on peut travailler. Dans un environnement criminel, on est limité par la loi, et la loi est différente d'un pays à l'autre.
2. Selon le type d'enquête et lorsque on est appelé à la scène, toutes les étapes peuvent ne pas s'appliquer.

On Suppose que le suspect déclare qu'un ordinateur que on avait déterminé comme faisant partie d'une escroquerie liée à la drogue n'a jamais été connecté à Internet.

Dans ce cas, avant de commencer toute recherche de maison, toute l'équipe visitera le site et entrera dans le bâtiment concerné. Les premières tâches consistent à sécuriser le bâtiment, de localiser tous les ordinateurs qui fonctionnent et de s'assurer qu'ils restent en marche. Selon le cas et le contexte, cela peut se faire de diverses façons. La sécurisation de la zone est prioritaire dans cette étape, et c'est le devoir des agents de police.

L'examinateur légiste sur ordinateur peut entrer dans le bâtiment après qu'il est sécurisé et commencer à travailler sur tous les ordinateurs qui sont allumés. On peut déterminer si on doit mener les investigations en direct ou documenter d'abord la zone. En général, la tâche de documenter la zone pourrait être confiée à un agent de police.

En regardant le processus principal dans cette étape de l'investigation, Les buts ultimes de l'investigation en direct devraient être les suivants :

1- Documenter ce qui est visuellement présent à l'écran.
2- Collecter des données volatiles.
3- Vérifier que les données sont cryptées et sécurisées dans le stockage crypté.
4- Fournir un guide pour la recherche continue d'une maison.

II.5.2.3 Enquête en direct: Réflexions sur le passé

La première chose à faire est de rédiger un protocole qui décrit ce qui a été fait et tous les conclusions possibles pendant l'investigation direct. La tâche de documentation peut varier d'une législation à l'autre, mais vous devriez à tout le moins documenter ce que vous avez fait pendant les enquêtes en cours. Idéalement, ce processus sera mené avec des policiers et d'autres experts judiciaires.

II.6 Analyse des données et rapport d'investigation

On discutera de la façon d'analyser les données et de rédiger des rapports, c'est l'étape où on travaille avec les preuves que on avait correctement collecté à l'étape précédente, Cela inclut par exemple des images de disque judiciaires et des décharges de mémoire.

Pour commencer cette discussion, j'aimerais dire que le reste du chapitre est rédigé du point de vue de l'application de la loi.

II.6.1 Réglage de la scène

Une règle générale dans les enquêtes criminelles est que toute personne est innocente jusqu'à preuve du contraire et que les enquêtes ne doivent pas viser à accuser une personne en particulier mais à découvrir la vérité.

On doit souligner que lors de l'analyse et de la rédaction du rapport, l'expert légiste doit s'assurer que son travail répond aux exigences suivantes :

L'impartialité signifie que les preuves incriminantes et exonérées sont considérées et prises en compte. L'essentiel, c'est qu'il n'est pas important que l'examinateur légiste trouve des preuves incriminantes [Käv18], Il doit plutôt être important de trouver une réponse correcte et objective.

Reproductible signifie que la base de vos conclusions est bien documentée pour que quelqu'un d'autre puisse reproduire votre analyse. L'idée générale est que si quelqu'un fait la même chose, il obtiendra la même conclusion. [Käv18].

Par exemple, si on a demandé à l'utilisateur de vérifier si un ordinateur était contrôlé à distance et de conclure qu'il n'avait pas trouvé de telles preuves, il doit documenter comment il a recherché le logiciel de contrôle à distance, analysé les journaux du pare-feu, etc.

II.6.2 Analyse criminalistique

Discussion qu'on vient d'avoir nous amène des limites et des exigences de l'analyse judiciaire, il est temps de passer à cette étape. Qu'une analyse judiciaire consiste essentiellement à répondre aux questions de l'investigation [Käv18].

Pour répondre à la question, on doit analyser les données trouvées sur les images légales que on avait créées pendant la phase de " collecte de preuves ". Il faut savoir aussi qu'il est courant d'inclure des renseignements provenant d'autres sources, comme des interrogatoires ou tout ce qui semble raisonnable dans votre cas. En tant qu'examinateur judiciaire, votre tâche consiste à cerner les problèmes sous-jacents et à en déduire des conclusions.

Il y a deux façons de résoudre ce problème :

1. Je pense qu'il y a certaines informations de base que on devrait toujours découvrir à partir de tous les ordinateurs que on étudie. De plus, Les informations qui devraient être incluses dans cet ensemble de base varieraient d'une législation à l'autre, ainsi que d'un ministère à l'autre.
2. La deuxième partie des "autres" informations serait des informations qui peuvent être liées à des crimes mais pas directement à la question à laquelle on êtes prêt à répondre. La manière dont on devrait traiter ce type d'informations dépend fortement de votre législation et, parfois, des conditions du mandat de perquisition.

La discussion que on vient d'avoir on amène au processus d'une analyse criminalistique. On pourrait dire simplement que on doit poser des questions et trouver des données qui on aident à y répondre, mais on suggèrerait d'adopter une approche plus structurée.

Dans la figure 6, on a essayé de donner un bref aperçu de la façon dont un processus de criminalistique judiciaire pourrait être structuré.

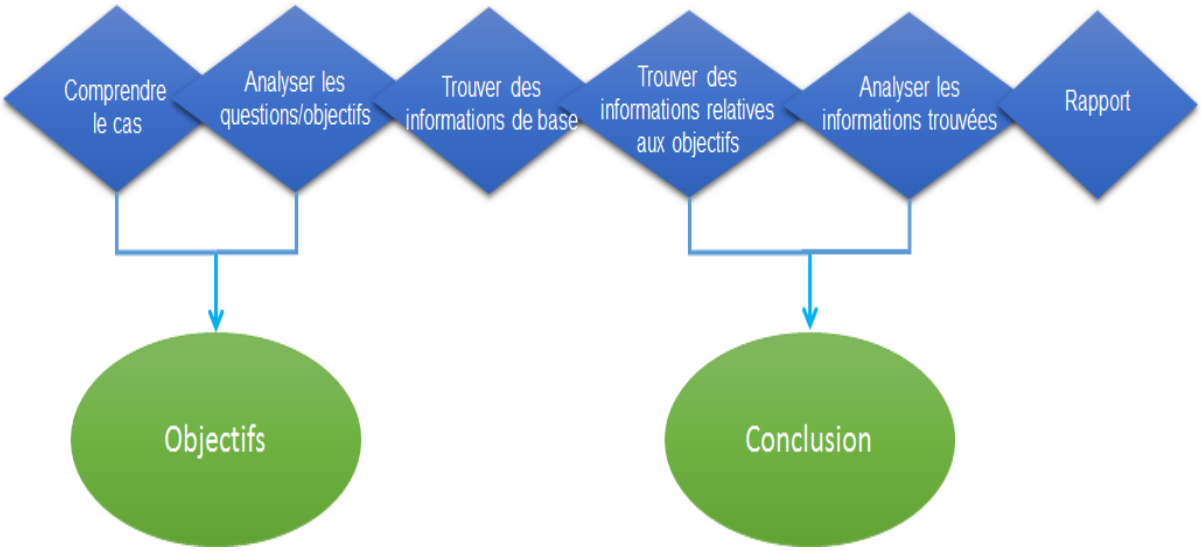


Figure 6. Survol d'un processus criminalistique [Käv18].

1. La première étape serait d'obtenir une compréhension de l'affaire. Cette étape comprendrait généralement de lire comment l'enquête a commencé, des types de preuves qui se trouvent dans l'affaire et des interrogatoires avec les suspects et témoins dans l'affaire.

2. Le deuxième point que on devrait analyser, ce sont les questions de l'enquêteur. Cela permet de déterminer l'objectif réel de notre analyse. Dans cette étape, l'examinateur judiciaire doit utiliser sa connaissance de l'affaire et étudier les questions/objectifs de l'enquête pour définir ses propres objectifs.

3. Le troisième point est on devrait commencer de travailler sur les objectifs définis précédemment, on va chercher des preuves qui sont avec ou contre nos objectifs, Notre objectif n'est pas de forcer une personne de commettre un crime, mais de donner une vision juste de ce qui s'est passé.

4. Le quatrième point est que on a terminé notre enquête et utilisé différentes méthodes pour chercher des preuves sur l'ordinateur, qu'il est le temps d'analyser l'information que on avait trouvé et de déduire des conclusions. Ce qui est remarquable dans cette étape, est que nos conclusions reflètent la façon dont on interprète les preuves.

5. Le cinquième point est que on a terminé notre analyse, qu'il est temps de rapporter nos résultats dans un document bien structuré.

II.6.3 Rapports

La dernière étape d'une analyse criminalistique consiste à rédiger un rapport. Le rapport a essentiellement deux objectifs :

1. C'est ici qu'on présente nos résultats objectifs et que on peut ensuite inclure nos conclusions en se basant sur ces derniers.
2. Le contenu d'un rapport varie selon la législation et les politiques du pays.

Cependant, il est courant que tous les rapports incluent:

1) Données du cas.
2) But de l'examen.
3) Résultats.
4) Conclusions.

II.7 Collection de données

La collecte de données provenant de différentes sources est abordée dans ce section. On peut vouloir collecter des données de différentes sources telles que les disques durs, le registre de Windows, et les sources volatiles.

II.7.1 Imagerie

L'imagerie est le processus de copie d'un disque dur ou d'autres supports de stockage secondaire dans une image légiste qui peut être utilisé pour l'examen criminalistique [Käv18].

Un aspect important d'un examen criminalistique est de s'assurer que les données sur le disque dur qui doit être examiné ne sont pas altérées, et la seule méthode qui permet de réaliser un examen complet consiste à en créer une image légiste, puis de l'examiner. Le meilleur et le plus efficace pour réaliser une image judiciaire est de créer une image physique d'un disque dur.

II.7.2 Collecte des Dumps de mémoire

Un dump mémoire est un processus qui permet de consulter et de stocker le contenu de la mémoire en cas de crash du système ou d'une application [Tec19].

Le dump mémoire assiste les développeurs et les administrateurs système pour diagnostiquer, identifier et résoudre le problème qui a conduit à la défaillance de l'application ou du système.

II.7.3 Collecte des données du Registre

Le Registre Windows est une base de données hiérarchique qui stocke les paramètres de bas niveau pour le système d'exploitation Microsoft Windows et des applications qui y ont accès. Le noyau, les pilotes de périphérique, les services, le gestionnaire de comptes de sécurité et l'interface utilisateur peuvent tous utiliser le registre [Apl19].

Par exemple, si un programme a été installé, on ajoute une nouvelle sous-clé contenant des paramètres tels que le chemin et la version du programme, et le mode d'exécution, dans le Registre Windows.

II.7.4 Collecte de vidéo surveillance

Un système de vidéo-surveillance numérique est un système capable de capturer des images et des vidéos qui peuvent être compressées, stockées ou envoyées sur des réseaux de communication [Eye19].

Si les caméras détectent une intrusion ou un mouvement suspect dans la zone surveillée, une alerte est immédiatement envoyée par téléphone, SMS ou e-mail à la personne concernée : propriétaire, société de sécurité, société de vidéo-surveillance, société de sécurité, … etc.

Les caméras ont plusieurs fonctions :

- Détecter les intrus
- Prévenir et alerter
- Enregistrer les images (ce qui permet d'identifier les intrus).

II.7.5 Processus d'un examen en direct

Le processus d'un examen en direct est souvent plus exhaustif que les processus d'imagerie, collecte de mémoire et collecte de registre. Au minimum, on souhaite également collecter des informations sur les paramètres de temps, l'utilisateur actif, les périphériques connectés via USB, ou le réseau et documenter les programmes actifs [Käv18].

On doit aussi se rappeler quelques considérations intéressantes :

1. On essaie de capturer le plus possible d'informations fiables, ce qui implique de décharger la mémoire et les images sans altérer le contenu dans les meilleurs délais.
2. On veut capturer ce qui apparaît sur l'écran dans les meilleurs délais pour éviter de le perdre.
3. On veut capturer des informations sur les autres appareils intéressants en temps utile.
4. On veut s'assurer que l'ordinateur ne va pas mettre en mode veille ou ne sera pas manipulé à distance.

L’essentiel est que le disque dur doit être imagé s'il y a des soupçons que des logiciels de cryptage fonctionnent sur l'ordinateur et que les données actuellement lisibles peuvent passer dans un état crypté lorsque l'ordinateur est éteint.

II.8 Cracking des mots de passe

Dans la cryptanalyse et la sécurité informatique, le craquage de mots de passe est le processus de récupération de ces derniers à partir de données qui qui ont été enregistrées ou transmises par un système informatique.

Dans cette section, on décrit comment le craquage des mots de passe se fait en utilisant AccessData Password Recovery ToolKit (PRTK) et l'alternative open-source Hashcat.

II.8.1 Craquage des mots de passe en utilisant PRTK

PRTK est un outil capable de craquer un grand nombre de types des fichiers et systèmes de cryptage différents. La partie la plus importante du craquage des mots de passe est le moteur de développement des dictionnaires [Käv18].

Lorsque on trouve un fichier crypté avec un mot de passe que on doit craquer, on peut résumer la procédure comme suit:

1. Créer des dictionnaires.
2. Créer un profil d'attaque.
3. Lancer l'attaque.

II.8.2 Craquage des mots de passe en utilisant Hashcat

Hashcat est un craqueur de mots de passe bien connu, il est conçu pour casser même les mots de passe les plus complexes [Cyp19]. Il permet de craquer un mot de passe spécifique de multiples façons, combinées avec flexibilité et rapidité. Craquer les hachages de mots de passe avec Hashcat est une approche très différente de celle de l'utilisation du PRTK.

Les différences les plus notables sont peut-être les suivantes :

1. Hashcat est un outil de crack et non un utilitaire de dictionnaire.
2. C'est un cracker de la CLI (command-line interface), pas une interface graphique.
3. Il utilise le GPU pour cracker les mots de passe, au moins c'est quand il fait de son mieux.

II.9 Récupération des données

II.9.1 Récupération de données effacées

La récupération des informations perdues et supprimées est une tâche très usuelle pour un expert légiste sur ordinateur.

Dans les enquêtes criminelles, les fichiers supprimés sont souvent d'une grande importance car, en fait, les criminels préfèrent couvrir leurs traces. En plus, dans un environnement d'entreprise, il est également possible de récupérer des fichiers supprimés lorsqu'un fichier est supprimé par erreur ou un problème survient sur un support de stockage.

Le processus de récupération des fichiers supprimés est généralement appelé récupération de données et peut être effectué de différentes manières[Käv18].

Il y a deux manières suivantes :

- Récupération de fichiers supprimés de la MFT
- Sculpture de fichier

Il est important que l'examinateur connaît les détails du système de fichiers avant de commencer avec les outils disponibles, ce qui permet de vérifier les résultats [Sam12]. Le système de fichiers Windows NT (NTFS) fournit une combinaison de performances, la fiabilité et la compatibilité ne trouve pas dans le système de fichiers FAT.

II.9.1.1 Récupération de fichiers supprimés de la MFT

On a rappelé que les systèmes Windows utilisent couramment le système de fichiers NTFS et que tous les fichiers sont répertoriés dans la Master File Table (MFT).

Quand un fichier est détruit, on supprime le pointeur du fichier dans le MFT, mais le fichier se trouve généralement sur le disque dur, le fichier est conservé jusqu'à ce qu'il soit écrasé On peut restaurer les fichiers supprimés de cette manière en recherchant dans les secteurs de la partition pour trouver contenant des fichiers non inclus dans le fichier MFT. Car le fichier n'est n'a pas été perdu, le processus de récupération est simple [Nih19].

Ce processus peut être réalisé avec des outils disponibles en ligne et est souvent effectué automatiquement par des outils d'investigation.

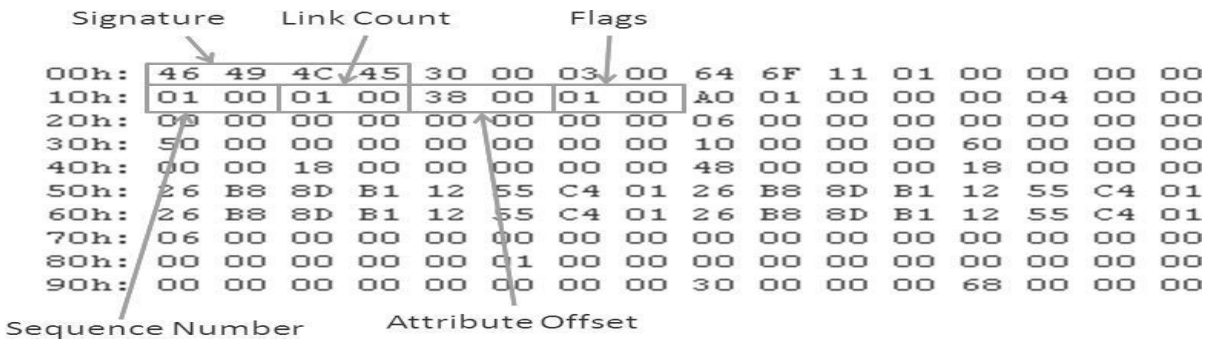


Figure 7. Fichiers de métadonnées MFT [Nih19].

II.9.1.2 Sculpture de fichiers

La sculpture de fichiers est un type avancé de récupération de données, généralement utilisé dans les enquêtes numériques pour extraire un fichier spécifique (en utilisant des informations d'en-tête et de footer du fichier) depuis un espace non alloué (données brutes), sans l'aide de toute structure système (p.ex. MFT).

La sculpture de fichiers est une excellente méthode pour récupérer des fichiers et des fragments de fichiers lorsque des entrées de répertoire sont corrompues ou manquantes [Nih19], Ceci est particulièrement utilisé par les experts criminalistique dans les affaires criminelles pour récupérer des preuves.

Par exemple, l'US Navy Seals ont récupéré les disques durs et les supports de stockage amovibles de la base d'Oussama ben Laden pendant leur raid. Les experts médico-légaux ont utilisé des techniques de gravure de fichiers pour extraire toutes les informations de ce support [Ins19].

II.9.1.2.1 Sculpture de fichier avec un éditeur Hex

La sculpture de fichiers peut être effectué en utilisant uniquement un éditeur hexadécimal. Cependant, il existe certains outils qui peuvent aider les examinateurs. Voici quelques outils gratuits permettant de sculpter des fichiers:

1. Foremost (http://foremost.sourceforge.net).
2. Scalpel (https://github.com/sleuthkit/scalpel).
3. Jpegcarver (www.seedstech.net/jpegcarver).
4. Liste des outils de récupération de données (y compris certains outils de gravure de fichiers) de forensics wiki (www.forensicswiki.org/wiki/Tools:Data_Recovery).

II.9.2 Analyse des fichiers de métadonnées

Il est facile de se concentrer sur le contenu des fichiers, mais les métadonnées sont souvent d'un intérêt égal pour un examinateur judiciaire.

Le reste de cette section traite des types de métadonnées qui intéressent souvent les experts légistes, de la façon suivante :

- Horodatage NTFS
- Données EXIF
- Métadonnées Office

II.9.2.1 Horodatage NTFS

Tous les fichiers créés sur un système de fichiers NTFS comportent des horodatages, et ces derniers peuvent vous donner des informations sur le passé d'un fichier [Käv18].

Les horodatages qui se trouvent dans le système de fichiers NTFS sont les suivants [Knu16]:

- Création, indiquant quand le fichier a été créé dans le système
- Modification, indiquant quand le fichier a été modifié pour la dernière fois
- Accéder, indiquant quand le fichier a été lu pour la dernière fois
- Heure modifiée MFT, indiquant quand la dernière modification des métadonnées de fichier.

II.9.2.2 Données EXIF

Les données EXIF constituent une source d'information très riche, un format de fichier normalisé pour les images utilisées par les appareils photographiques numérique[Käv18].

Les données EXIF donnent beaucoup d'informations sur une photo, y compris ce qui suit :

- Nom de l'appareil qui prend la photo
- Numéro de série de l'appareil qui prend la photo
- Version et modèle de l'appareil qui prend la photo
- Coordonnées GPS décrivant le lieu où la photo a été prise

II.9.2.3 Métadonnées Office

Les fichiers Microsoft Office contiennent beaucoup de métadonnées qui peuvent intéresser un expert légiste sur un ordinateur [Sof09].

Les métadonnées office sont riches de plusieurs éléments d'information, notamment :

- Nom de l'auteur original.
- Nom de la dernière personne qui a enregistré le document.
- Date de création d'origine.
- Date du dernier enregistrement.
- La dernière impression du document.
- Temps total de traitement du document.

II.9.3 Analyse les fichiers de Log

Dans un examen criminalistique informatique, il est possible d'analyser les différentes applications qui ont été utilisées. On propose un exemple d'une approche qui permet d'analyser les logs de chat et qui peut également être appliquée aux logs d'application.

En bref, ce processus d'analyse des logs de chat peut être résumé comme suit :

- Utiliser les outils préférés de la police scientifique pour trouver les logs du chat.
- Comprendre le log du chat en l'examinant et, au besoin, conduire des tests.
- Préparer le log de chat pour la présentation, de préférence de manière automatique.

II.9.4 Analyse des données non organisées

Le disque dur contient des données non organisées, Il s'agit notamment de Pagefile, Hiberfile et décharges de mémoire [Käv18]. Le point commun de ces sources de données est qu'elles ne sont pas traitées comme le reste du système d'exploitation, ce qui en fait une source d'artefacts qui peut être extrêmement utiles !.

Pagefile et Hiberfile contient généralement le même type d'informations que celles d'un dump mémoire.

Le Pagefile est utilisé par l'ordinateur lorsqu'il doit paginer des parties de la RAM pour décharger dans un autre espace [Käv18].

Le Hiberfile enregistre l'état actuel de la machine lorsqu'un ordinateur Windows est en mode hibernation, afin de permettre à l'ordinateur de se réveiller à nouveau [Käv18].

L'information trouvée ici est très utile, car il s'agit généralement d'informations qui sont utilisées par l'ordinateur dans un passé très proche, c'est particulièrement vrai pour les décharges mémoire. De plus, Il est important de noter que le succès de ce type de recherche dépend en grande partie de l'expérience : savoir ce que l'on cherche et ce qui se trouve en général. Cela étant dit, les sources de données non organisées constituent une excellente source de données à ne pas négliger.

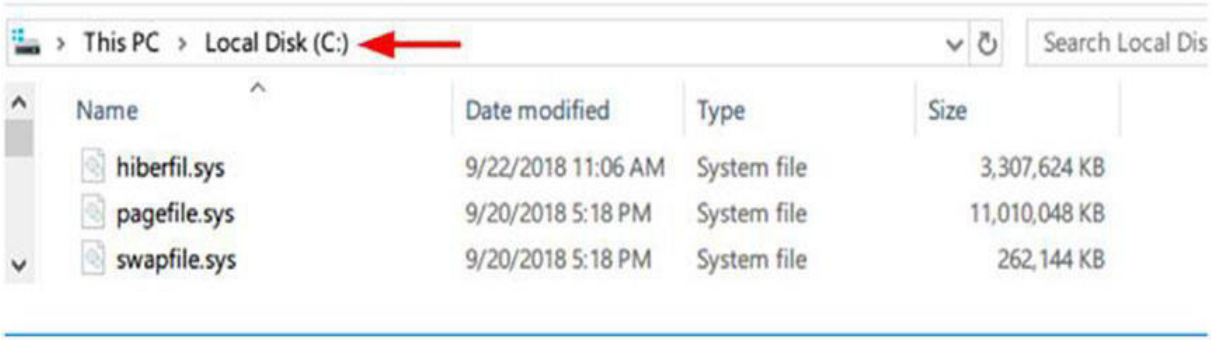


Figure 8. L'emplacement des fichiers Swapfile.sys, Hiberfil.sys et Pagefile.sys [Nih19].

II.10 Conclusion

Dans ce chapitre, On a présenté une introduction au domaine des investigations numériques et la cybercriminalité, couvrant le processus et les définitions des termes de base, et on inclura également la recherche, la collecte et l'analyse des preuves numériques.

III.1 Introduction

L'investigation numérique de la mémoire est le domaine le plus réussi, le plus intéressant et le plus exigeant de la criminalistique numérique. L'exploration sur la mémoire numérique fournit une vision fiable de l'état d'exécution du système, notamment des processus exécutés, connexions réseau ouvertes et commandes récemment exécutées[Gra14].

On peut extraire ces artefacts de manière indépendante du système que on étudie, réduisant ainsi le risque que des logiciels malveillants ou des rootkits impactent avec nos résultats.

Ce chapitre décrit les principes fondamentaux du mémoire volatile et fournit une introduction pratique à l'outil Volatility qui peut être utilisé pour l'analyse de cette mémoire dans un contexte criminel ou d'entreprise.

III.2 Investigation numérique sur la mémoire

L'investigation sur la mémoire physique (mémoire volatile) est un domaine relativement nouveau de l'ordinateur forensics, où un enquêteur recueille la mémoire dump et effectue l'analyse dans un environnement isolé.

Le but de cette information est principalement de diagnostiquer les problèmes et d'obtenir une idée de ce qui se passe sur le système pendant son fonctionnement [Bri04] et [Kev03].

III.2.1 La mémoire volatile

La mémoire volatile préserve l'information pendant une courte période de temps. En effet, l'ordinateur a besoin d'énergie pour conserver les données, mais lorsqu'il est éteint, il perd rapidement ses informations. Les données volatiles se trouvent dans les registres, le cache et la RAM, qui est la source la plus importante [Dha16].

La mémoire vive (RAM) d'un ordinateur peut contenir de nombreuses informations : mots de passe, identifiants, clefs de chiffrement ou encore processus actifs. Lors d'une analyse forensic, l'étude de l'image mémoire d'un système peut s'avérer utile [Lord18].

Voici d'autres exemples de données volatiles :

- **Le temps du système :**

Les analystes doivent enregistrer l'heure et la date sur le système suspect et les comparer avec celle du système actuel. L'enregistrement des dates et des heures permet aux analystes de documenter le moment où une enquête sur un incident a commencé, de recueillir des données volatiles et de complète l'investigation sur un incident [Dha16].

- **Les connexions de réseau :**

La capture d'un instantané des connexions réseau existantes donnera aux analystes une idée des ports ouverts, ainsi que des processus qui ont établi des connexions et des indices sur les données qui ont été transmises [Dha16].

- **Historique des commandes :**

L'examen de l'historique des commandes d'un système suspect montre les activités récentes des utilisateurs et sert de trace de vérification des activités d'enquête. Dans la mesure du possible, l'historique des commandes devrait être obtenu pour chaque compte d'utilisateur sur un système suspect. Les commandes à rechercher incluent celles utilisées pour gérer des comptes d'utilisateurs, configurer les périphériques et installer les logiciels [Dha16].

- **Processus en cours :**

L'examen de la liste des processus s'exécutant sur un système permet les analystes à détecter les attaques malveillantes ou des processus suspects. Lors de la récupération de données volatiles, les analystes doivent savoir que les programmes malveillants peuvent avoir des noms qui semblent valides [Dha16].

III.2.2 Acquisition de la mémoire volatile

L'acquisition de la mémoire vive intervient généralement lors d'une réponse à incident et sur un système en fonctionnement. Lors de l'acquisition, l'analyste doit éviter au maximum toutes modifications afin de récupérer une image fidèle du système à analyser. Certains outils permettent spécifiquement de copier le contenu de la mémoire RAM.

III.2.2.1 L'utilitaire Dumpit

Afin de réaliser une image de la RAM, on utilisera l'utilitaire gratuit de la société Moonsols: Dumpit, disponible sur Internet [Zel17].

Cet outil est composé des 2 utilitaires suivants:

- Win32DD: permettant de réaliser une image mémoire d'un système Windows 32 bits.
- Win64DD: permettant de réaliser une image mémoire d'un système Windows 64 bits.

On lance l'utilitaire

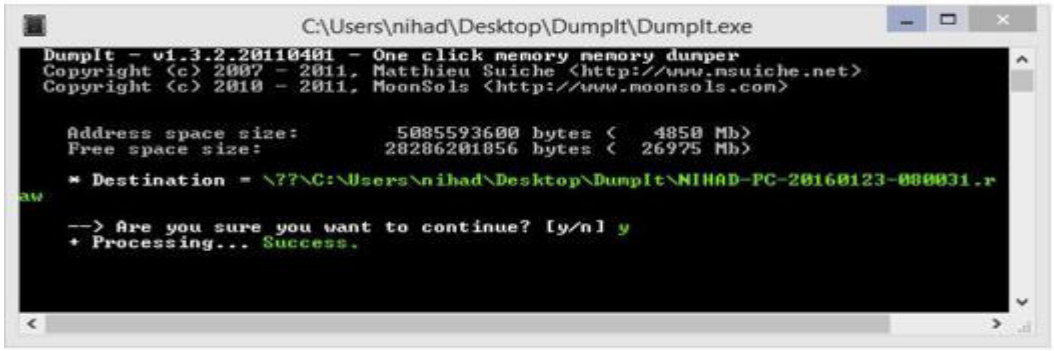
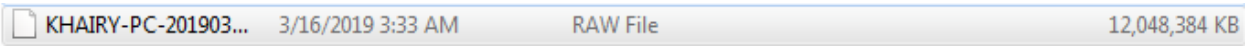


Figure 9. Copie de la mémoire vive avec l'utilitaire Dumpit [Zel17].

On obtient ici une image mémoire de 12.4 GB, qui pourra par la suite être analysée.



III.2.2.2 Extraction de la mémoire vive par « cold boot »

Lorsque l'on éteint subitement un ordinateur, la totalité des données présentes dans la mémoire vive est disponible environ 5 à 10 secondes. Le tableau ci-dessous indique le temps de rémanence des données après extinction.

Temps	% de données
5 à 10 secondes	100%
10 à 60 secondes	80%
1 à 3 minutes	50%
après 3 minutes	0% à 2%

Tableau 1. Temps de rémanence des données contenues dans la RAM [Pet11].

Une étude menée par des chercheurs de l'université de Princeton aux Etats-Unis en 2008, a montré que refroidir la mémoire vive à très basse température permettait de conserver plus longtemps les données présentes [Pet11]. Il devient ainsi possible d'extraire les données sans être contraint par le temps. Par exemple, à moins 50 degrés les données peuvent être conservées jusqu'à 10 minutes. En plongeant les barrettes mémoires dans l'azote liquide, le temps de conservation peut alors s'élever à plusieurs heures, voire plusieurs jours.

Pour réaliser cette technique, on devrait:

1. Refroidir les barrettes mémoires sur un ordinateur allumé
2. Couper brusquement l'alimentation de l'ordinateur
3. Brancher les barrettes sur un autre ordinateur (ou sur le même)
4. Booter sur un système tiers afin d'éviter que le système n'écrase les données
5. Dumper la mémoire.

Une bombe à air sec (disponible sur le marché), On permet de refroidir les barrettes mémoires.

Figure 10. Bombe à air sec.

Puis on utilise l'utilitaire MSRAMDMP[Rob16], Refroidissement de la mémoire vive de l'ordinateur allumé:

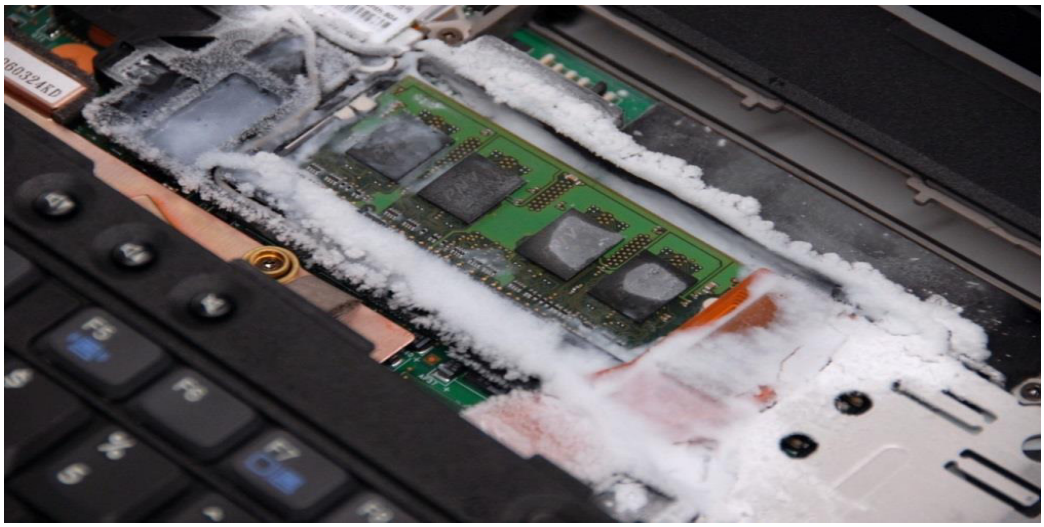


Figure 11. Gel de barrette mémoire [Pet11].

Redémarrage de l'ordinateur et boot sur clé USB préparée avec l'utilitaire MSRAWSMP et copie du contenu de la RAM pour récupération et exploitation des données:

Figure 12. Copie de la mémoire vive avec l'utilitaire MSRAWSDMP [Pet11].

III.2.3 Analyse de la mémoire volatile

Afin de procéder à l'analyse de la mémoire volatile, on discutera ici de l'outil le plus connu et le plus utilisé dans l'investigation numérique pour les systèmes Windows : le Volatility Framework [Vol18].

III.2.3.1 Présentation du Framework Volatility

Volatility est un Framework open-source écrit en python, utilisé pour les réponses à incident ou pour l'analyse de malware. Il est distribué par la société Volatile System [Gra14].

Le Framework Volatility dispose des fonctionnalités suivantes:

- Informations sur l'image (date, heure, nombre d'UC).
- Extraction des processus en cours.
- Extraction des processus SID et des variables d'environnement.
- Affichage des sockets réseau.
- Affichage des connexions réseau.
- Dump des hash LM/NTLM et secrets LSA.

La liste des options de Volatility est disponible en annexe.

III.2.3.2 Identification du profil

La première étape pour l'analyse de l'image mémoire est d'utiliser l'option « imageinfo » pour obtenir les informations concernant le système d'exploitation de l'image.

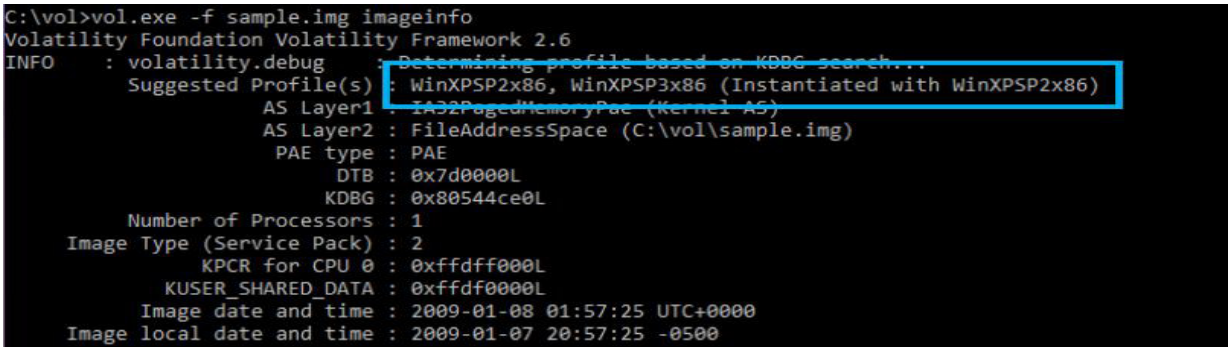


Figure 13. Option imageinfo de Volatility [Gra14].

La commande retourne les informations du système. Il faudra alors utiliser le profile Win7SP1x86 ou Win7SP0x86 avec l'option « --profile= », pour le reste de l'analyse.

III.2.3.3 Lister les processus en cours

Afin d'identifier les applications utilisées lors de la récupération de l'image mémoire, Volatility permet de lister les processus en cours d'utilisation avec l'option « pslist ».

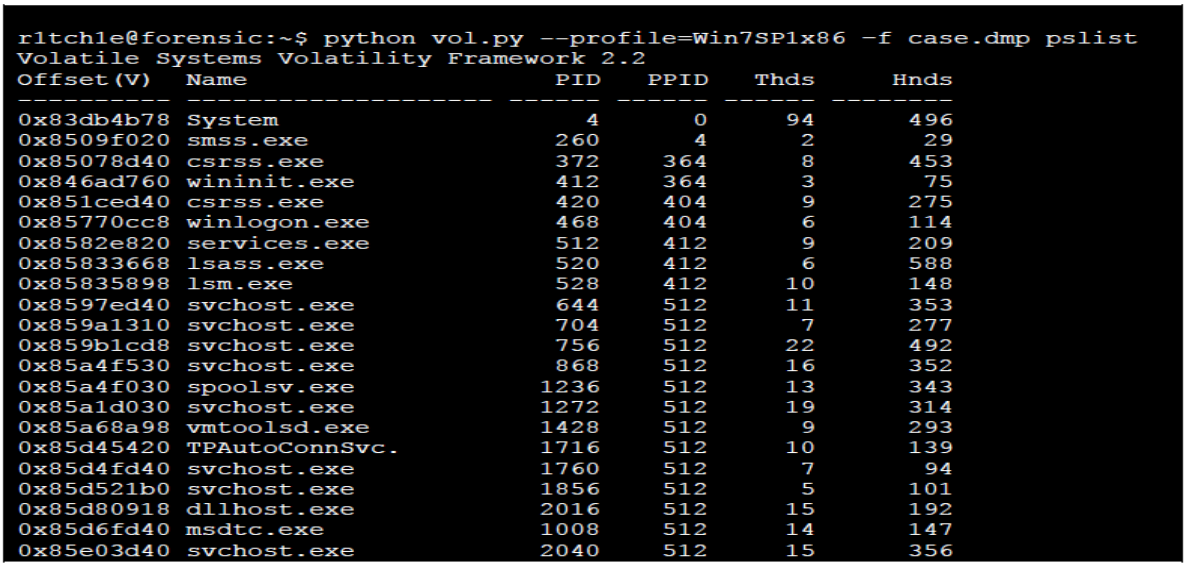


Figure 13. 1. Option pslist de Volatility [Gra14].

Dumper l'image mémoire d'un processus l'option « memdump » permet d'extraire l'image d'un processus afin de l'analyser. Pour dumper le processus il faut d'abord récupérer son PID à l'aide de la commande « pslist », Ici on dumpe le processus « KeePass.exe » avec le PID 852.

```
r1tch1e@forensic:~$ python vol.py --profile=Win7SP1x86 -f case.dmp memdump
-p 852 --dump-dir /tmp/
Volatile Systems Volatility Framework 2.2
************************************************************************
Writing KeePass.exe [  852] to 852.dmp
```

Figure 13. 2. Option memdump de Volatility [Fort17].

Dans ce cas, Volatility crée un fichier nommé 852.dmp qui peut ensuite être analysé. Il est également possible de décharger le processus en tant qu'exécutable avec l'option "procmemdump". L'exécutable peut alors être décompilé et analysé. Par exemple, le processus "svchost.exe" avec le PID 2040.

```
r1tch1e@forensic:~$ python vol.py --profile=Win7SP1x86 -f case.dmp
procmemdump -p 2040 --dump-dir /tmp/
Volatile Systems Volatility Framework 2.2
Process(V) ImageBase  Name                 Result
---------- ---------- -------------------- ------
0x85e03d40 0x00890000 svchost.exe          OK: executable.2040.exe
```

Figure 13. 3. Option procmemdump de Volatility [Fort17].

III.2.3.4 Extraction des hash LM/NTLM

Les données du registre Windows peuvent être récupérées lors d'un dump de mémoire volatile. Volatility offre plusieurs options pour afficher, explorer (hivelist, hivescan...) le registre et extraire les identifiants système (hashdump). On procédera comme suit pour extraire les hash NTLM.

```
r1tch1e@forensic:~$ python vol.py --profile=Win7SP1x86 -f case.dmp hivelist
Volatile Systems Volatility Framework 2.2
Virtual     Physical    Name
----------  ----------  ----
0x960de9d0 0x135329d0 \SystemRoot\System32\Config\SECURITY
0x961b78e8 0x12a3f8e8
\??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x82bb3140 0x02bb3140 [no name]
0x8760c008 0x19e8d008 [no name]
0x8761c008 0x19e1b008 \REGISTRY\MACHINE\SYSTEM
0x8763c6b8 0x19cbb6b8 \REGISTRY\MACHINE\HARDWARE
0x876c69d0 0x19e569d0 \SystemRoot\System32\Config\DEFAULT
0x882569d0 0x074a09d0 \SystemRoot\System32\Config\SAM
0x882f43f0 0x071193f0
\??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8b2999d0 0x1966c9d0 \Device\HarddiskVolume1\Boot\BCD
0x8cfb9008 0x17aaa008 \SystemRoot\System32\Config\SOFTWARE
0x900ec008 0x04efb008 \??\C:\System Volume Information\Syscache.hve
0x90bf5008 0x0157c008 \??\C:\Users\R1TCH1E\ntuser.dat
0x92ac7650 0x0f165650
\??\C:\Users\R1TCH1E\AppData\Local\Microsoft\Windows\UsrClass.dat
```

Figure 13. 4. Liste des ruches systèmes avec l'option hivelist de Volatility [Fort17].

On utilisera ensuite l'option « hashdump » en précisant le « -y » (qui correspond à l'OFFSET de la ruche SYSTEM) et l'option « -s » (qui correspond à l'OFFSET de la ruche SAM, où sont stockés les mots de passe).

```
r1tch1e@forensic:~$ python vol.py --profile=Win7SP1x86 -f case.dmp hashdump
-y 0x8761c008 -s 0x882569d0
Volatile Systems Volatility Framework 2.2
Administrateur:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c
0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
R1tch1e:1000:aad3b435b51404eeaad3b435b51404ee:b9f917853e3dbf6e6831ecce60725930:::
```

Figure 13. 5. Extraction des hash avec l'option hashdump de Volatility [Fort17].

III.2.3.5 Casser les hashes

Enfin, on peut manipuler le hachage à l'aide d'un outil local (comme HashCat) ou d'un outil en ligne comme HashKiller, Après avoir récupérer les hash il est possible de les décrypter. La copie d'écran ci-dessous illustre le décryptage du mot de passe [Md5]:

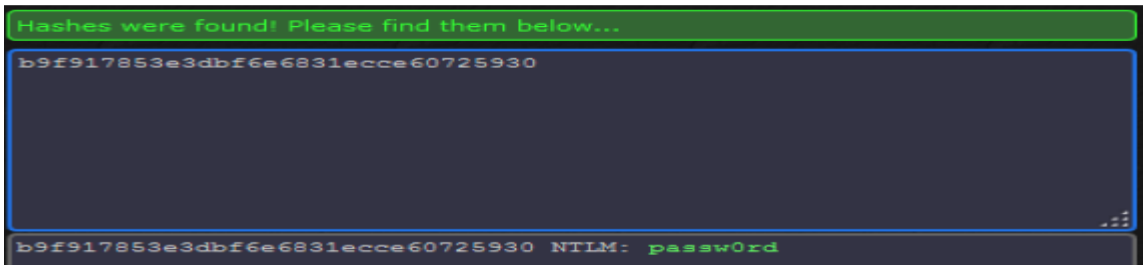


Figure 13. 6. Décryptage du hash NTLM [Fort17].

Le hash retrouvé correspond ici à « passw0rd », On notera tout de même qu'un mot de passe fort (constitué de plus de 8 caractères alphanumériques minuscules, majuscules et caractères spéciaux) est très difficile à décrypter.

III.2.4 Analyse des Malware

Le nombre de cyber-attaques est sans aucun doute en augmentation, ciblant les secteurs gouvernemental, militaire, public et privé [Ran17], Ces cyber-attaques visent à cibler des individus ou des organisations dans le but d'extraire des informations précieuses.

Les malwares (Trojan, Botnet, Vers…) sont depuis quelques années au cœur des problématiques de sécurité et ne cessent de se développer tout en devenant de plus en plus complexes.

L'analyse de malware est une branche spécifique de l'investigation numérique, réalisée généralement en laboratoire, elle peut intervenir lors d'une réponse à incident dans le but d'identifier les machines infectées [Mon18].

III.2.4.1 Définition et types de logiciels malveillants

Un logiciel malveillant est un code qui exécute des actes malveillants (vols de données, espionnage industriel, dénis de service…), il peut être sous forme exécutable, script, code, ou tout autre logiciel [Mon18]. Généralement, il est introduit dans votre système de manière non autorisée et peut être transmis par des divers canaux de communication tels que le e-mail, le Web ou les clés USB.

L'illustration ci-dessous présente de manière générale la diffusion d'un malware.

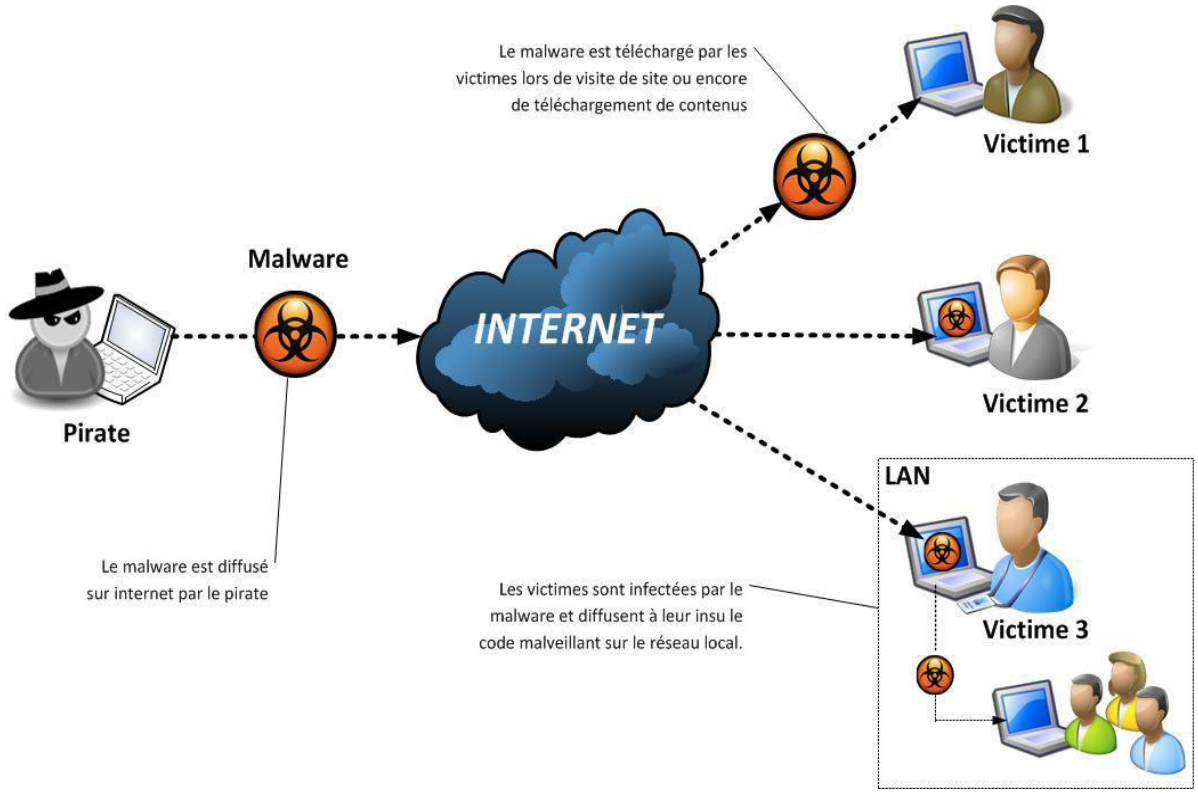


Figure 14. Représentation de la diffusion de malware.

Voici la liste des principaux types de malwares :

Virus ou ver : Logiciel malveillant capable de se copier et de se propager à d'autres ordinateurs. Un virus a besoin de l'intervention de l'utilisateur, alors qu'un ver peut se propager sans intervention de l'utilisateur [Mon18].

Trojan : Logiciel malveillant qui qui se déguise en programme régulier pour inciter les utilisateurs à l'installer sur leurs systèmes [Mon18], Une fois installé, il peut effectuer des actions malveillantes telles que le vol de données sensibles, le téléchargement de fichiers sur le serveur de l'attaquant ou la surveillance des webcams.

Adware : Logiciel malveillant qui propose à l'utilisateur des publicités indésirables (annonces). En général, ils sont transmis via des téléchargements gratuits et il est possible de forcer l'installation de logiciels sur notre système [Mon18].

Rootkit : logiciel malveillant qui fournit à l'attaquant un accès privilégié au système infecté et dissimule sa présence ou celle d'autres logiciels [Mon18].

Ransomware : est un type de malware de la cryptovirologie qui menace de publier les données de la victime ou de bloquer en permanence l'accès à ces données, Une somme d'argent est demandée en échange de la clef de déchiffrement [Mon18]. Fin 2013, le ransomware CryptoLocker avait contaminé près de 250 000 machines dont 5,8% se trouvait en France.

III.2.4.2 Pourquoi l'analyse des logiciels malveillants ?

Le motif principal de l'analyse des logiciels malveillants est d'extraire des informations de l'échantillon de ce dernier, ce qui peut aider la réponse à un incident de logiciel malveillant [Mon18]. Voici certaines des raisons pour lesquelles ont effectué une analyse des logiciels malveillants :

- ✓ Déterminer la nature et le but du malware. Par exemple, il peut-on aider à déterminer si un logiciel malveillant est un voleur d'informations, un bot HTTP, un bot spam, un rootkit, un keylogger ou un fichier RAT, etc…

- ✓ Comprendre comment le système a été piraté et son impact. Par exemple, au cours de notre analyse, si on détermine qu'un programme malveillant contacte une adresse domaine/IP particulière, on peut utiliser cette adresse domaine/IP pour créer une signature et surveiller le trafic réseau pour identifier tous les hôtes qui contactent cette adresse domaine/IP.

- ✓ Déterminer l'intention et le motif de l'attaquant. Par exemple, au cours de l'analyse, si on trouve que le malware vole des identifiants bancaires, on peut en déduire que le motif de l'attaquant est un gain financier.

III.2.4.3 Approche pratique de l'analyse des logiciels malveillants par l'outil Volatility

Pour le travail expérimental, on examinera un échantillon de mémoire qui est infecté par un cheval de Troie appelé **Spyeye**.

On se rappellera les étapes 1 et 2 de la phase d'analyse de la mémoire volatile.

III.2.4.3.1 Identification du profil

III.2.4.3.2 Lister les processus actifs

III.2.4.3.3 Explorer dans l'analyse des facteurs de risques

On commencera à rechercher des pointeurs pour identifier le père et le fils. Pour afficher les processus sous un format Père-fils, on peut organiser les processus de la manière la plus structurée en utilisant l'option pstree.

```
C:\vol>vol.exe -f sample.img --profile=WinXPSP2x86 pstree
Volatility Foundation Volatility Framework 2.6
Name                                                  Pid   PPid   Thds   -
-------------------------------------------------- ------ ------ ------
 0x817cc7f8:System                                      4      0     53
. 0x8140f600:smss.exe                                 516      4      3
.. 0x81712170:csrss.exe                               588    516      9
.. 0x8172d2d8:winlogon.exe                            612    516     20
... 0x81459b38:services.exe                           656    612     16
.... 0x812aa3c0:svchost.exe                          1232    656      6
.... 0x8163d020:alg.exe                               408    656      6
.... 0x815db628:svchost.exe                          1020    656     18
.... 0x81504c30:spoolsv.exe                          1980    656     16
.... 0x814068b0:svchost.exe                           984    656     80
..... 0x814bc988:wscntfy.exe                         1048    984      1
.... 0x8143c388:svchost.exe                           888    656     10
.... 0x813df020:msiexec.exe                           412    656      3
.... 0x8170cd50:svchost.exe                          1304    656     13
... 0x812ee9a0:lsass.exe                              668    612     18
 0x81290920:explorer.exe                             1928   2000     13
```

Figure 15. Forensics de mémoire de volatilité | Pstree [Gra14].

Avec ce résultat, on peut vérifier à partir de la liste de processus que les paramètres sont appropriés. Maintenant, on va examiner si des processus terminés sont actifs dans la mémoire en utilisant l'option psxview.

```
$ python vol.py -f infected.vmem --profile=WinXPSP3x86 psxview
Volatility Foundation Volatility Framework 2.6
Offset(P)  Name                PID pslist psscan thrdproc pspcid csrss session deskthrd ExitTime
---------- -------------------- ------ ------ ------ -------- ------ ----- ------- -------- --------
0x01956b08 alg.exe              564 True   True   True     True   True  True    True
0x01857910 lsass.exe            712 True   True   True     True   True  True    True
0x01945da0 wuauclt.exe         1452 True   True   True     True   True  True    True
0x019e2818 svchost.exe         1112 True   True   True     True   True  True    True
0x01587710 explorer.exe        1456 True   True   True     True   True  True    True
0x01859020 services.exe         700 True   True   True     True   True  True    True
0x015dc1a8 winlogon.exe         656 True   True   True     True   True  True    True
0x015254b0 wmiprvse.exe         420 True   True   True     True   True  True    True
0x015d7688 svchost.exe          884 True   True   True     True   True  True    True
0x015b0da0 vmtoolsd.exe        1984 True   True   True     True   True  True    True
0x0156a0e8 ctfmon.exe          1764 True   True   True     True   True  True    True
0x0170b020 svchost.exe         1184 True   True   True     True   True  True    True
0x01553c88 lsass.exe           1664 True   True   True     True   True  True    True
0x016ba360 nvid.exe            1700 False  True   True     True   True  True    True
0x01af5d10 svchost.exe          964 True   True   True     True   True  True    True
0x01706c68 spoolsv.exe         1388 True   True   True     True   True  True    True
0x015cf5a0 svchost.exe         1052 True   True   True     True   True  True    True
0x016d8380 smss.exe             380 True   True   True     True   False False   False
0x013ee858 cmd.exe             2284 False  True   False    False  False False   False    2014-10-17 09:17:21 UTC+0000
0x01bcc830 System                4 True   True   True     True   False False   False
0x01aa1868 csrss.exe           632 True   True   True     True   False True    True
```

Figure 15. 1. Forensics de mémoire de volatilité | Psxview [Gra14].

De la sortie ci-dessus, on peut affirmer qu'aucun des processus est terminé puisque tous les processus apparaissent dans la pslist et la sortie psscan. Les processus terminés sont marqués comme faux dans la liste des processus. Cela signifie que ces processus ont été arrêtés avant de capturer la mémoire de la machine.

L'étape suivante est de vérifier si un processus essaye de connecter les adresses IP à distance. Avec l'option connscan, il est possible de trouver un tel processus.

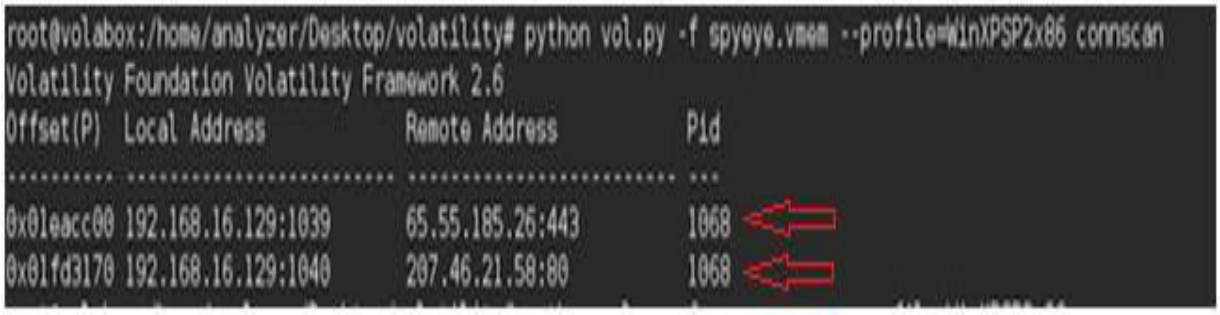


Figure 15. 2. Forensics de mémoire de volatilité | ConnScan [Pri18].

A partir des résultats ci-dessous, les points importants à noter sont les suivants :

- L'IP locale 192.168.16.129 avec Process ID 1068 essaye de connecter deux adresses distantes 207.46.21.58 & 65.55.185.26.
- Le processus avec PID 1068 est svchost.exe avec père ID 656, c'est-à-dire services.exe.

L'étape suivante est la vérification de la mémoire des dlls suspectes pour déterminer la connexion probable disponible dans la machine victime qui est introduite par le processus malveillant. Avec l'option dlllist affiche toutes les dll disponibles en mémoire. Une inspection manuelle est nécessaire pour vérifier si une fichier dll suspecte est trouvée ou non.

```
C:\vol>vol.exe -f sample.img --profile=WinXPSP2x86 dlllist
```

Figure 15. 3. Forensics de mémoire de volatilité | DllList [Pri18].

La sortie de la dlllist est assez grande. Dans cet exemple, on obtient l'entrée suspecte dll comme dll.dll dans le chemin C:\WINDOWS\system32\

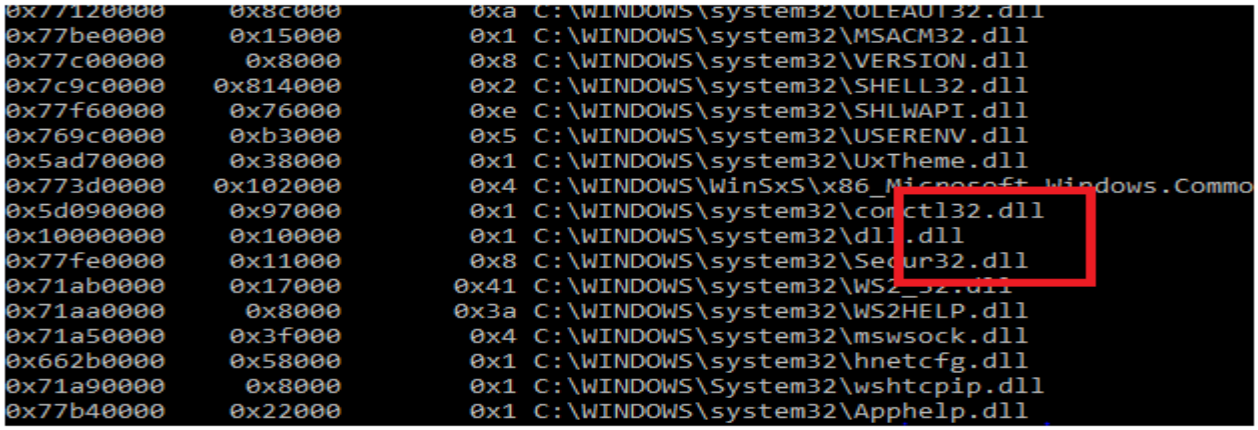


Figure 15. 4. Forensics de mémoire de volatilité | résultat de la dlllist [Pri18].

Puisque on a une entrée dll suspecte qui doit être vérifiée en détail, une investigation ultérieure avec quelques commandes plus liées à la DLL on aidera à trouver les traces de la partie infectée de la mémoire et des processus correspondants. Avec l'option ldrmodules détecte la dll non associée dans la mémoire. Le résultat de cette commande est très gros, mais l'inspection rapide peut-on amener à obtenir les traces appropriées.

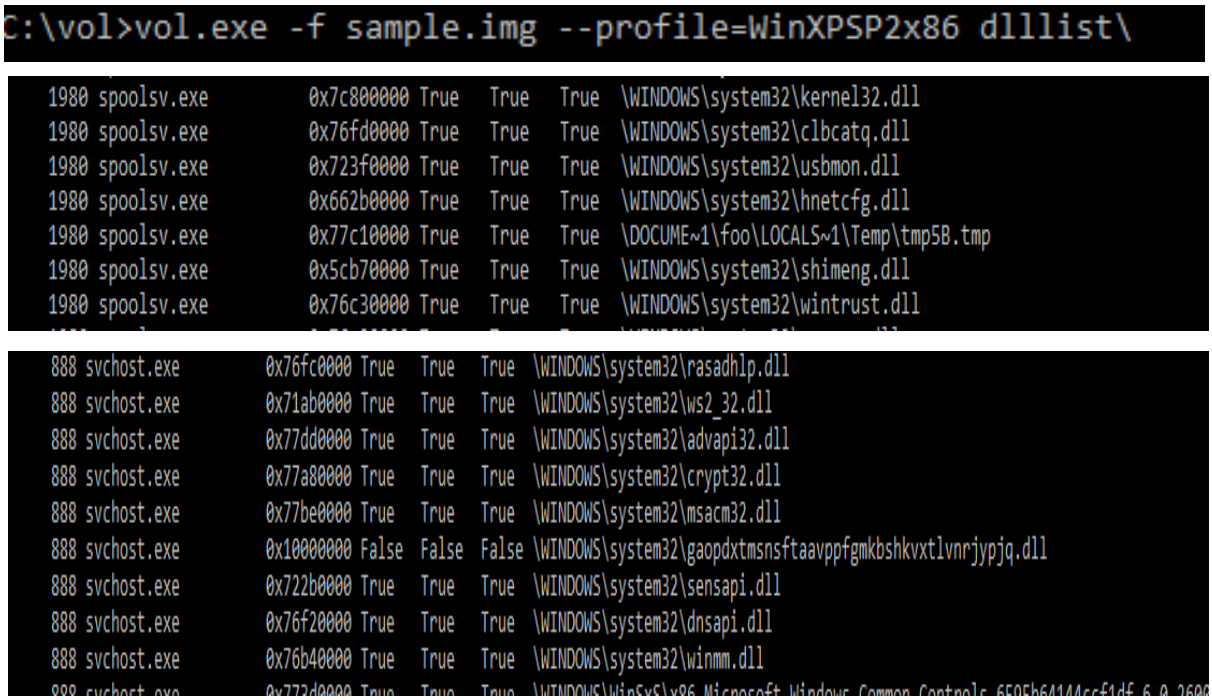


Figure 15. 5. Forensics de mémoire de volatilité | ldrmodules [Pri18].

Avec l'option ldrmodules donne les deux résultats suspects dans le cas étudié, **un nom de fichier dll suspect dans un chemin system32** et **le fichier temporaire qui aurait pu être créé lors de l'exécution d'un malware dans la mémoire de la machine**.

III.2.4.3.4 Analyse et résultats

D'après l'analyse approximative, on a recueilli suffisamment d'IOC pour détecter l'infection par le malware. De plus, une analyse plus précise de la mémoire permet également de tracer le chemin complet de la mémoire et l'emplacement exact dans cette dernière.

Ce qui suit sont les indicateurs d'infections par des logiciels malveillants dans l'ordinateur de la victime :

- L'IP locale 192.168.16.129 avec Process ID 1068 essaye de connecter deux adresses distantes 207.46.21.58 & 65.55.185.26.
- Le processus avec PID 1068 est svchost.exe avec Père ID 656, c'est-à-dire services.exe.
- L'IP local 192.168.16.129 se connecte à lui-même via le Process ID 1980.
- La présence du nom suspect de la fichier DLL "gaopdxtmsnsftaavppfgmgmkbshkvxtlvnrjypjypjq.dll" dans la commande ldrmodules confirme la dll détachée qui a pu fonctionner sur le système. La fichier dll est de nouveau associée au PID 888 (svchost.exe).

Maintenant, pour confirmer les résultats de la démonstration expérimentale on vérifiera les détails des adresses IP distantes ainsi que les dlls malveillantes. Il est possible que les adresses IP distantes soient des adresses IP de serveurs d'applications légitimes.

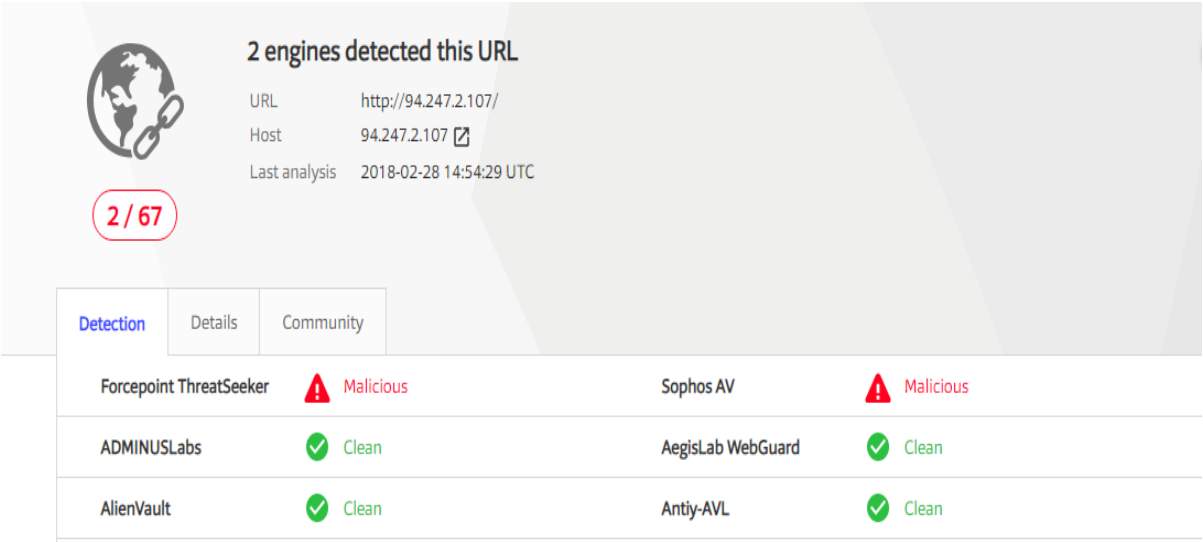


Figure 15. 6. Virus Total détecté ip malveillante [Pri18].

Deux moteurs du virus total ont détecté qu'il s'agit de l'IP malveillant, En vérifiant l'adresse IP de la connexion à distance dans la base de données whois, on a obtenu les résultats suivants:

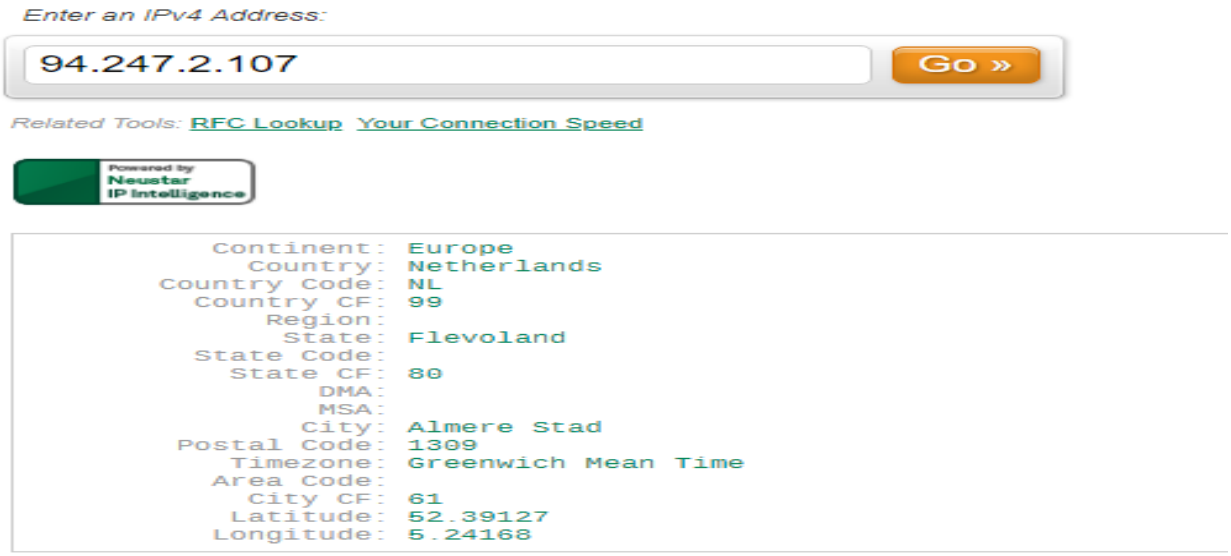


Figure 15. 7. Résultats détecté ip malveillante [Pri18].

D'après les résultats, il est clair que l'IP est enregistrée dans le pays distant et une analyse plus précise de la menace on amène à la conclusion que l'adresse IP est liée aux activités malveillantes. On peut obtenir la même conclusion après avoir analysé les fichiers dll et les fichiers temporaires suspects dumpés sur la solution en ligne qui traite l'infection par les logiciels malveillants dans la machine.

III.3 Travaux connexe

III.3.1 Extraction de mots de passe

La plupart des recherches ont été consacrées à l'énumération des processus et des threads en accédant aux objets internes dans la mémoire. Cependant, la collecte d'informations sensibles à la casse à partir du contenu de la mémoire extraite est important et difficile dans l'ordinateur forensics. De plus, le mot de passe enregistré dans le système va jouer un rôle important dans l'identification du comportement de l'utilisateur.

Les auteurs **[Xu and Wang, 13]**, ont proposé une approche pour extraire les informations de texte en clair du mot de passe enregistrés dans la mémoire physique du système Windows.

III.3.1.1 Extraction le mot de passe dans un système en fonctionnement

La réalisation de l'algorithme de décryptage dépend principalement sur les trois points suivants [Sek13] :

1. Acquérir ciphertext et sa longueur en inversant Wdigest.dll.
2. Obtenir l'adresse de la fonction de décryptage pour récupérer cette adresse, on recherche une valeur hexadécimale spéciale dans lsasrv.dll.
3. Extraire les variables globales à l'aide de la fonction de déchiffrement.

III.3.1.2 Extraction le mot de passe enregistrés dans la mémoire physique

L'analyse du texte en clair du mot de passe à partir d'un fichier image mémoire est différente de celle d'un ordinateur en cours d'exécution.

1. On doit être capable d'extraire le chiffrement du mot de passe du fichier image.
2. Il est important de transférer correctement les paramètres environnementaux nécessaires à la fonction de décryptage dans l'ordinateur forensique.

Sur la base de l'analyse ci-dessus, ont proposé une méthode pour obtenir le mot de passe du système loggé en texte clair à partir du fichier image en mémoire Windows.

L'idée de la méthode est décrite comme suit:

1. Récupérez le fichier "lsasrv.dll" depuis le fichier image de la mémoire de l'ordinateur cible.
2. Analyser le texte chiffré du mot de passe et sa longueur.
3. Assurez-on que la fonction de déchiffrement fonctionne parfaitement dans l'ordinateur forensique.
4. Fournissez le texte chiffré et sa longueur comme paramètres et exécutez la fonction de décryptage.

L'analyse de texte chiffré des mots de passe et la fonction de décryptage de l'ordinateur forensique sont basées sur le déchargement de la DLL relative dans le fichier image de la mémoire physique.

III.3.1.3 Limites des performances de la méthode

- On ne peut pas récupérer le texte du mot de passe en clair si l'utilisateur ne s'est pas identifié dans le système. Cette méthode n'est pas nécessairement d'obtenir le texte en clair du mot de passe en raison des restrictions sur la Rainbow table.

- Le taux de réussite est influencé par les facteurs suivants:

    1. La première page du "lsasrv.dll" commuté en mémoire à l'extérieur ne sera pas trouvée dans la mémoire si la distance entre le temps légal et le temps de connexion est trop longue, donc on ne peut obtenir les données du ".data" et le fichier vidé "lsasrv.dll" en entier.

    2. Les données pointées par des pointeurs importants seront mises en mémoire à l'extérieur, donc on ne peut pas trouver la variable requise dans le fichier image de la mémoire.

Un travail supplémentaire est nécessaire pour déterminer si les idées développées sur Windows 7 sont adaptées à Windows XP.

III.3.2 Une GUI pour mémoire volatile

La volatilité est actuellement fournie avec une interface de ligne de commande uniquement, ce qui peut constituer un obstacle pour certains investigateurs lors de l'utilisation de l'outil.

[**Marko et al., 14**], ont proposé une interface graphique et des extensions pour le Volatility Framework, qui d'une part simplifie l'utilisation de l'outil et d'autre part offrent des fonctionnalités supplémentaires comme le stockage des résultats dans une base de données.

III.3.2.1 Limites CLI pour la mémoire volatile

- Bien qu'un CLI offre beaucoup de flexibilité et soit la forme d'interface préférée de nombreux informaticiens, il limite l'utilisabilité de l'outil.

- Cet ensemble d'outils précieux pourrait ne pas être utilisé par autant d'enquêteurs qu'il le mériterait.

- La version actuelle de Volatility est ses possibilités limitées de corrélation des données extraites et de sauvegarde des résultats d'extraction.

III.3.2.2 Avantage GUI pour la mémoire volatile

Ce document présente une interface utilisateur graphique (GUI) et des extensions pour Volatility Framework, qui aideront à résoudre les problèmes d'utilisabilité décrits ci-dessus.

- Fournit une interface simple et facile à installer qui remplace l'interface CLI, ce qui peut être un obstacle à l'utilisation de la Volatilité par les investigateurs.
- Il est maintenant possible de sauvegarder (une partie des) résultats d'extraction dans une base de données et de les charger à partir de celle-ci.
- Des séquences complexes de commandes peuvent être réduites à un simple clic de bouton.
- L'outil pouvant être facilement étendu, il est également à l'épreuve du temps en ce qui concerne les nouveaux plugins ou même les nouvelles toolkits.

III.3.3 Exploration des mémoires a décharge pour les MV

[**Lei et al., 14**], ont proposé une sorte de moyens et méthodes efficaces de la machine virtuelle en temps réel forensics machine virtuelle, dans un premier temps pour sauvegarder la mémoire ou le processus en cours de machines virtuelles sur l'hôte physique, puis analyser le fichier mémoire de sauvegarde.

III.3.3.1 Conception d'algorithme

Les auteurs [**Lei et al., 14**], on a proposé un algorithme de décharge de mémoire et d'analyse forensique basé sur la machine virtuelle comprend principalement trois modules suivantes :

- Le module de rechercher tous les processus en cours d'exécution en parcourant sous le système Linux et déterminez le numéro PID du processus du système virtuel.
- Le module de décharge mémoire complète du processus du système virtuel dans un fichier.
- Le module d'analyse de mémoire pour analyser et légaliser le fichier de mémoire de décharge, on peut obtenir des preuves des informations de la machine virtuelle, telles que des informations sur le processus, le réseau, l'utilisateur, etc.

Le schéma fonctionnel de l'algorithme de la figure 16, tel qu'illustré.

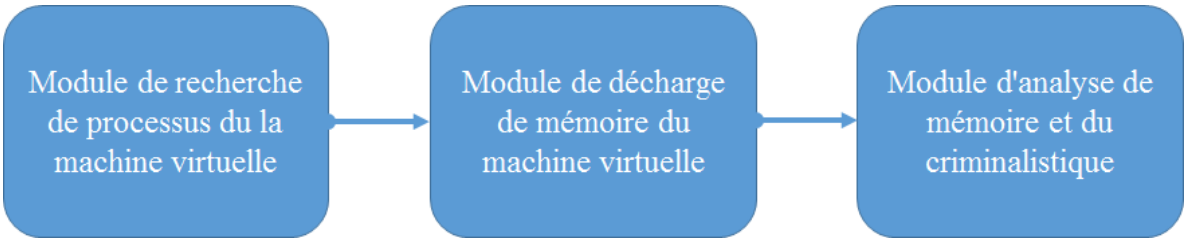


Figure 16. Diagramme de l'algorithme [Lei et al.,14].

III.3.3.2 Limites des performances de la méthode

Le principal inconvénient de faire un dump de mémoire machine physique est le fait que l'utilisateur change d'état d'ordinateur et peut facilement faire des étapes qui peuvent mener à une corruption irrémédiable des preuves.

III.3.4 Extractions de preuve basée mémoire

C'est devenu un nouveau défi concernant les attributs volatils de l'information et les données générées par la version Web de la messagerie instantanée.

Les auteurs [**Chang et al.,13**], ont proposé une procédure d'analyse de la mémoire physique pour les enquêtes sur les éléments de preuve clés lors de l'utilisation de la messagerie instantanée.

III.3.4.1 Mémoire forensique dans notre schéma

Dans notre schéma, les auteurs [**Chang et al., 2013**], ont proposé nombreux avantages sont démontrés comparé aux méthodes criminalistiques traditionnelles.

- La mémoire forensic est plus efficace comparée à l'ordinateur forensics traditionnel.
    - Le dumping de mémoire est la première étape pour obtenir les données complètes de mémoire et cela prend moins de temps que le dumping de toute la mémoire statique.
- La méthode de criminalistique n'est pas compliquée.
    - On essaye de trouver des preuves-clés en cherchant des mots-clés connexes dans des logiciels de forensique tels que MadEdit et WinHex.

- Ces logiciels forensiques exigent une petite taille dans le disque dur et sont faciles à manipuler avec des interfaces utilisateur simples.

- L'adresse mémoire et le contenu peuvent être affichés clairement dans les logiciels d'investigation, ce qui est pratique pour les investigateurs de capturer l'écran du moniteur et de recueillir des preuves par clé.

Différents éléments de preuve ont pu être trouvés en raison de différentes situations de cas, comme le montre le TABLEAU 2. Objectif est de récupérer les enregistrements de conversations. Si les investigateurs ont des enregistrements de conversation en main, ceux-ci affectent beaucoup la décision du tribunal.

On utilise le logiciel de messagerie instantanée commun "Skype" et la version Web de "Facebook Messenger" comme cas cibles.

	Case 1 in Skype	Case 2 in Facebook
Evidence of user's login ID	V	V
Evidence of user's login password	V	V
Get user's contact list	V	X
Evidence of user's conversation records	V	V
Evidence of the parties who join the conversation	V	V

Tableau 2. Les preuves obtenues de la mémoire forensique dans les cas 1 et 2.

V: récupération || X : non disponible

III.3.5 Modèle de Comparaison

III.3.5.1 Table de comparaison

Phases du Framework		Approches			
		[Xu and Wang, 13]	**[Chang et al., 13]**	**[Marko et al., 14]**	**[Lei et al.,14]**
PROCESSUS D'INVESTIGATION	Acquisition de la mémoire	✖	✖	✖	✖
	Analyse de la mémoire	✓	✓	✓	✓
	Génération de rapports	✓	✓	✓	✓
MÉTHODES D'INVESTIGATION	L'analyse réponse en direct	✖	✖	✖	✓
	L'analyse d'images	✓	✓	✓	
MÉTHODES D'ANALYSE LA MÉMOIRE	Recherche de chaîne		✓	✖	✖
	Recherche d'objet de processus	✓	✖	✓	✓
	Recherche de signature de fichier	✖	✓	✖	✖
	Sculpture de fichier	✓	✓		
TYPE DE PREUVE RÉCUPEREÉ	Temps Système	✓	✓	✓	✓
	Connexions Réseau		✓	✓	✓
	Historique des commandes	✖	✖	✖	✓
	Processus Actifs	✓	✓	✓	✓
COLLECTION DE DONNÉES	Mémoire physique	✓	✓	✓	✓
	Fichier hibernation	✓	(-)	(-)	(-)
	Fichier page	✓	(-)	(-)	(-)
Récupération des données		Taux de Réussite 81,1%	Plus Facile	(-)	Plus Difficile et Plus Longue
Impact sur les données volatiles		Inférieure	Supérieur	Supérieur	Inférieure
L'impact sur le système à l'étude		Faible	Faible	Faible	Élevé
Perte de preuves volatiles		Faible	Faible	Faible	Élevé

Tableau 3. Comparaison entre les approches existantes.

III.3.5.2 Synthèse des travaux existants

Après avoir étudier chaque approche et établi le tableau comparative ci-dessus, on peut noter les limites suivantes :

- L'impact de ces méthodes de réponse en direct sur le système à l'étude peut être significatif.
- La majorité des approches ne couvre pas toutes les phases de L'investigation numérique.
- La plupart des approches ne prennent pas en compte la vie privée des utilisateurs.
- En raison de l'augmentation significative du volume de données numériques, la tâche d'investigation devient fastidieuse et ennuyeuse.
- Les outils d'investigation disponibles sont principalement dédiés à l'analyse statique des données numériques.

Mike Paquette, VP des produits chez Prelert, a également cité que l'apprentissage machine est la réponse de la cyber-sécurité à la détection des violations avancées [Mch15]. Si l'on peut développer une solution qui combine l'apprentissage machine et l'outil, ce sera une excellente solution.

III.4 Applications connexes

Les outils présentés ci-dessous sont les plus largement utilisés et font partie des produits que les tribunaux reconnaissent dans les investigations où les médias numériques sont analysés.

III.4.1 EnCase

EnCase est l'outil d'investigation numérique le plus connu qui permet la collecte, l'identification, l'analyse et la création de rapports efficaces et fiables sur le plan judiciaire, de manière reproductible et défendable [Muh18].

C’est un logiciel très utilisé par les départements d’investigation numérique de la police et la gendarmerie. Il constitue une des seules suites logicielles reconnues par les tribunaux pour la recherche de preuves. Il supporte les analyses par mot clé, par comparaison d’empreintes et implémente également son propre langage de scripting permettant ainsi le développement de ses propres modules pour des analyses personnalisées.

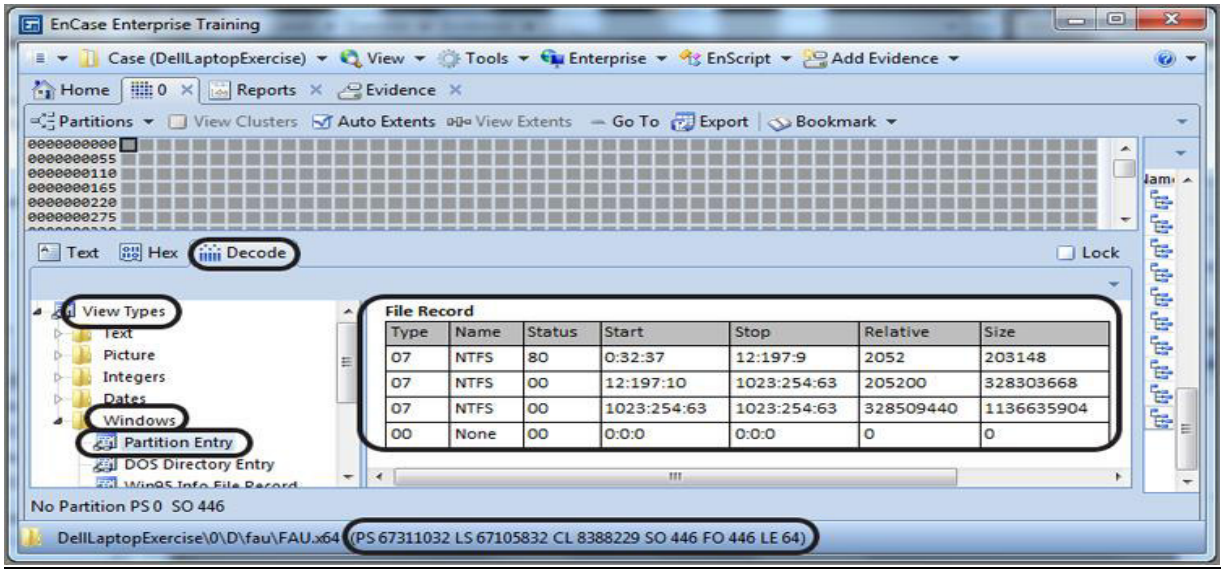


Figure 17. Interface d'EnCase.

III.4.2 FTK (Forensic ToolKit)

FTK (Forensic ToolKit) a été développée par AccessData Corporation [Acc18]. C'est un outil commercial bien connu pour l'analyse des disques et de la mémoire sur les ordinateurs Windows. Elle fonctionne uniquement sur Windows. Elle est livrée avec un moteur de base de données PostgreSQL qui indexe toutes les données présentes sur le support et réduit considérablement le temps de réponse des analyses. FTK est également capable d'analyser des images créées à partir d'autres outils comme EnCase.

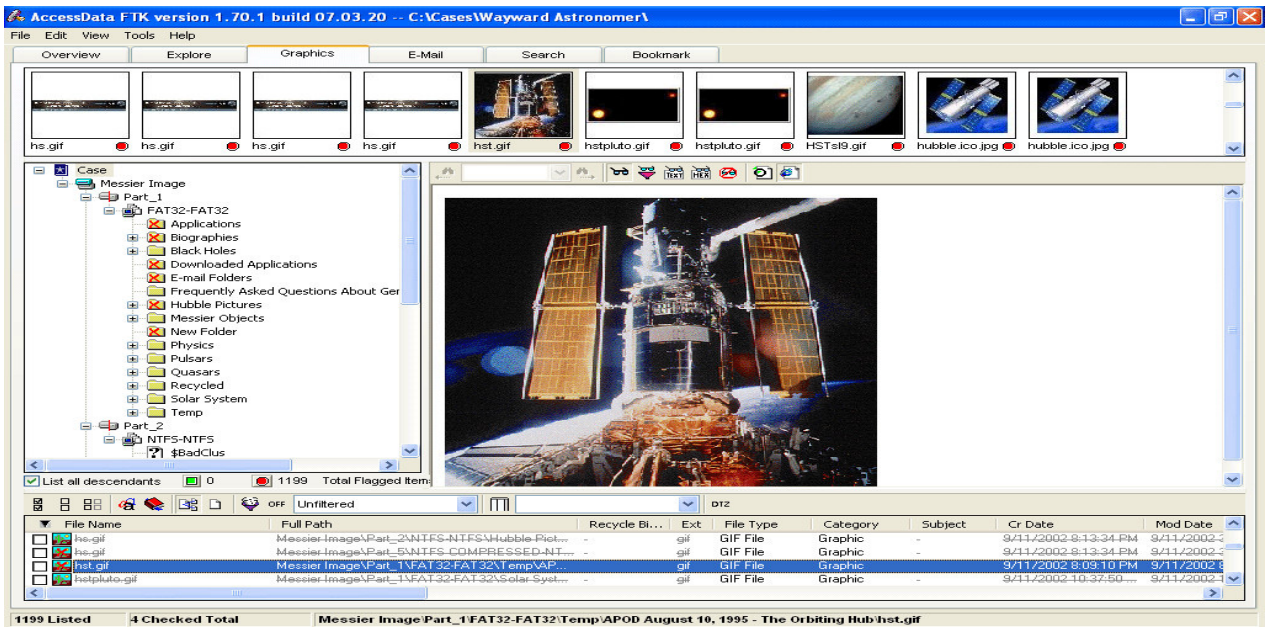


Figure 18. Interface de FTK.

III.4.3 Autopsy

Autopsy est un programme d'investigation numérique open source basé sur une interface graphique permettant d'analyser efficacement les disques durs et les téléphones intelligents.

Autospy est utilisé par des millions d'utilisateurs dans le monde pour investiguer ce qui s'est passé sur l'ordinateur. Il a une architecture de plug-in qui on permet de trouver des modules additionnels ou de développer des modules personnalisés en Java ou Python. [Muh2018].

Certains des modules fournissent une analyse chronologique, la recherche de mots-clés, la sculpture de données, l'indicateur de compromis à l'aide de STIX. On peut même l'utiliser pour récupérer des photos de la carte mémoire de notre appareil photo.

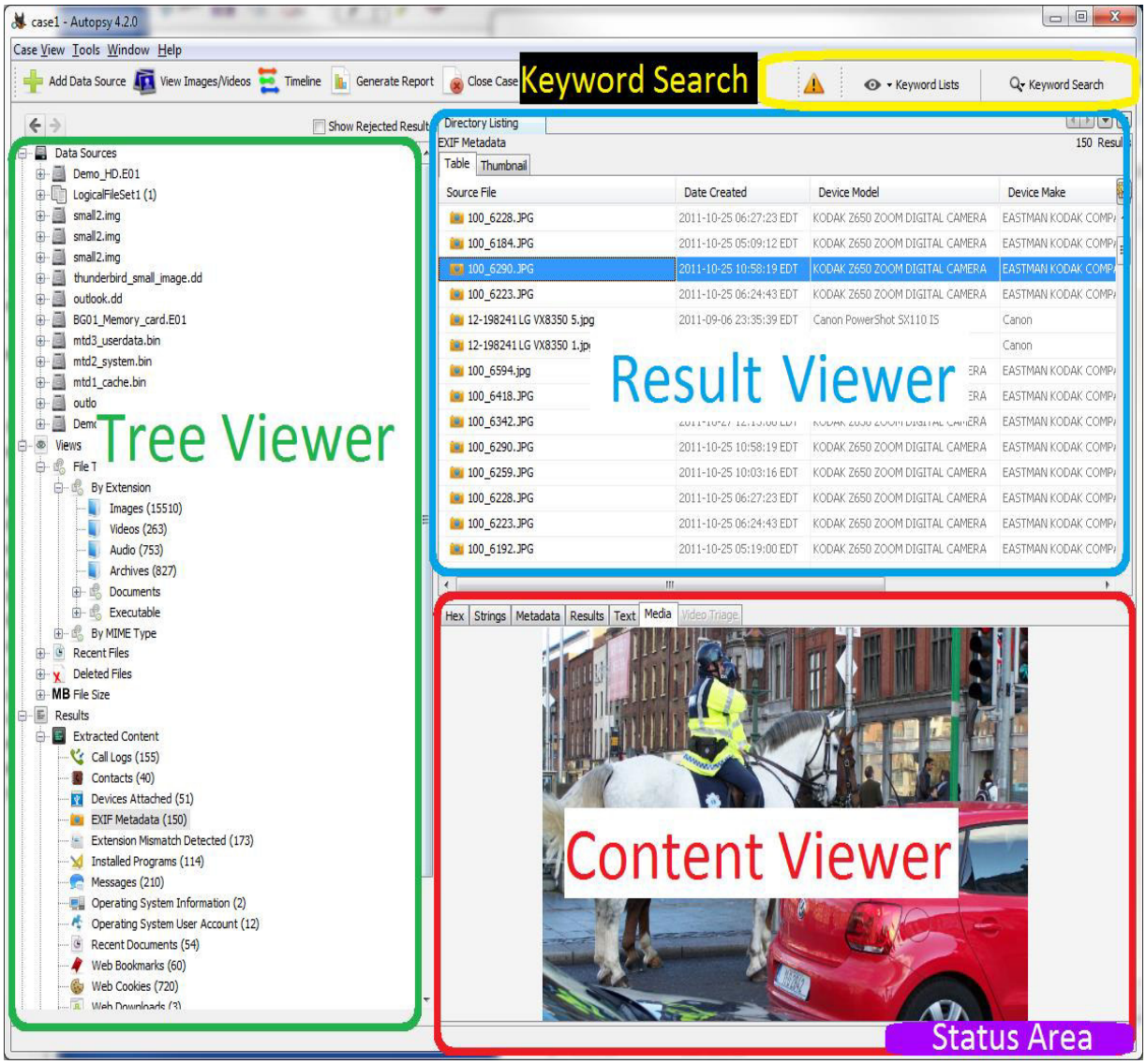


Figure 19. Interface Autopsy.

III.4.4 Etude comparative et synthèse

III.4.4.1 Comparaison des outils basés sur leurs fonctionnalités

	FTK	Autopsy	EnCase
Licence de logiciel	Commercial	Freeware	Commercial
Support de plate-forme	Windows 7 Windows 8, Windows 10, Windows Server	Windows7,WindowsVista ,Windows XP, Linux, Unix	Linux, Mac, Windows, Solaris
Développeur	AccessData	Brian Carrier	Guidance Software
Performance	Élevée	Logiciel de Faible Qualité	Élevée
Coût	Coûteux	Logiciel open source	Coûteux
Tâches principales	Identification, Imagerie, Analyse, Rapport	Acquisition de données, Analyse, Sculpture de données	Identification des Données, Acquisition, Analyse, Documentation, Rapport

Tableau 4. Comparaison entre les outils existants.

III.5 Conclusion

Ce chapitre a été consacré à soulignera également plus de détails en comparant les différentes approches et en parvenant ensuite à l'état actuel de l'investigation numérique de la mémoire. On a présenté les logiciels les plus utilisés par les départements d'investigation numérique de la police et la gendarmerie et on a obtenu un tableau de comparatif qui donne un aperçu de chaque outil en se basant sur les fonctionnalités.

IV.1 Introduction

Quand un crime numérique se produit, l'objectif principal des experts légistes est d'acquérir et d'analyser des données non volatiles des machines suspectes. Mais cette approche n'est plus exhaustive.

Après avoir présenté les divers travaux connexes sur ce sujet dans le chapitre précédent. On a proposé un nouveau modèle d'investigation numérique pouvant couvrir tous les cas, de la scène du crime au laboratoire, jusqu'à la présentation du cas.

Dans ce chapitre, on va présenter l'architecture globale de notre système, ainsi que le détail de chaque composant de cette architecture, puis on développera une modélisation détaillée avec 'UML' dans laquelle la structure globale du système est fixée.

IV.2 Conception générale du système proposé

IV.2.1 Architecture globale

La visée principale de cette section est de concevoir une architecture générale afin de choisir le modèle le plus approprié pour une tâche d'investigation numérique.

Notre système se compose de :

- **Les victimes de fraude numérique**
- **Agence d'investigation numérique**
- **L'équipe d'investigation numérique**

En général, les membres de l'équipe d'investigation se répartissent en trois groupes :

- **Spécialiste légiste**
- **Investigateur légiste**
- **Examinateur légiste**

En stockage : On utilise le **Cloud Privé.**

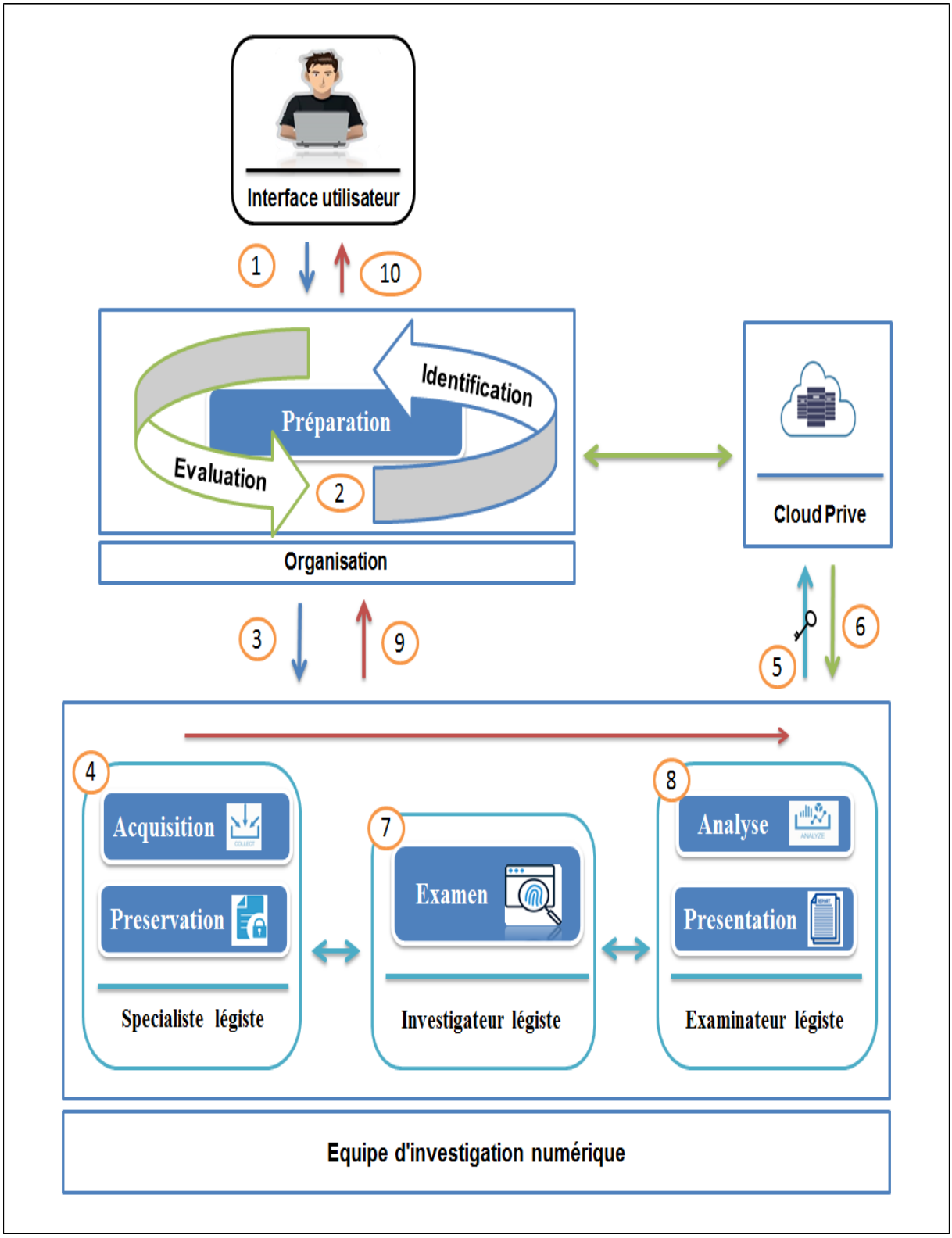


Figure 20. L'architecture générale du système.

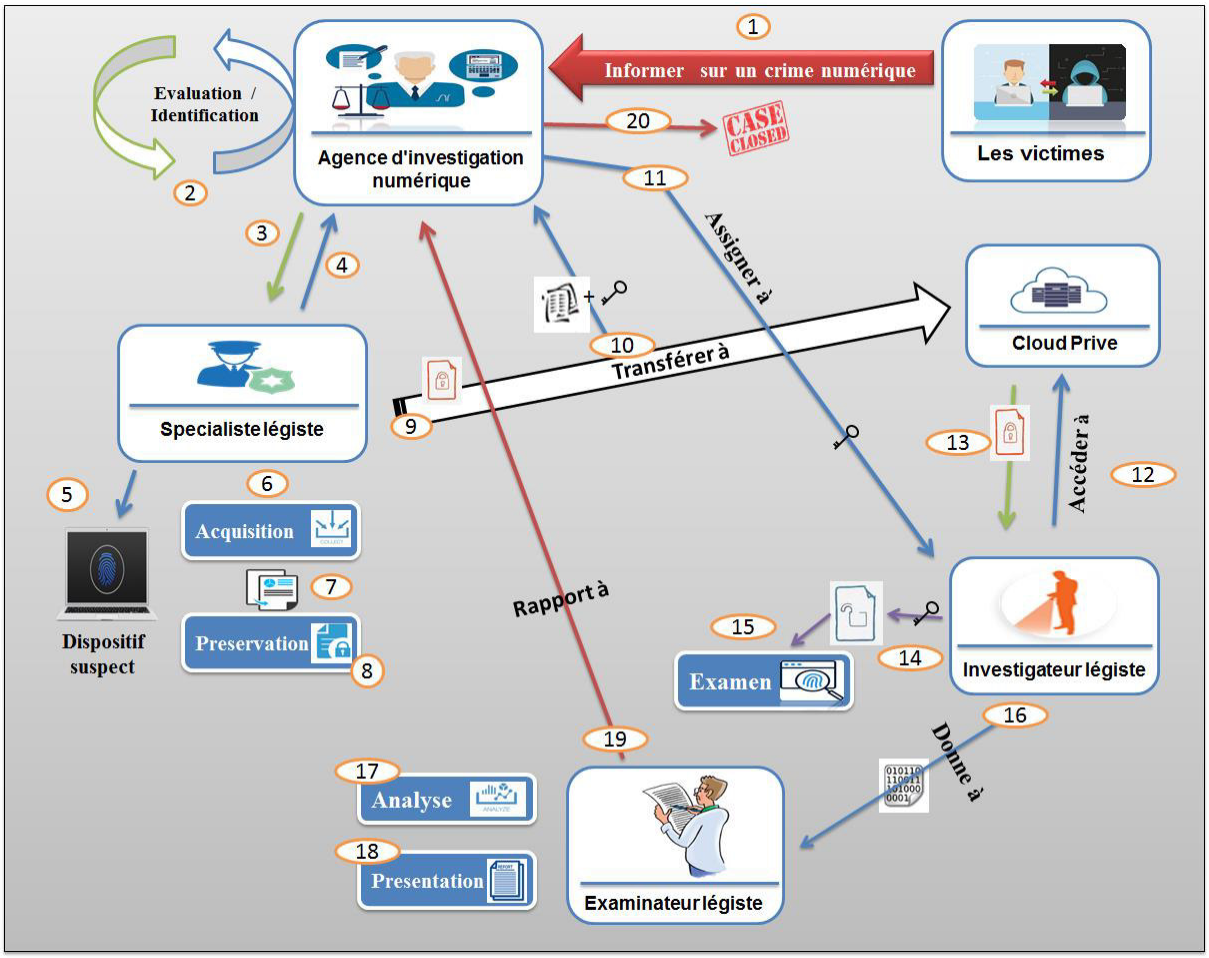


Figure 21. Scénarios de l'étude.

Description du système proposé :

Notre architecture est composée d'une collection de composants afin de trouver le modèle le plus approprié pour une tâche d'investigation de mémoire numérique, à cet égard on a utilisé un ensemble de composants :

La composante de l'agence d'investigation numérique inclut la phase de préparation, qui comprend deux processus: l'évaluation et l'identification des enquêtes numériques. Et on a l'équipe d'investigation numérique est divisée en trois groupes: Spécialiste légiste, Investigateur légiste, Examinateur légiste. Au moins chaque composante comprend une phase essentielle du processus d'investigation de la mémoire numérique. Et on a le composant stockage et fiabilité du cloud.

Toutes ces composantes travaillent ensemble pour assurer le modèle le plus approprié pour une tâche d'investigation numérique sur la mémoire physique. Cette section est consacrée à une explication détaillée de chaque étape du processus.

IV.2.2 Architecture détaillée

IV.2.2.1 Le composant de l'agence d'investigation numérique

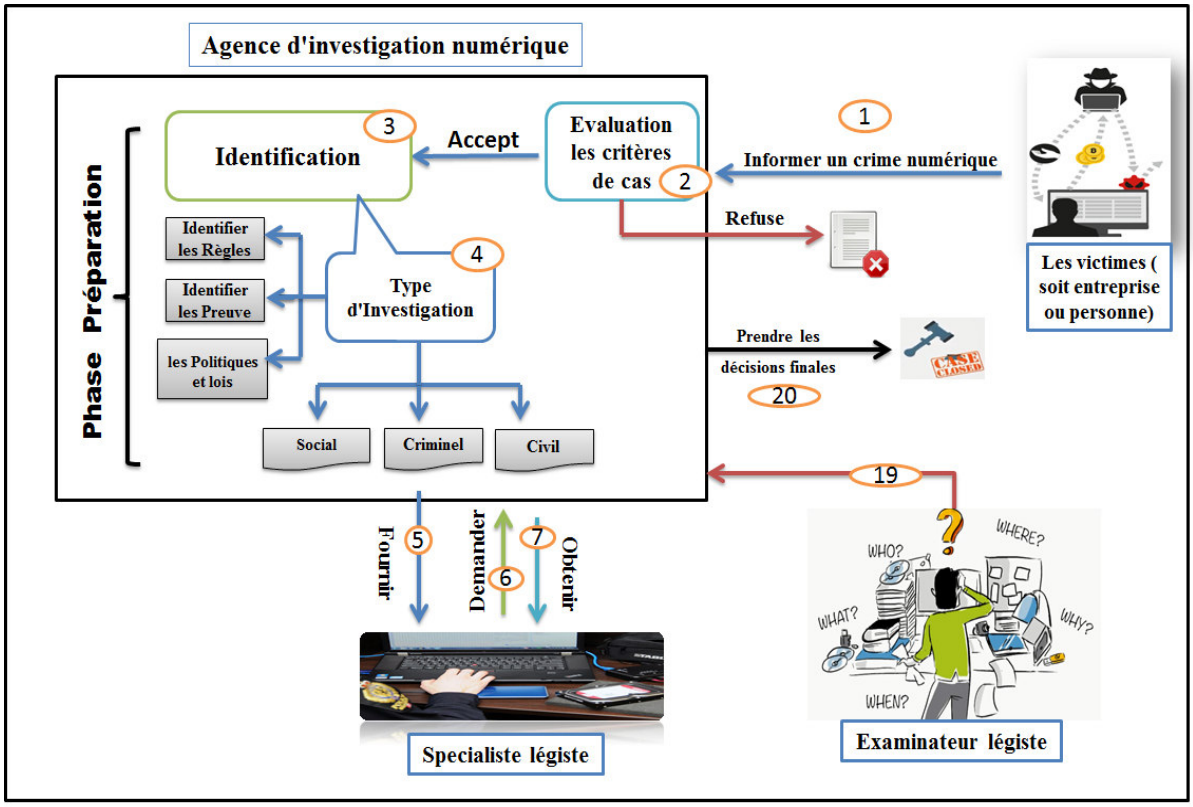


Figure 22. L'architecture des tâches L'agence d'investigation numérique.

Le rôle du composant de l'agence d'investigation numérique

- L'évaluation et l'identification des enquêtes numériques
- Prendre les décisions finales

Phase principale du composant:

Phase Préparation :

La première étape dans les procédures de traitement des incidents, qui doit être appliquée avant de traiter toute investigation.

- Evaluation initiale du cas : déterminer les exigences relatives au cas.
  - Situation de l'affaire.
  - Nature de l'affaire.
  - Précisions sur l'affaire.

- Identifier les critères d'acceptation des cas et s'en tenir à ces critères.
- Identifier le type d'investigation
  - Civil
  - Criminel
  - Social
- Fournir les informations nécessaires sur l'affaire
- Donner l'autorisation requise sur l'affaire
- Préparer une fiche administrative et une description narrative sur l'affaire

IV.2.2.2 Le composant Spécialiste légiste

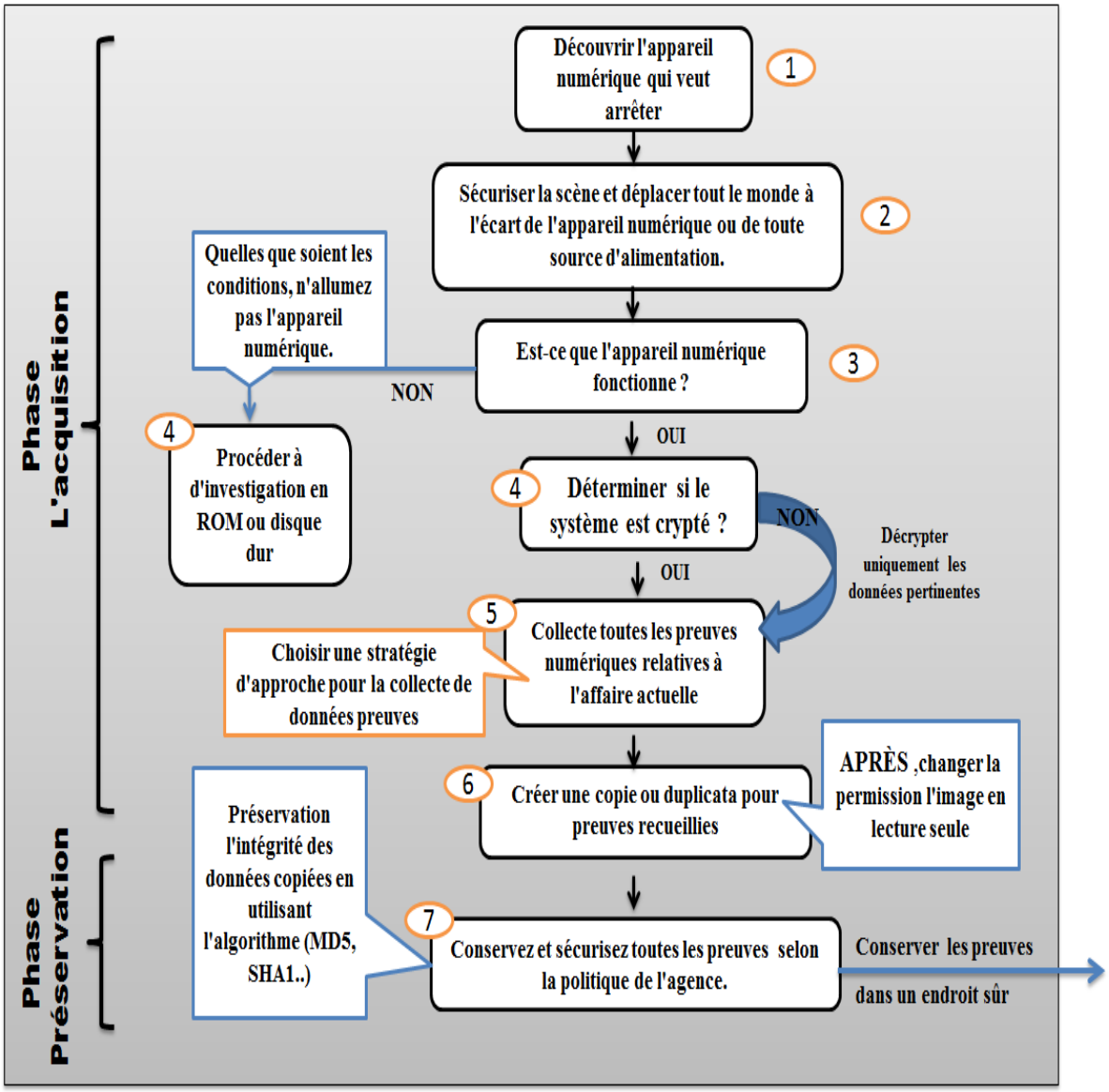


Figure 22. 1. L'architecture des tâches Spécialiste.

Le rôle du composant spécialiste légiste:

Le Spécialiste légiste est celui qui effectue en premier lieu toutes les actions de saisie de données pour recueillir des preuves au début du processus de criminalistique.

- Appliquer des techniques de capture d'images pour collecter des données à partir de périphériques de stockage.
- Appliquer une chaîne de contrôle appropriée aux preuves légistes recueillies.
- Encrypte les données pertinentes selon la clé de l'administrateur.

Deux phase principale du composant:

Phase 1 : L'acquisition de la mémoire

L'acquisition est la deuxième étape du processus judiciaire et est essentielle pour assurer l'intégrité de la preuve.

- La partie la plus technique de l'investigation et peut être le moment le plus critique pour faire des erreurs.
- L'acquisition est le processus de collecte de toutes les preuves numériques relatives à l'affaire actuelle.
- Créer une copie numérique à l'aide de méthodes et de techniques légistes solides.

Phase 2 : Préservation de la mémoire

Préservation est l'étape essentielle pour sécuriser et évaluer la preuve numérique relative à l'affaire actuelle.

- On n'a pas modifié les preuves pendant l'acquisition (chaîne de possession).
- La protection des preuves contre la destruction intentionnelle par des pirates ou la modification accidentelle par du personnel non formé.
- Calculer une empreinte numérique qui identifiera le fichier à l'aide d'une fonction de hachage (MD5, SHA1, …).

IV.2.2.3 Le composant Investigateur légiste

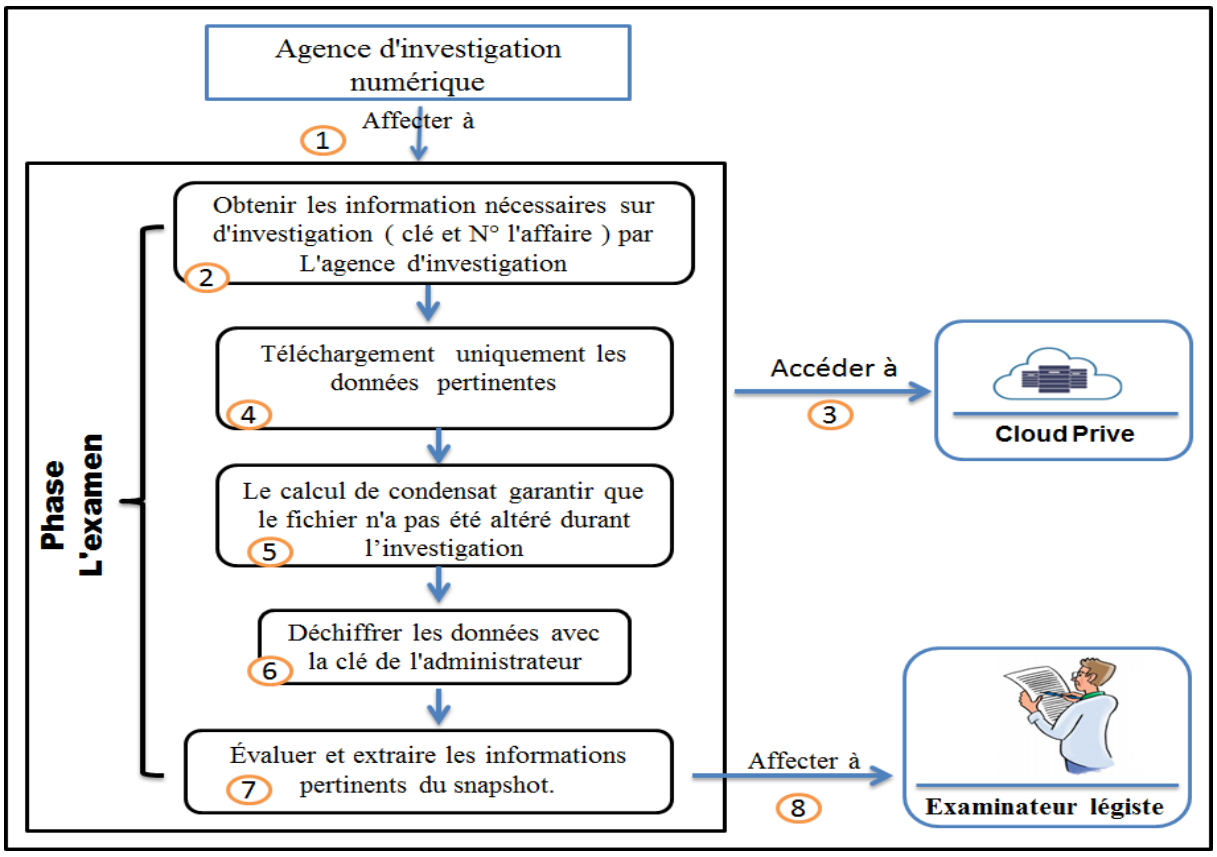


Figure 22. 2. L'architecture des tâches investigateur légiste.

Le rôle du composant investigateur légiste:

- Responsable de l'examen des preuves recueillies par le Spécialiste légiste.
- Déchiffrer les données avec la clé de l'administrateur
- Assurez-vous que les données copiées sont une copie exacte des données d'origine.

Phase principale du composant:

Phase l'examen de mémoire :

- Le processus implique un traitement méthodique et logique des données.
- Évaluer et extraire les éléments d'information pertinents du Snapchat.
- Assurer leur exactitude et leur fiabilité les outils de criminalistique.
- Le calcul de condensat garantir que le fichier n'a pas été altéré durant l'investigation.

IV.2.2.3.1 Calcul de condensat

Lors d'une analyse forensic, il est primordial de calculer une empreinte qui identifiera le fichier à l'aide d'une fonction de hachage. Cette empreinte doit être unique car elle permet de valider que le fichier n'a pas été altéré durant l'investigation.

Le calcul de condensat permettra ainsi de garantir l'intégrité des fichiers analysés.

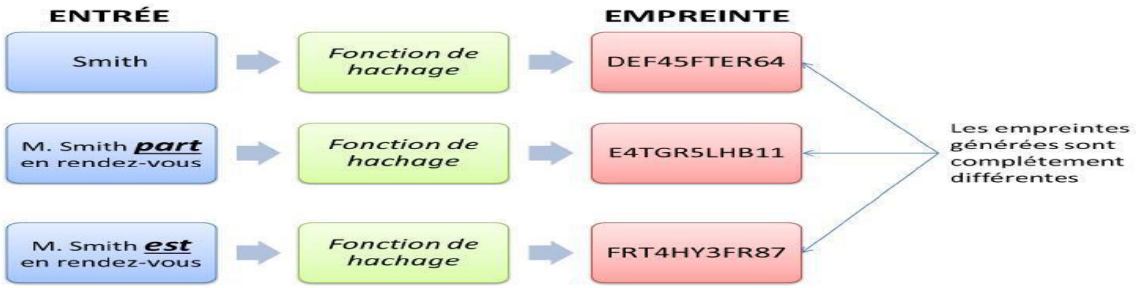


Figure 22.2.1. Principe de fonctionnement d'une fonction de hachage sur 3 entrées différentes.

IV.2.2.4 Le composant Examinateur légiste

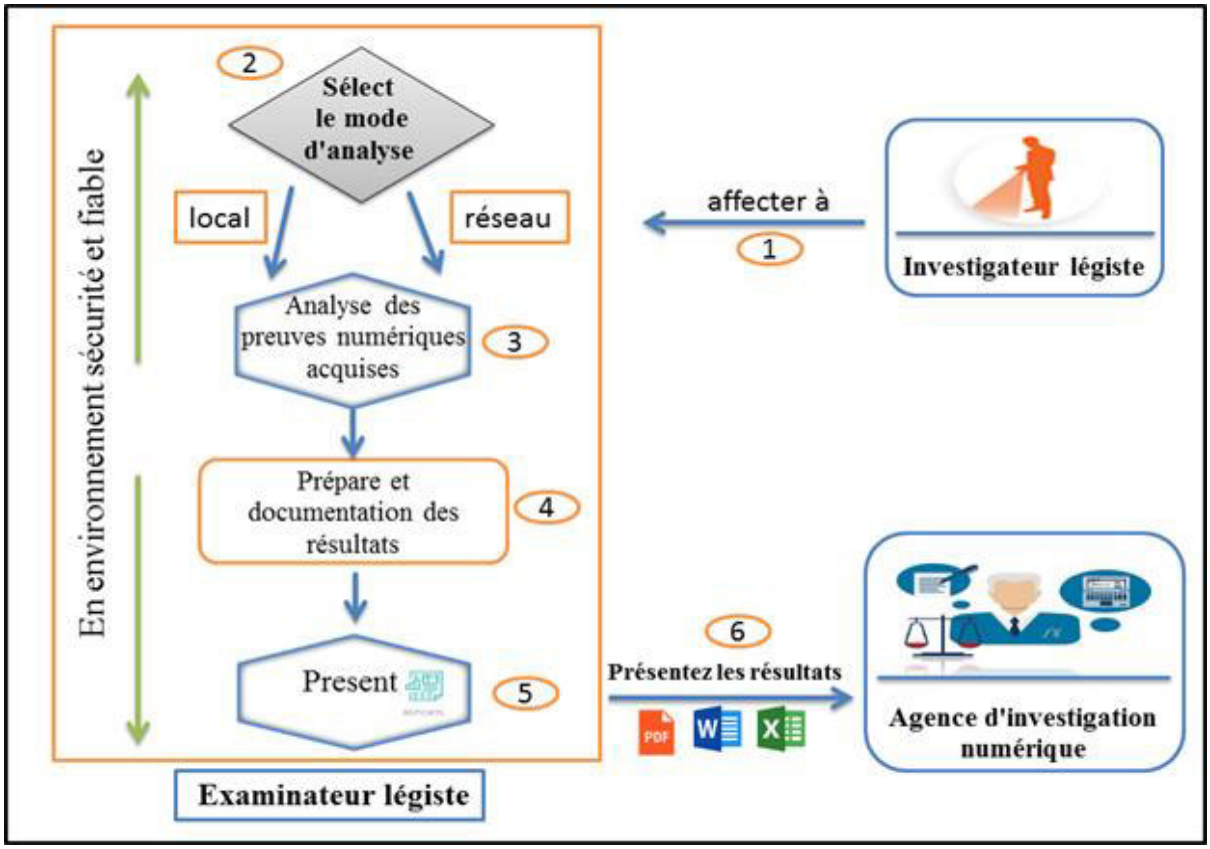


Figure 22. 3. L'architecture des tâches examinateur légiste.

Le rôle du composant Examinateur légiste:

L'examinateur légiste est responsable de l'analyse de la preuve, on fournit une évaluation professionnelle de la nature des preuves et de leur utilisation.

Deux phase principale du composant:

Phase 1 : L'analyse de mémoire

- Analyser l'information pertinente pour extraire des preuves d'un Snapchat donné par l'investigateur judiciaire.
- Analyse des artéfacts navigateurs Web et emails des utilisateurs.
- Analyse des registres d'événements et logs.
- Analyse de trace réseau.
- L'analyse doit être précise, complète, impartiale, enregistrée, reproductible et complète.
- Extraire les éléments qui constitueront le dossier final.

Phase 2 : Présentation (génération des rapports)

- Documenter toutes les phases précédentes et génère des rapports et des conclusions qu'il présente au jury ou aux unités de gestion respectives.
- Rapporter les bonnes réponses aux questions posées à l'examinateur à l'étape précédente.
- Le contenu d'un rapport varie selon la législation et les politiques locales.
- Création de rapport dans plusieurs formats disponibles tels que HTML, PDF, Word, Excel et autres.
- Il est courant que tous les rapports incluent :
    - Données du cas.
    - But de l'examen.
    - Constatations.
    - Conclusions.

IV.3 Conception et modélisation détaillée avec UML

IV.3.1 Diagramme de séquence

1. Diagramme de séquence de l'agence d'investigation numérique:

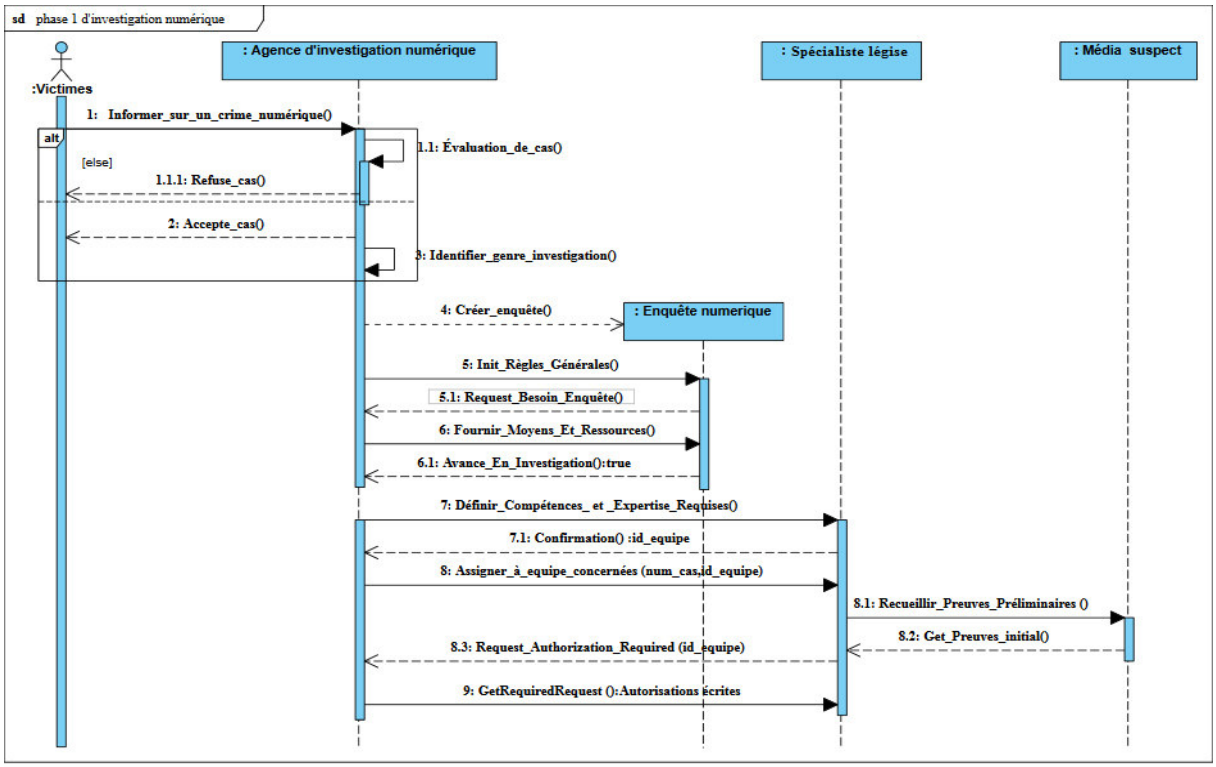


Figure 23. Diagramme de séquence des tâches de l'agence d'investigation numérique.

Description : La première tâche de l'investigation sur mémoire est l'évaluation et l'identification des enquêtes numériques

1. Les victimes de fraude numérique doivent fournir à Agence d'investigation numérique des preuves suffisantes qu'un crime a été commis.
2. L'agence d'investigation numérique doit déterminer les critères d'acceptation ou de refus de chaque cas individuel.
3. L'agence de préparation opérationnelle et d'infrastructures sont effectués, comprend la détection, la notification, la confirmation et l'autorisation.
4. L'agence définit les besoins de son équipe légiste, les compétences et l'expertise requises.
5. Confirme et autorise le Spécialiste légiste à mener une enquête de manière exhaustive sur l'incident et le lieu du crime.

2. Diagramme de séquence de Spécialiste légiste:

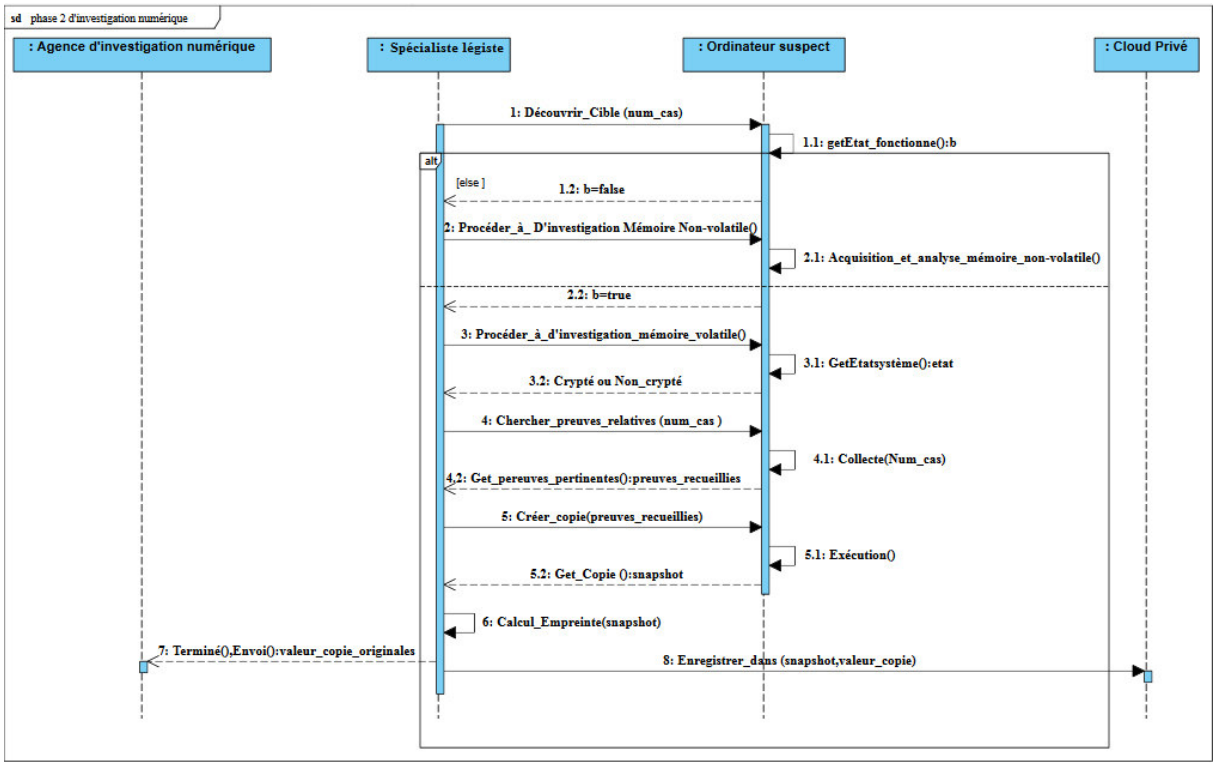


Figure 23. 1. Diagramme de séquence des tâches Spécialiste légiste.

Description : la deuxième tâche de l'investigation numérique est d'identifier les sources potentielles de preuves.

1. Après avoir identifié les sources de données potentielles par un Spécialiste légiste, on doit acquérir les preuves à partir de ces sources.

   ❖ L'acquisition des données doit s'effectuer en trois étapes:

   - Etablissement d'un plan d'acquisition de données
   - Acquérir suffisamment de preuves
   - Vérifier l'intégrité des données acquises

2. Le Spécialiste légiste doit stocker les preuves dans un endroit sûr Après avoir d'acquisition, dans notre cas en cloud.

3. Le Spécialiste légiste est chargé conservez et sécurisez toutes les preuves selon la politique de l'agence.

3. Diagramme de séquence de L'investigateur légiste:

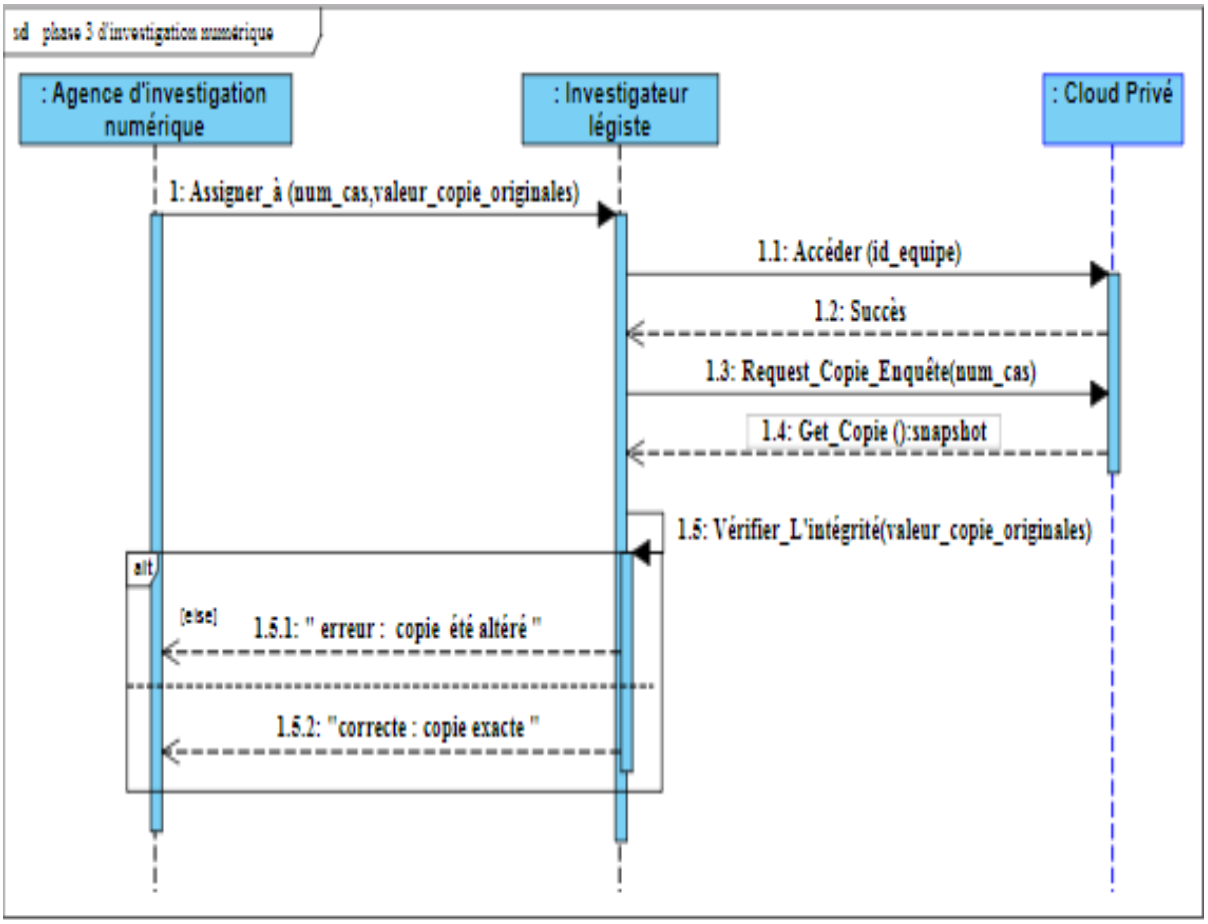


Figure 23. 2. Diagramme de séquence des tâches Investigateur légiste.

Description : la troisième tâche de l'investigation numérique est de vérifier l'intégrité des preuves originales et copiées.

1. Investigateur légiste doit vérifier que les données copiées sont une copie exacte des données originales.

2. Investigateur légiste doit évaluer et extraire les informations pertinentes des données recueillies.

4. Diagramme de séquence de l'examinateur légiste:

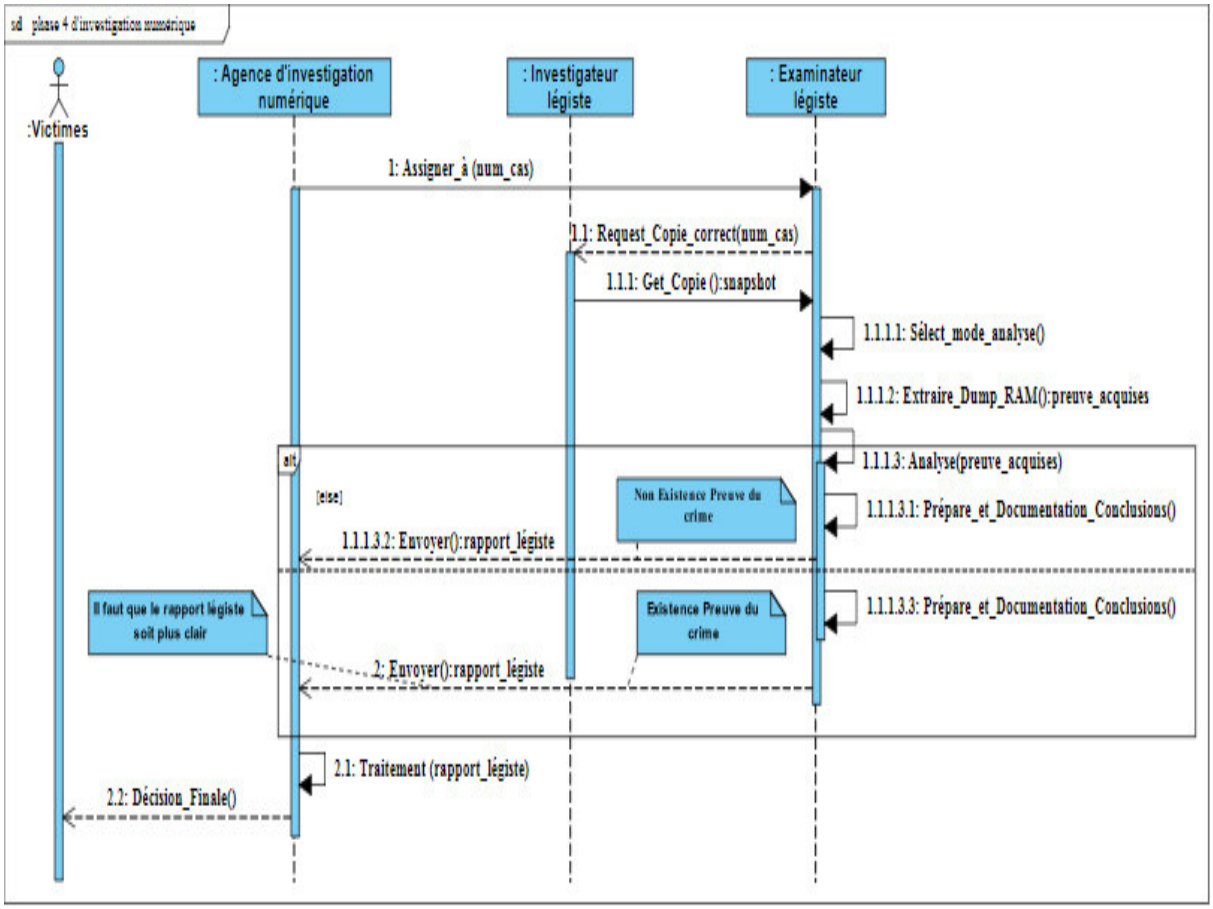


Figure 23. 3. Diagramme de séquence des tâches examinateur légiste.

Description : le quatrième tâche de l'investigation numérique est d'analyser les données pour en extraire des preuves.

1. les analystes judiciaires analysent les données pour extraire des preuves à partir d'un fichier d'images correct que fournit l'examinateur légiste.

2. L'examinateur légiste doit analyser en profondeur les preuves numériques acquises afin de rechercher les informations requises en fonction du type d'enquête. L'analyse est effectuée dans un environnement contrôlé, de préférence sur un ordinateur équipé des outils et programmes requis .

3. L'examinateur légiste devra clarifier le rapport judiciaire, afin que la personne moyenne puisse comprendre la preuve qui a été trouvée.

4. L'examinateur légiste génèrent et présentent des rapports et des conclusions au l'agence.

IV.3.2 Diagramme d'activité

1. Diagramme d'activité du le processus de préparation

Ce composant nous permet de configurer l'infrastructure appropriée et opérationnelle pour l'état actuel. C'est là que l'organisation se donne les moyens de traiter efficacement les différents types d'incidents.

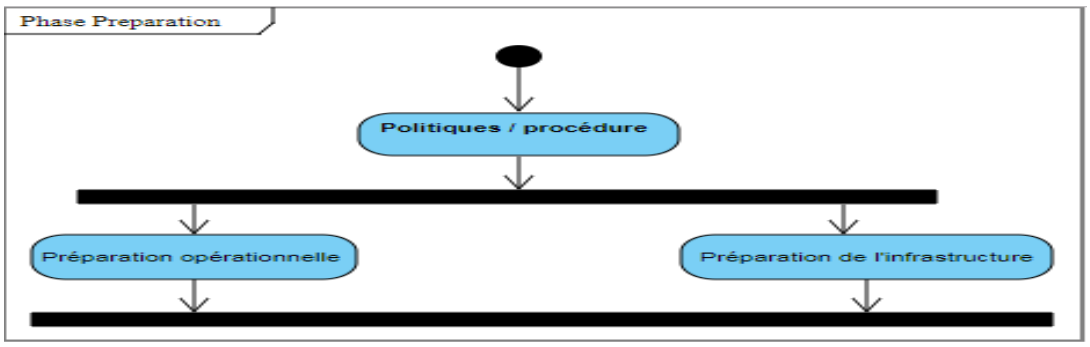


Figure 24. Diagramme des activités du processus de préparation.

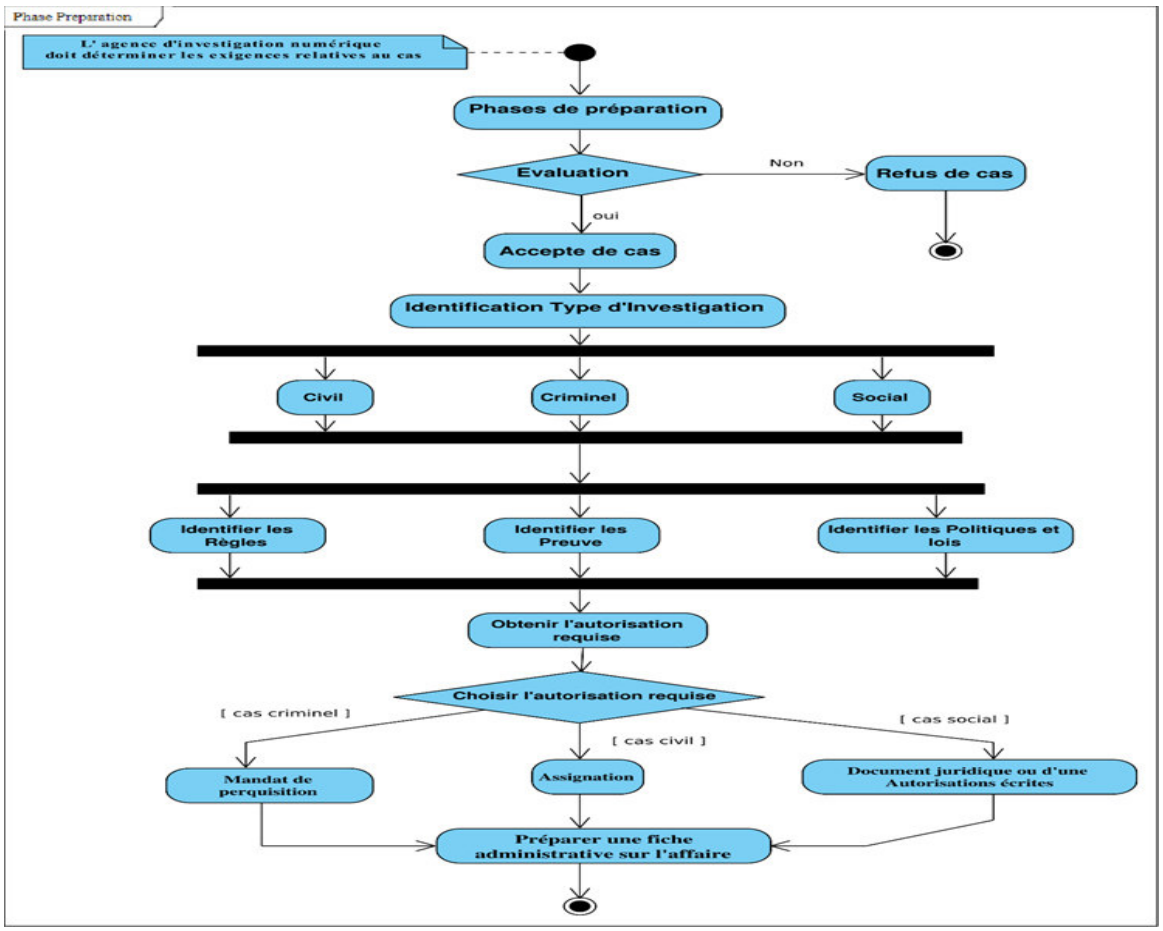


Figure 24. 1. Diagramme des activités effectuées par agence d'investigation numérique.

2. Diagramme des activités des processus d'acquisition et de préservation

Ce composant permet on identifie les dispositifs numériques pour détecter la présence de preuves numériques, le cas échéant, et on les recueille de telle sorte que l'état initial ou le contenu de la preuve reste le même sans altérer ou modifier la preuve.

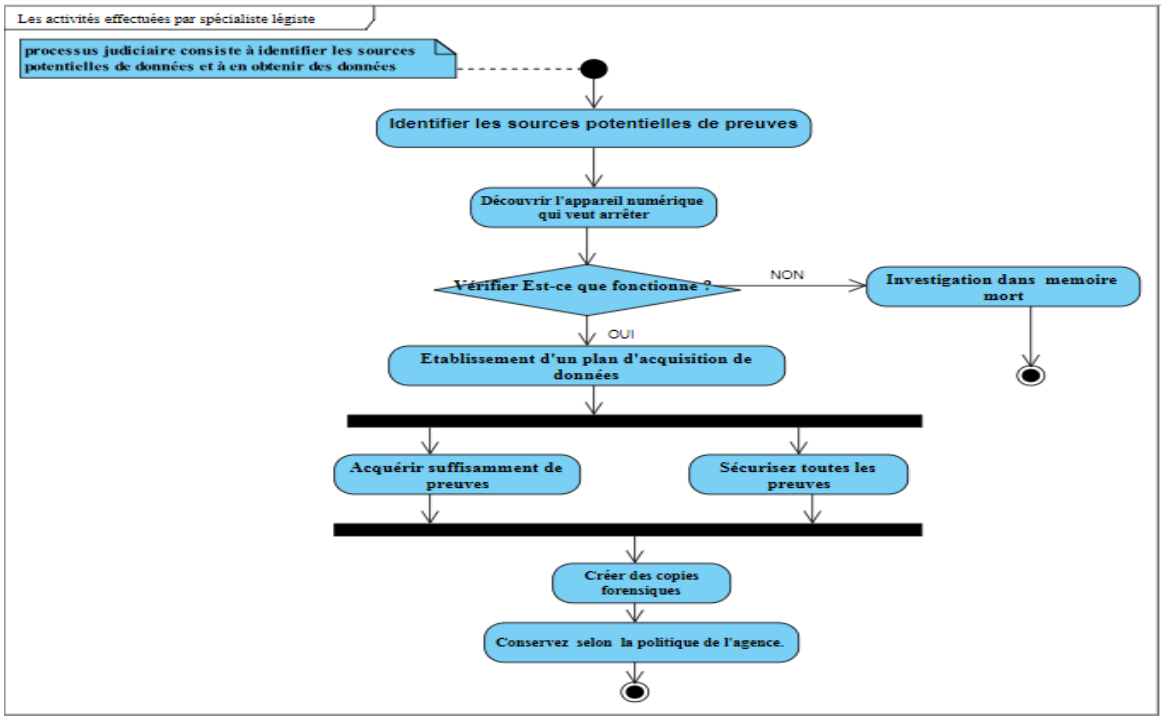


Figure 25. Diagramme des activités du processus d'acquisition et préservation.

3. Diagramme d'activité du processus d'examen

Ce composant nous permet d'examiner les données et de calculer leur valeur de hachage, d'exporter les données et d'extraire les métadonnées sans endommager aucune d'entre elles.

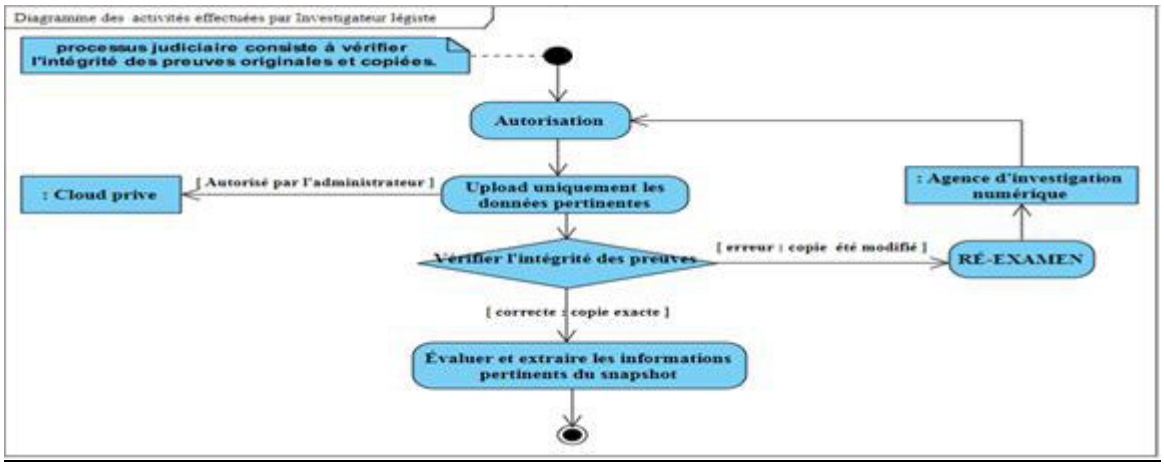


Figure 26. Diagramme des activités du processus d'examen.

4. Diagramme d'activité du processus analyse

Ce composant qui permet à analyser les données collectées et évaluer outils d'extraction de preuves et la création d'une image de ce qui a pu se produire.

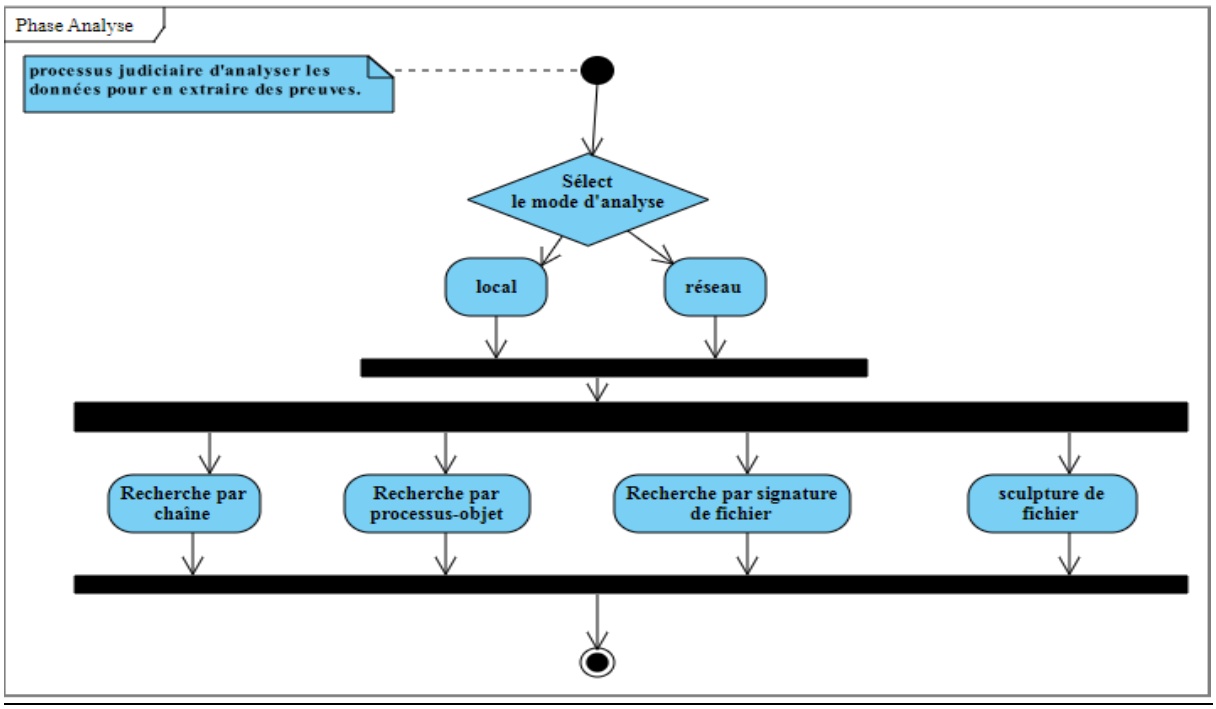


Figure 27. Diagramme des activités du processus d'Analyse.

5. Diagramme d'activité du composant Présentation

Cette phase présente les conclusions et les éléments de preuve obtenus de l'enquête criminalistique.

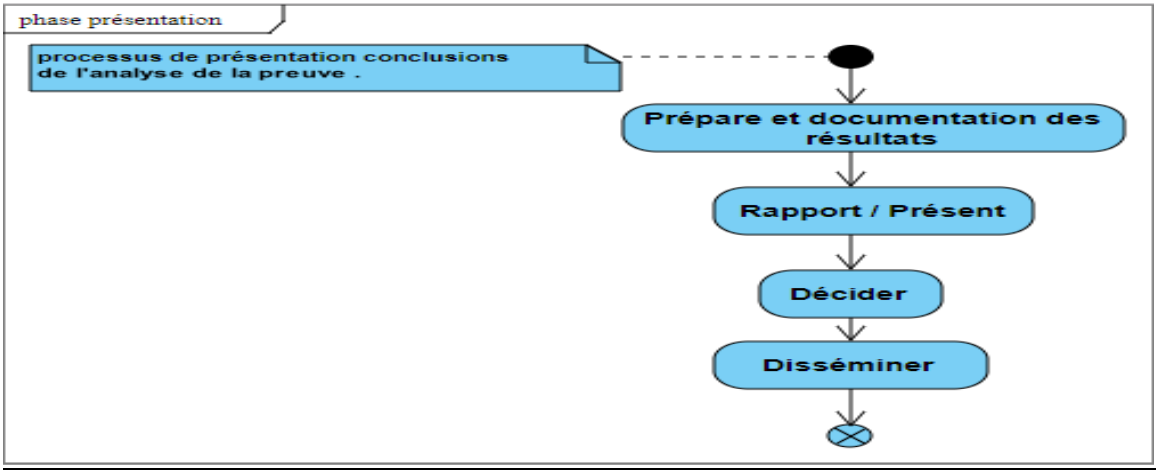


Figure 28. Diagramme des activités du processus de Présentation.

6. Diagramme des activités d'investigation numérique

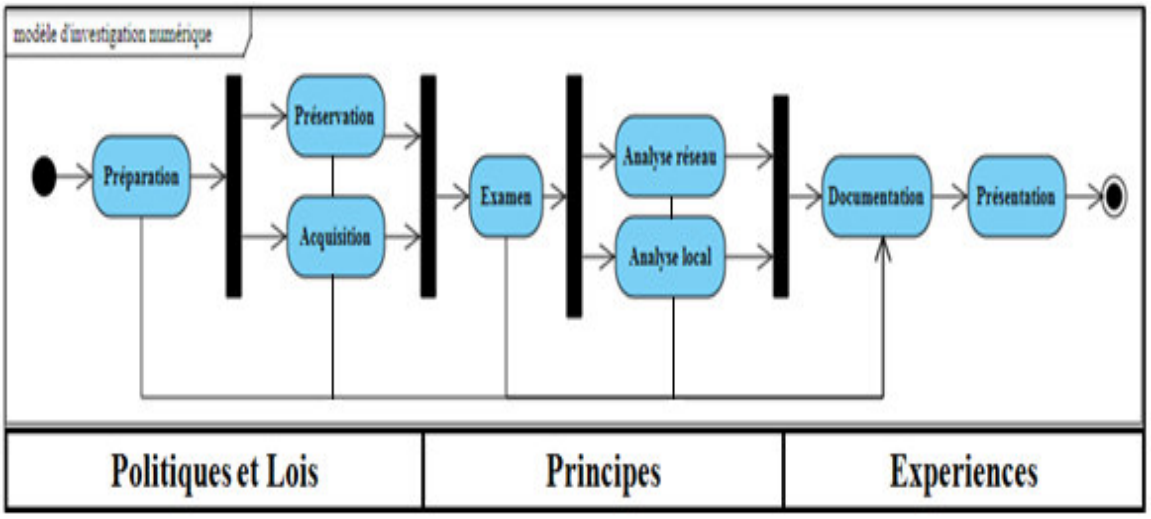


Figure 29. Diagramme d'activité du processus d'investigation numérique complet.

IV.4 Conclusion

Dans ce chapitre, on a présenté notre structure en considérant les avantages et les limites des modèles judiciaires, qui ont permis de choisir le modèle le plus approprié pour une tâche d'investigation numérique sur la mémoire.

V.1 Introduction

Dans ce chapitre, nous allons décrire la mise en œuvre des différentes étapes de notre système conçu dans le chapitre précédent. Nous commencerons par justifier l'environnement de développement utilisé, ensuite nous détaillerons les structures de données utilisées. Et enfin nous présenterons les fonctionnalités de notre application et nous illustrerons aussi un exemple réel dans un incident de criminalité numérique, de la scène du crime vers laboratoire, jusqu'à la présentation du cas.

V.2 Environnement de développement

Pour réaliser notre système, nous avons un PC I7 doté de Windows 7 (64bits) comme environnement de test avec les détails suivants :

Fabricant	Acer
Modèle	Aspire E1-571G
Processeur	Intel(R) Core (TM) i7-3632QM CPU @ 2.20 GHz.2.20 GHz
Mémoire physique	9082 Mo
Système d'exploitation	Windows 7 Professional
Type de système	Système d'exploitation 64 bits

Tableau 5. Description technique de l'environnement.

V.3 Les outils de développement du système

Dans cette section nous présenterons les outils de développement dans lesquels nous avons développé notre application, à travers la présentation des différents outils et langages utilisés en plus de la justification des choix de ces outils. Nous les décrivons brièvement ci-dessous.

V.3.1 Langages de programmation

Le langage JAVA : est un langage de programmation orienté objet développé par Sun Microsystems. Les premières versions datent de 1995, il a réussi à intéresser et intriguer beaucoup de développeurs à travers le monde. [Oracle, 2019]

La syntaxe générale du langage java est très proche de celle du langage C, et parmi ces avantages :

- Java assure une totale indépendance des applications vis-à-vis de l'environnement d'exécution : c'est à dire que toute machine supportant Java est en mesure d'exécuter un programme sans aucune adaptation (ni recompilation, ni paramétrage de variables d'environnement).
- Java est un langage orienté objet, c'est à dire que nous n'allons pas manipuler des fonctions et des procédures mais des objets qui vont s'échanger des messages. Le principal avantage est que l'on peut réaliser une programmation modulaire : tous les objets peuvent être mis au point séparément.
- Java nous permet un accès simplifié aux bases de données, soit à travers la passerelle JDBC-ODBC ou à travers un pilote JDBC spécifique au SGBD.
- Il est particulièrement adapté au développement d'application communiquant à l'intermédiaire d'un réseau. L'API java est très riche : différents packages permettant d'accéder aux réseaux, aux entrée/sortie et aux différents composants graphiques.

Le langage PowerShell : est une interface en ligne de commande basé sur les tâches, un langage de script s'appuie sur le Framework Microsoft .NET. [Microsoft,2019]

PowerShell permet aux administrateurs système et aux utilisateurs chevronnés d'automatiser rapidement les tâches qui gèrent les systèmes d'exploitation (Linux, MacOS et Windows) et les processus.

Les avantages de langage PowerShell :

- Les fournisseurs PowerShell nous permettent d'accéder aux mémoires de données, telles que le registre et le référentiel de certificats, aussi facilement que nous pouvons accéder au système de fichiers.

- ❖ Les commandes PowerShell nous permettent de gérer les ordinateurs de la ligne de commande.

PowerShell inclut un analyseur d'expressions riche et un langage de script développé complétement.

Le langage Python: est le langage de programmation le plus utilisé dans le domaine de la Machine Learning, Big Data et Data Science. [Lebigdata,2019]

Créé en 1991, le langage de programmation Python apparu à l'époque comme une façon d'automatiser les éléments les plus ennuyeux de l'écriture de scripts ou de réaliser rapidement des prototypes d'applications. Créé par le programmeur Guido van Rossum en 1991. Il tire son nom de l'émission Monty Python's Flying Circus.

Depuis quelques années, toutefois, ce langage de programmation s'est hissé parmi les plus utilisés dans le domaine du développement de logiciels, de gestion d'infrastructure et d'analyse de données. Il s'agit d'un élément moteur de l'explosion du **Big Data**.

V.3.2 Les outils de technologies

Eclipse est un environnement de développement intégré libre extensible, universel et polyvalent qui va nous permettre de développer nos applications, donc notre logiciel va permettre de traduire nos programmes Java en langage compilé. Mais celui-ci ne peut pas être compris par l'ordinateur, ce code compilé s'appelle du byte-code, il n'est compréhensible que par un environnement Java, vulgairement appelé JRE (Java Runtime Environment) disponible sur le site de Sun Microsystems.

Eclipse IDE 2019-03 est un environnement de développement libre permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation (C++, PHP...). Eclipse IDE est principalement écrit en Java.

La spécificité d'Eclipse IDE vient du fait que son architecture est totalement développée autour de la notion de **plug-in**. Cela signifie que toutes les fonctionnalités de celui-ci sont développées en tant que **plug-in**.

Pour faire court, si vous voulez ajouter des fonctionnalités à Éclipse, vous devez :

- Télécharger le plug-in correspondant.
- Copier les fichiers spécifiés dans les répertoires spécifiés.
- Démarrer Eclipse.

C'est pour toutes ces raisons que j'ai choisies Eclipse comme outil de développement ; de plus, il est relativement simple d'utilisation.

Figure 30. Logo de l'Eclipse.

JDK « Java Développement Kit »de Sun Microsystems, qui est une mise en œuvre de référence puisqu'elle est fournie par le créateur du Java. Le JDK est utilisable sur les plates-formes Linux, Solaris, Mac OS (Macintosh operating system), et Windows (Vista, NT et 95/98/2000).

La version qu'on a utilisée dans notre projet c'est **jdk-6u20,** qui est disponible sur le site suivant :[http://java.sun.com/javase/downloads/widget/jdk6.jsp].

SQLite est une bibliothèque écrite en langage C qui propose un moteur de base de données relationnelle accessible par le langage SQL.

SQLite implémente en grande partie le standard SQL-92 et des propriétés ACID. Contrairement aux serveurs de bases de données traditionnels, comme MySQL ou PostgreSQL, sa particularité est de ne pas reproduire le schéma habituel client-serveur mais d'être directement intégrée aux programmes [SQLite, 2019]. L'intégralité de la base de données (déclarations, tables, index et données) est stockée dans un fichier indépendant de la plateforme.

On peut trouver diverses fonctionnalités dans SQLite **[** https://www.sqlite.org/features.html] :

1. Implémentation SQL multifonctionnelle avec des fonctionnalités avancées comme les index partiels, les index sur expressions, JSON, les expressions de tables communes et les fonctions de fenêtres.
2. Zéro-configuration - aucune installation ou administration nécessaire.
3. Supporte les bases de données de taille téraoctet
4. API simple et facile à utiliser.
5. Rapide : Dans certains cas, SQLite est plus rapide que les E/S directes du système de fichiers.

MySQL & phpMyAdmin :

MySQL : est un système de gestion de bases de données relationnelles (SGBDR). Il fait partie des logiciels de gestion de base de données les plus utilisés au monde. MySQL fait référence au Structured Query Language, le langage de requête utilisé.

PhpMyAdmin: est une interface d'administration pour le SGBD MySQL. Il est écrit en langage PHP et s'appuie sur le serveur HTTP Apache.

V.4 Diagramme de classe de l'application

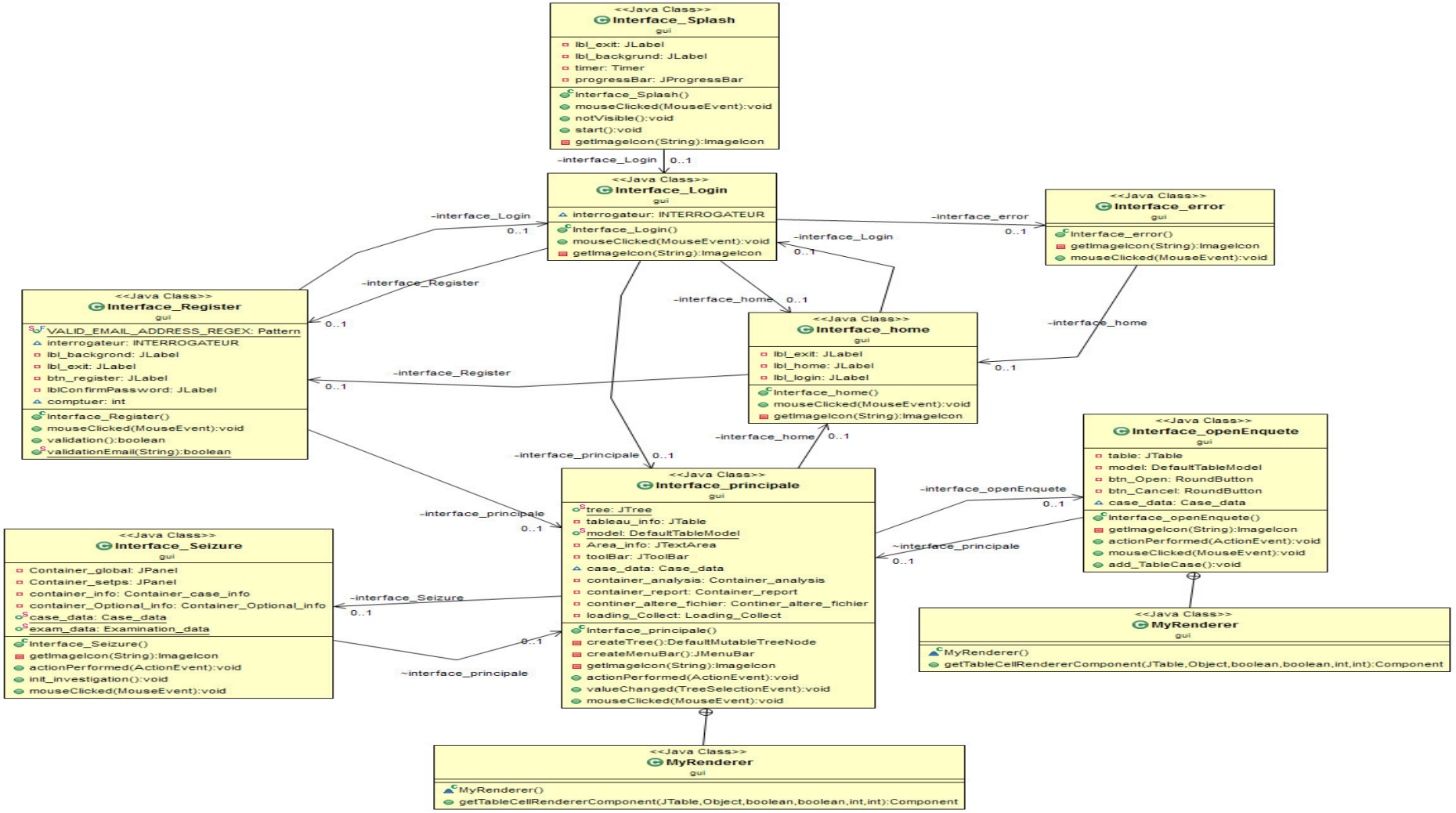


Figure 31. Génération d'un diagramme de classes à l'aide d'eclipse pour le plugin objectAid.

V.5 Base de données du système proposé

V.5.1 Schéma générale de la base de données

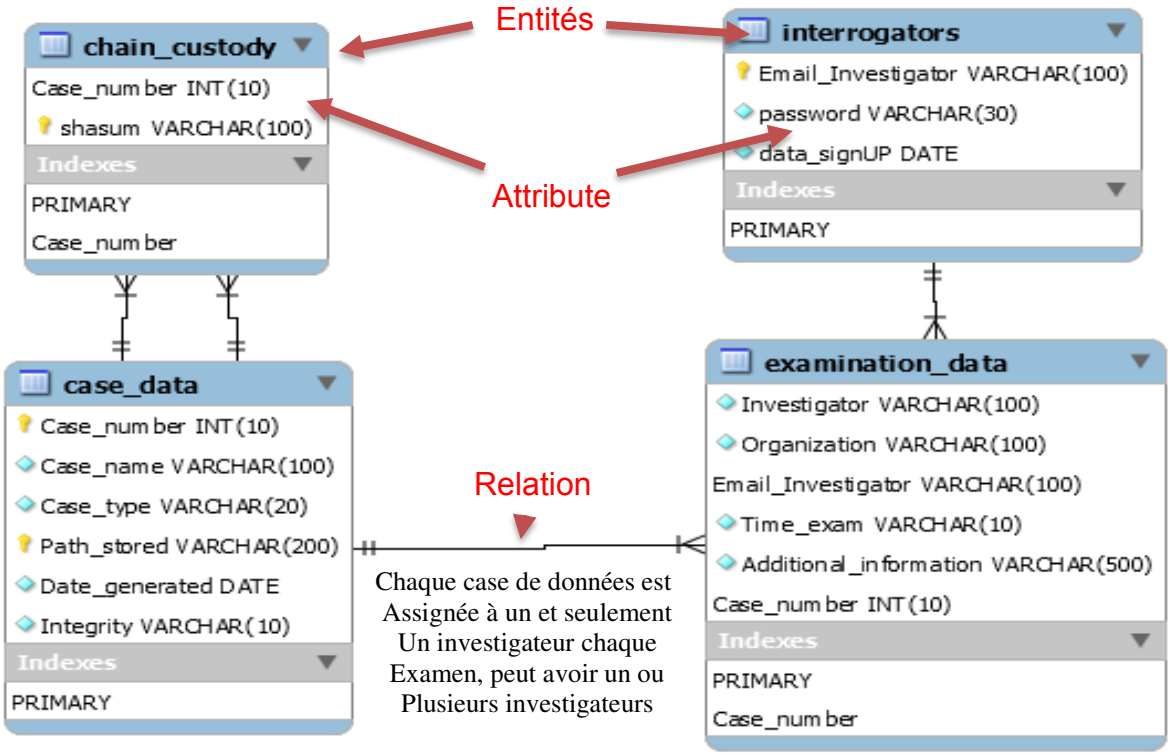


Figure 32. Génération un schéma de base de données avec MySQL Workbench.

V.5.2 Principaux table de la base de données

V.5.2.1 Traduction vers le modèle relationnel :

Case_Data (**Case_number**, Case_name, Case_type, Path_stored, Date_generated, Integrity)

Examination_Data (**#Email_Investigator, #Case_number**, Investigator, Organization, Time_exam, Additional_information)**.**

Chain_Custody (**Shasum**, **#Case_number**)**.**

Interrogateurs (**Email_Investigator**, Password, Data_signUP)

Dans notre système, les quatre tableaux de base sont les suivants :

Le Tableau 1, présente les données de cas est une description simple de l'enquête à laquelle l'examen.

MySQL:

```
CREATE TABLE IF NOT EXISTS Case_Data ( Case_number INT AUTO_INCREMENT UNSIGNED, Case_name VARCHAR(100) NOT NULL, Case_type VARCHAR(20) NOT NULL, Path_stored VARCHAR(500) , Date_generated DATE, Integrity BOOLEAN, PRIMARY KEY (Case_number) ) ENGINE=INNODB;
```

Le Tableau 2, présente les données de l'examen comprendrait le nom de la personne qui a ordonné l'examen, un identificateur pour l'enquête et des renseignements permettant d'identifier la preuve examinée.

MySQL:

```
CREATE TABLE IF NOT EXISTS Examination_Data ( Case_number INT AUTO_INCREMENT UNSIGNED, Investigator_Name VARCHAR(100) NOT NULL, Organization VARCHAR(100) NOT NULL, Email_Investigator VARCHAR(100) NOT NULL , Time_exam DATETIME, Additional_information VARCHAR(500) , PRIMARY KEY (Case_number, Email_Investigator), FOREIGN KEY (Case_number) REFERENCES Case_Data (Case_number)) , FOREIGN KEY (Email_Investigator) REFERENCES interrogators (Email_Investigator)) ) ENGINE=INNODB;
```

Le Tableau 3, présente la chaîne de contrôle pour calculer une empreinte digitale qui identifiera le fichier à l'aide d'une fonction de hachage. Cette empreinte digitale doit être unique parce qu'elle confirme que le fichier n'a pas été altéré au cours de l'enregistrement.

MySQL:

```
CREATE TABLE IF NOT EXISTS Chain_Custody (Case_number INT AUTO_INCREMENT UNSIGNED, Shasum VARCHAR (100) NOT NULL, PRIMARY KEY (Case_number, Shasum), FOREIGN KEY (Case_number) REFERENCES Case_Data (Case_number))) ENGINE=INNODB;
```

Le Tableau 4, présente des renseignements sur l'inscription des enquêteurs.

MySQL:

```
CREATE TABLE IF NOT EXISTS interrogators (Email_Investigator VARCHAR (100) NOT NULL, Password VARCHAR (100) NOT NULL, Data_signUP DATE, PRIMARY KEY (Email_Investigator)) ENGINE=INNODB;
```

V.6 Présentation de l'étude de cas

Lors d'une analyse forensic et particulièrement dans des environnements en cours de production, il est nécessaire de disposer des outils adéquats qui permettront de mener à bien l'investigation.

Afin d'expliquer notre objectif, on procédera à une étude de cas, où on appliquera notre approche à un exemple sur un système en exécution. En effet, des éléments tels que la RAM ou les processus actifs ne peuvent être analysés que lorsque le système est en cours d'exécution.

Ce travail s'effectue avec l'organisme d'application de la loi en tant qu'équipe de volontaires, car les autres n'ont pas l'autorisation légale de travailler avec l'analyse judiciaire numérique contre la cybercriminalité d'une personne soupçonnée.

On suppose le scénario suivant :

Étape 1. Au début, les victimes de fraude numérique fournissent à la Digital Investigation Agency suffisamment de preuves d'un crime commis.

Figure 33. Les victimes (entreprises ou particuliers) déclarent un crime numérique.

Étape 2. Après évaluation et identification des enquêtes numériques et acceptation du cas par l'Agence d'investigation numérique, ce dernier délègue l'enquêteur judiciaire et lui fournit les informations nécessaires sur l'affaire.

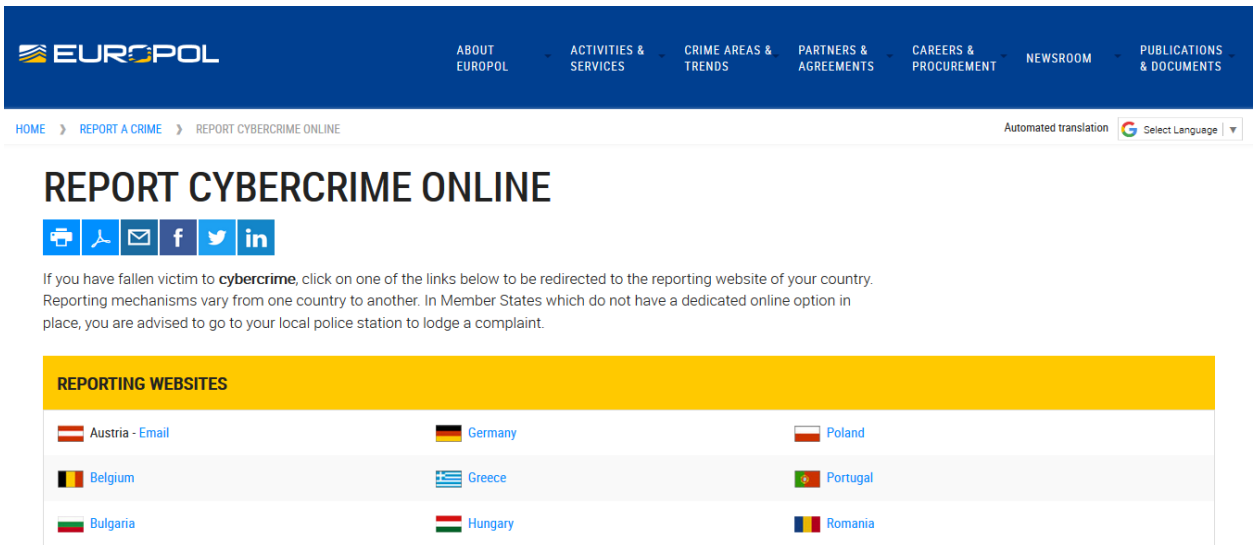


Figure 1. Interface principale pour signaler la cybercriminalité en ligne.

Étape 3. Dans le cas de notre échantillon, on a demandé à l'expert judiciaire d'effectuer un examen judiciaire d'un ordinateur soupçonné d'avoir participé à une affaire criminelle en ligne.

Le crime est le suivant :

- Le suspect, Alan, était soupçonné d'être agressif en politique, il a toujours navigué sur des sites Web étrangers sur les attaques terroristes via Internet pour essayer d'apprendre comment faire des attaques terroristes.
- Les voisins d'Alan ont rapporté qu'ils ont entendu des bruits qui ressemblent à ceux d'une explosion. Ils ont eu peur qu'Alan fabrique des bombes.

Étape 4. Après avoir reçu cette affaire, la police scientifique se dirige vers Alan et heureusement, ils l'ont trouvé devant son ordinateur dans une situation suspecte.

Étape 5. Si l'enquêteur judiciaire trouve un ordinateur avec un écran de login (ordinateur verrouillé). On peut passer la page de log-in Windows sans redémarrer en utilisant certains outils techniques pour éviter de perdre du contenu RAM :

Nous présentons l'outil Inception comme un exemple qui peut être poursuivi en justice pour accéder à un ordinateur verrouillé, en utilisant une attaque d'accès direct à la mémoire (DMA) pour extraire le mot de passe RAM pour notre système.

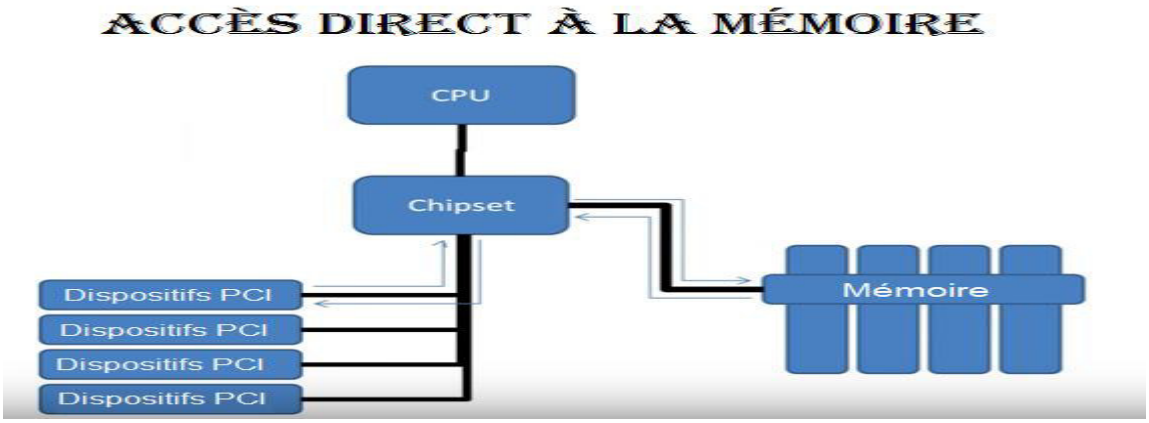


Figure 35. Fonctionnement de la technique DMA.

General usage:

1. Relier la machine attaquante (hôte) et la victime (cible) avec un câble FireWire
2. Run Inception

Pour déverrouiller un ordinateur verrouillé, tapez simplement :

```
incept unlock

 _|  _|      _|    _|_|_|  _|_|_|_|  _|_|_|    _|_|_|  _|    _|_|    _|      _|
 _|  _|_|    _|  _|        _|        _|    _|    _|    _|  _|    _|  _|_|    _|
 _|  _|  _|  _|  _|        _|_|_|    _|_|_|      _|    _|  _|    _|  _|  _|  _|
 _|  _|    _|_|  _|        _|        _|          _|    _|  _|    _|  _|    _|_|
 _|  _|      _|    _|_|_|  _|_|_|_|  _|          _|    _|    _|_|    _|      _|

v.0.4.0 (C) Carsten Maartmann-Moe 2014
Download: http://breaknenter.org/projects/inception | Twitter: @breaknenter

[?] Will potentially write to file. OK? [y/N] y
[*] Available targets (known signatures):

[1] Windows 8 MsvpPasswordValidate unlock/privilege escalation
[2] Windows 7 MsvpPasswordValidate unlock/privilege escalation
[3] Windows Vista MsvpPasswordValidate unlock/privilege escalation
[4] Windows XP MsvpPasswordValidate unlock/privilege escalation
[5] Mac OS X DirectoryService/OpenDirectory unlock/privilege escalation
[6] Ubuntu libpam unlock/privilege escalation
[7] Linux Mint libpam unlock/privilege escalation

[?] Please select target (or enter 'q' to quit): 2
[*] Selected target: Windows 7 MsvpPasswordValidate unlock/privilege escalation
[=============>                                                ]  227 MiB ( 22%)
[*] Signature found at 0xe373312 in page no. 58227
[*] Patch verified; successful
[*] BRRRRRRRAAAAAWWWWRWRRRMRMRMMRMRMMMM!!!
```

Figure 36. Console de terminal pour lancer Inception

Étape 6. Après avoir déverrouillé l'ordinateur, l'enquêteur insère une clé USB légale, il inclut un outil d'investigation numérique sur les systèmes Windows. Ces outils sont quelques fois difficiles à transporter, et souvent payants. C'est pour cela que nous avons créé un outil USB facilement transportable et regroupant les outils gratuits spécialisés.

L'enquêteur lance l'outil de légiste :

Figure 37. Interface de Splash

Étape 7. Afin de bénéficier de l'ensemble des services fournis par notre outil Zodiac Forensics, il est nécessaire de créer un compte qui sera utilisé pour s'authentifier. L'enquêteur doit avoir un compte à L'organisme d'investigation numérique. Toute inscription incomplète ne sera pas validée.

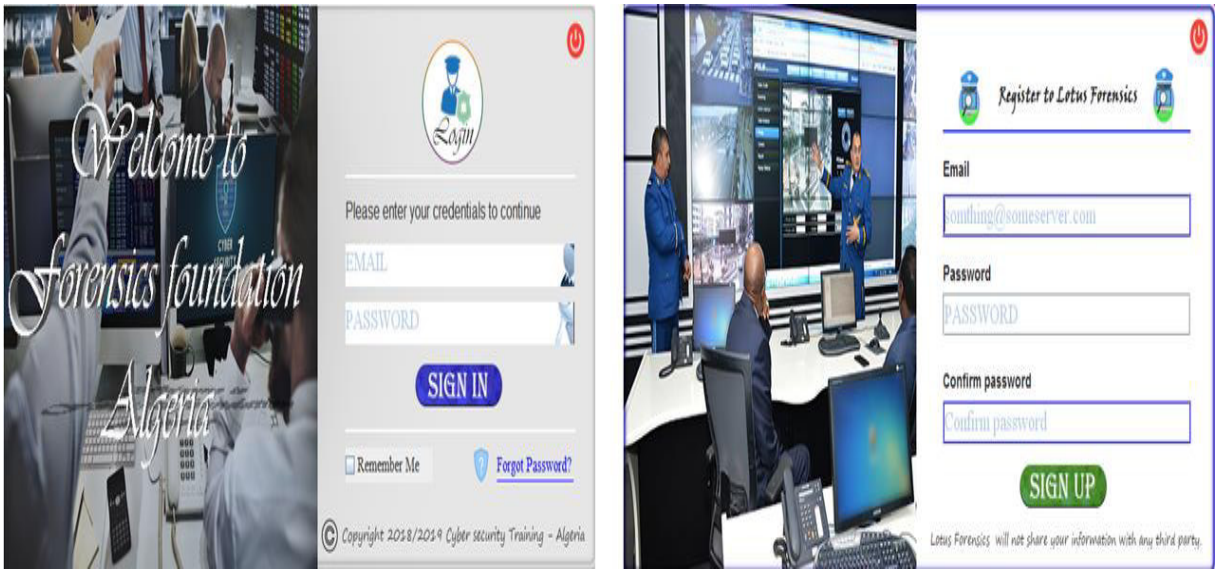


Figure 38. Interface de connexion et inscription.

Étape 8. L'enquêteur connecte dans son propre compte s'il est admissible, l'outil lui dirigera vers son compte dans l'organisation et vers l'interface principale de l'outil.

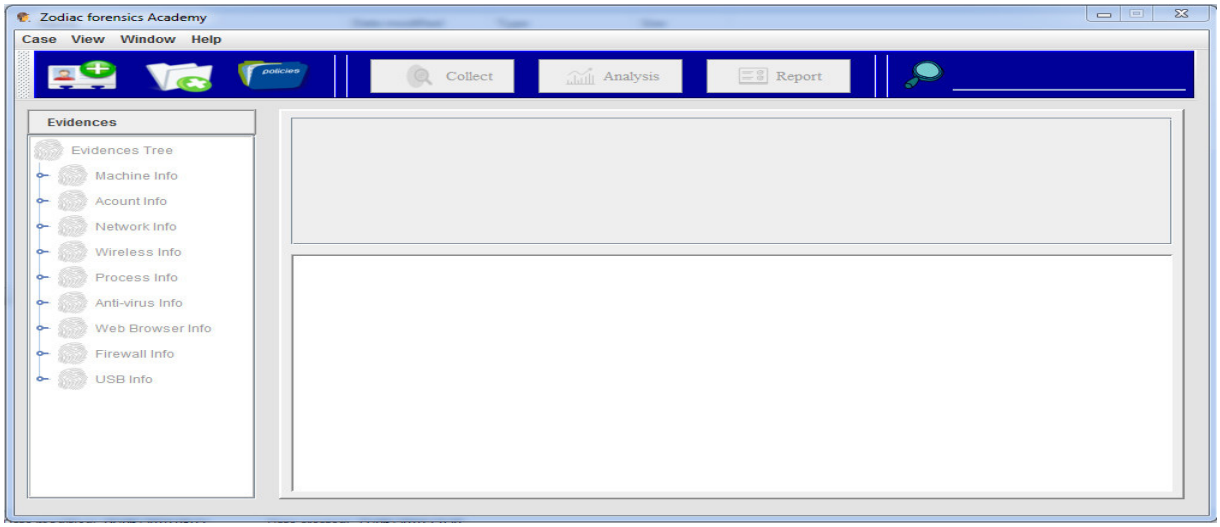


Figure 39. Interface principale de l'outil légiste

Étape 9. L'enquêteur doit créer un cas avant de pouvoir analyser les données, en utilisant l'option "Créer un nouveau cas" sur l'écran d'accueil ou dans le menu "Fichier". (Voir Figure)

Figure 40. Clique sur l'option "Créer un nouveau cas"

Étape 10. On ouvrira la fenêtre de dialogue de l'assistant du cas et on devra entrer tous les renseignements requis avant le début de l'investigation. Cela inclut le nom et type de l'affaire, et le répertoire de base dans lequel les résultats de l'enquête peuvent être conservés.

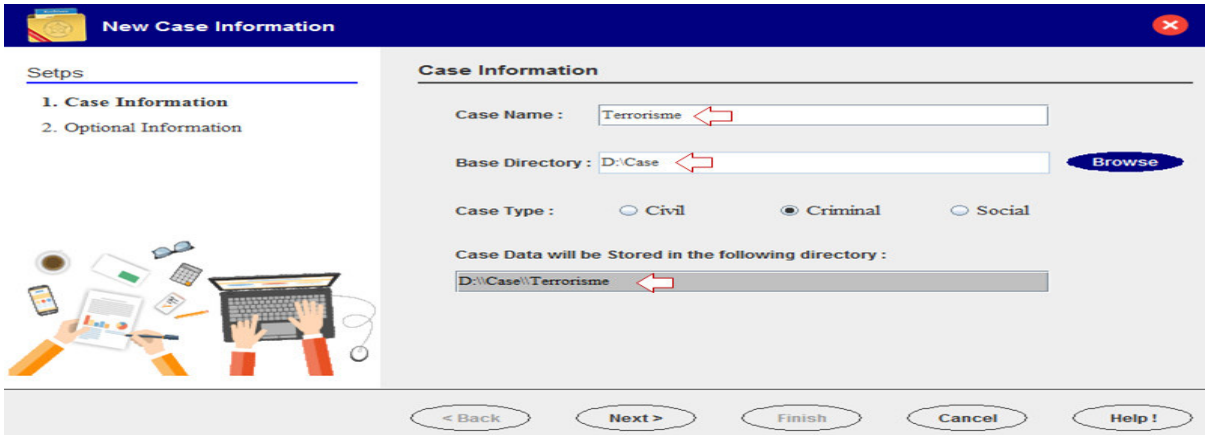


Figure 41. Interface pour créer un nouveau cas

Remarque

- ***Chaque cas a son propre répertoire qui est nommé en fonction du nom du cas.***
- ***Si le répertoire existe déjà, on doit soit supprimer le répertoire existant, soit choisir une autre combinaison de noms.***

Étape 11. On demandera également de donner des renseignements supplémentaires, comme le nom de l'enquêteur et le numéro de dossier et des renseignements qui identifient les éléments de preuve qui font l'objet de l'examen.

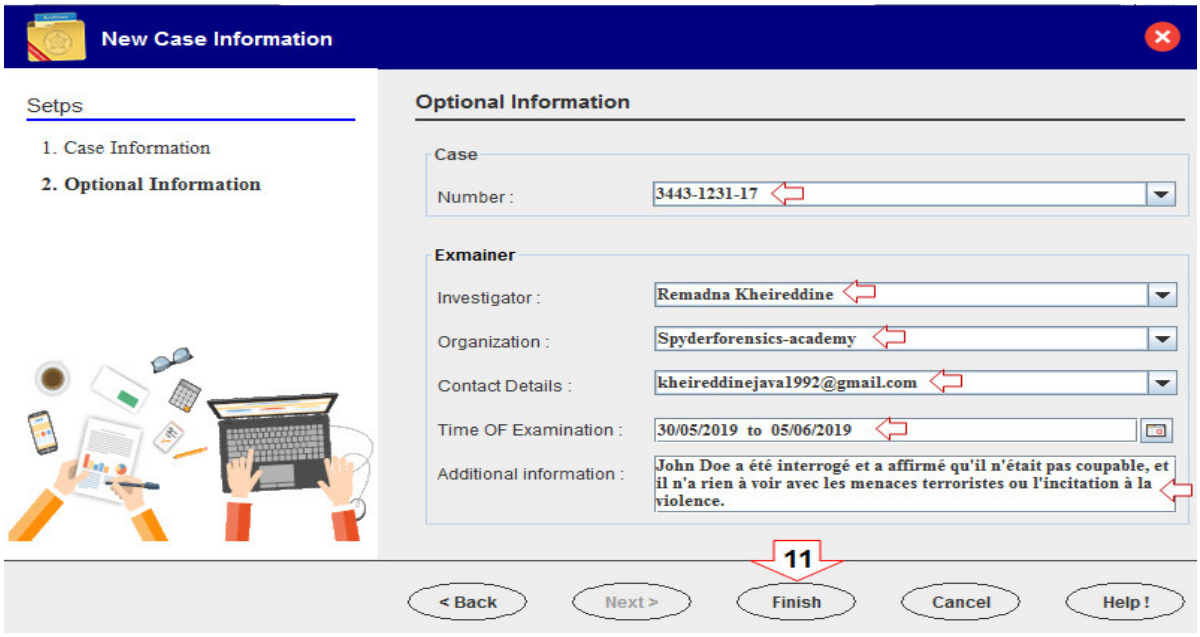


Figure 42. Interface pour les renseignements supplémentaires.

Étape 12. Après avoir terminé le processus de saisie des informations, on clique sur le bouton Terminer. L'outil va générer automatiquement un répertoire pour le cas dans le "répertoire de base de la clé USB légale". Le répertoire contiendra des fichiers de configuration, une base de données, des rapports et d'autres fichiers générés par les modules.

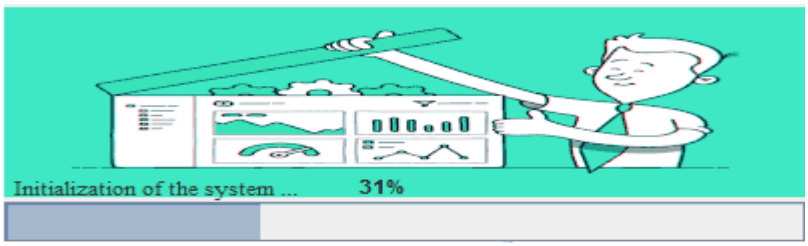


Figure 43. Interface pour charger la configuration du répertoire.

Les étapes suivantes sont des étapes d'investigation numérique sur la mémoire physique de l'ordinateur:

Étape 13. Les outils regroupés dans cette catégorie permettent à l'analyste de faire une copie fidèle d'un support numérique sur la mémoire RAM. Ils permettent également de copie des fichiers protégés par le système, on clique sur le bouton Collect (Voir Figure).

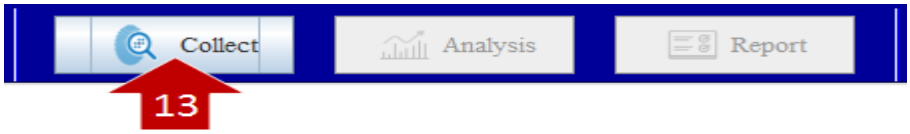


Figure 44. Clique sur le bouton "Collect "

Code source PowerShell :

```
# adapt these two lines to your loacal system
[System.Reflection.Assembly]::LoadFrom("C:\Users\khairy\eclipse-workspace\MiniProjet\PSSQLite-
master\PSSQLite\x64\System.Data.SQLite.dll")
$ConnectionString = "Data
Source=C:\Users\khairy\AppData\Roaming\Mozilla\Firefox\Profiles\jwabt4tn.default\places.sqlite"

$conn=new-object System.Data.SQLite.SQLiteConnection
$conn.ConnectionString=$ConnectionString
$conn.Open()
$sql = "SELECT * from moz_origins"
$cmd=new-object System.Data.SQLite.SQLiteCommand($sql,$conn)
$ds=New-Object system.Data.DataSet
$da=New-Object System.Data.SQLite.SQLiteDataAdapter($cmd)
$da.fill($ds)
$conn.close()
$ds.tables[0] | out-file -encoding ASCII D:\Forensic_Report_31-05-2019\Browser_Forensic_Info.txt -Append -Width 100
```

Figure 45. Récupération de l'historique de navigation avec script du PowerShell.

Remarque
Le script Get-MemoryDump complet pour l'acquisition de preuves volatiles est disponible en annexe.

Note Sur phase préservation

Après avoir terminé le processus d'acquisition des preuves, l'outil calcule automatiquement son empreinte numérique (MD5 et/ou SHA1) pour assurer l'intégrité des fichiers lors de l'acquisition.

Code source Java :

Algorithm 1 Algorithme de calcul hash d'un fichier

```
1:  procedure FILECHEKSUM(MessageDigestdigest, Filefile)                    ▷
2:      // Get file input stream for reading the file content
3:
4:      File[] myMusicFiles = file.listFiles();
5:      FileInputStream fis = new FileInputStream(myMusicFiles[0]);
6:
7:      // Create byte array to read data in chunks
8:      byte[] byteArray = new byte[1024];
9:      int bytesCount = 0;
10:
11:     // Read file data and update in message digest
12:     while ( (bytesCount = fis.read(byteArray)) ≠ −1 ) do                ▷
13:         digest.update(byteArray, 0, bytesCount);
14:
15:         // close the stream; We don't need it now.
16:         fis.close();
17:
18:         // Get the hash's bytes
19:         byte[] bytes = digest.digest();
20:
21:         // This bytes[] has bytes in decimal format;
22:         // Convert it to hexadecimal format
23:         StringBuilder sb = new StringBuilder();
24:
25:         for(int i = 0; i ¡ bytes.length; i++)
26:         sb.append(Integer.toString((bytes[i]&0xff)+0x100,16).substring(1));
27:
28:     return sb.toString()                          ▷ // return complete hash
```

Figure 46. Algorithme de hachage pour un fichier.

Algorithm 2 Algorithme de calcul hash MD5 d'un fichier

```
1:  procedure ENCRYPTPASSWORD(StringPath)                                   ▷
2:      String checksum = "";
3:      try {
4:
5:      File file = new File(derctory(Path));              ▷ Use MD5 algorithm
6:
7:      MessageDigest md5Digest;
8:      md5Digest = MessageDigest.getInstance("MD5");
9:      // Get the checksum
10:     checksum = getFileChecksum(md5Digest, file);          ▷ see checksum
11:
12:     } catch (NoSuchAlgorithmException — IOException e) {
13:     e.printStackTrace();
14:     }
15:
16:     return checksum                               ▷ // return complete hash
```

Figure 47. Procédure de calcul hach MD5 d'un fichier.

Note sur la phase d'examen

Avant de commencer le processus d'analyse criminalistique, l'outil confirme automatiquement que les données copiées ne sont pas modifiées pendant l'investigation.

Code source Java :

Algorithm 3 algorithm of forensic exman

```
1:  procedure EXMAN EMPREINTE(FileOrigine, FileCopie)                ▷
2:      Origine = Get_Filecheksum.encryptPassword(verification.derctory());
3:      Copie = EmpreinteSQL.Get_value_hash(verification.derctory()).trim();
4:      if FILE_compare(Origine, Copie) ≠ 0 then
5:          //le fichier n'a pas été altéré durant l'investigation
6:          Validateur= False;
7:      else
8:          //le fichier a été modifié au cours de l'enquête
9:          Validateur= True;
10:
11:     return Validateur                          ▷ Empreinte File est Vrai ou faux
```

Figure 48. Algorithme pour assurer l'intégrité de la preuve

Étape 14. Les outils regroupés dans cette catégorie permettent à l'enquêteur d'analyser le contenu d'un navigateurs internet, le système de fichier, les processus actifs, les logs et évènements du système. On clique sur le bouton Analysis. (Voir Figure).

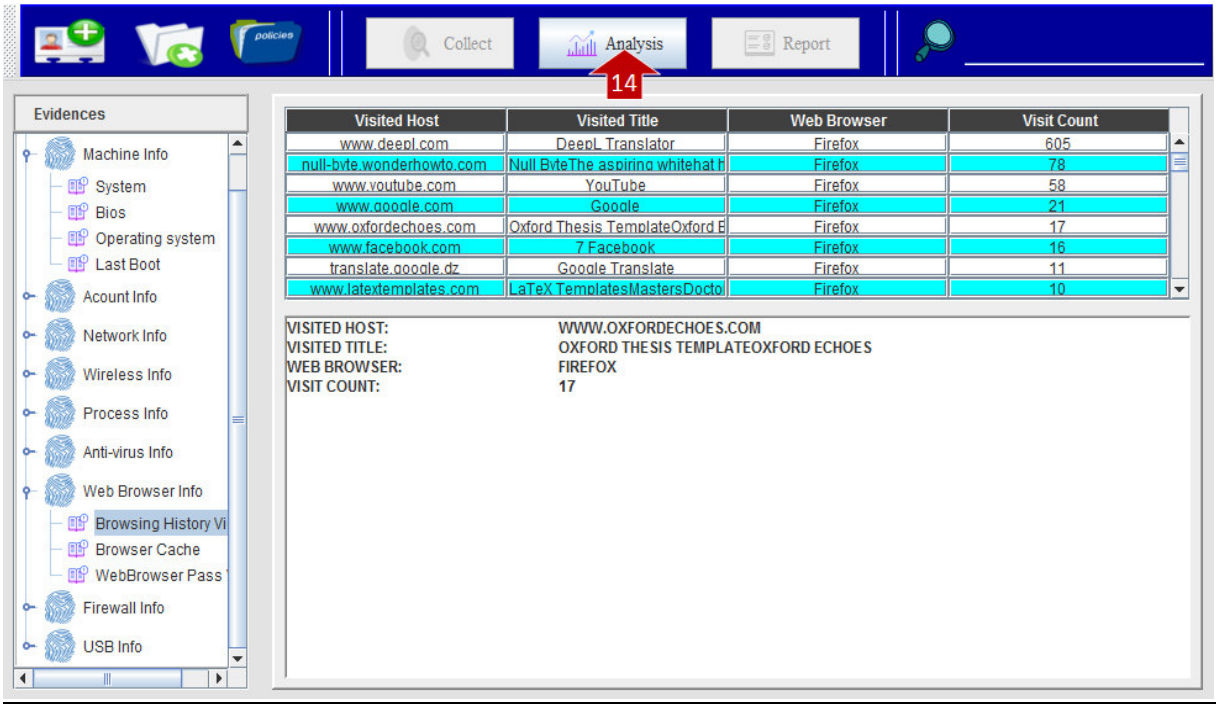


Figure 49. Interface pour l'analyse d'artefacts de navigateur web.

Code source Java :

Algorithm 4 Algorithm of forensic analysis

```
1:  procedure ANLAYSE HOST(FileBlack_listed)        ▷ liste noire de site web
2:      try {
3:      Class.forName("org.sqlite.JDBC");
4:      Connection conn = DriverManager.getConnection("jdbc sqlite +
5:      verification.derctory() + " // db // Web_host_info.db   ")
6:      Statement statement = conn.createStatement();
7:      ResultSet resultSet = null;
8:
9:      String website_balck = ReadFile.readfile(newFile(Black_listed));
10:     String[]List_balck =  website_balck.split(System.lineSeparator());
11:     for (inti = 1;i < List_balck.length;i + +){
12:     List_balck[i] = List_balck[i].substring(List_balck[i].indexOf(" : ")+1);
13:     Stringsql = "SELECT distinct Host, Frecency FROM WebFrecency
14:      WHERE Host =' " + List_balck[i].trim() + "'";
15:     res = statement.executeQuery(sql);
16:     while (res.getString("Host").equals(host) ≠ 0) do
17:         {
18:         if (Integer.parseInt(res.getString("Frecency")) ≥ max) then
19:             StringHost = res.getString("Host").trim().replaceAll("www.","");
20:             Host = Host.substring(0, Host.indexOf("."));
21:             String[]list1 = Host, resultSet.getString("Frecency").trim();
22:             arry_web_web.add(list1);
23:             max = Integer.parseInt(res.getString("Frecency"));
24:             host = res.getString("Host");
25:             Host = "";
26:         }                                          ▷ End while
27:     max = 0;
28:     }                                              ▷ End for
29:     conn.close();
30:     }catch(Exceptionexception){
31:     JOptionPane.showMessageDialog(null,"Erreur");
32:     }
33:     return arry_web_web                            ▷ Struct Website
```

Figure 50. Algorithme pour de navigateur web

Résultat de l'analyse du navigateur :

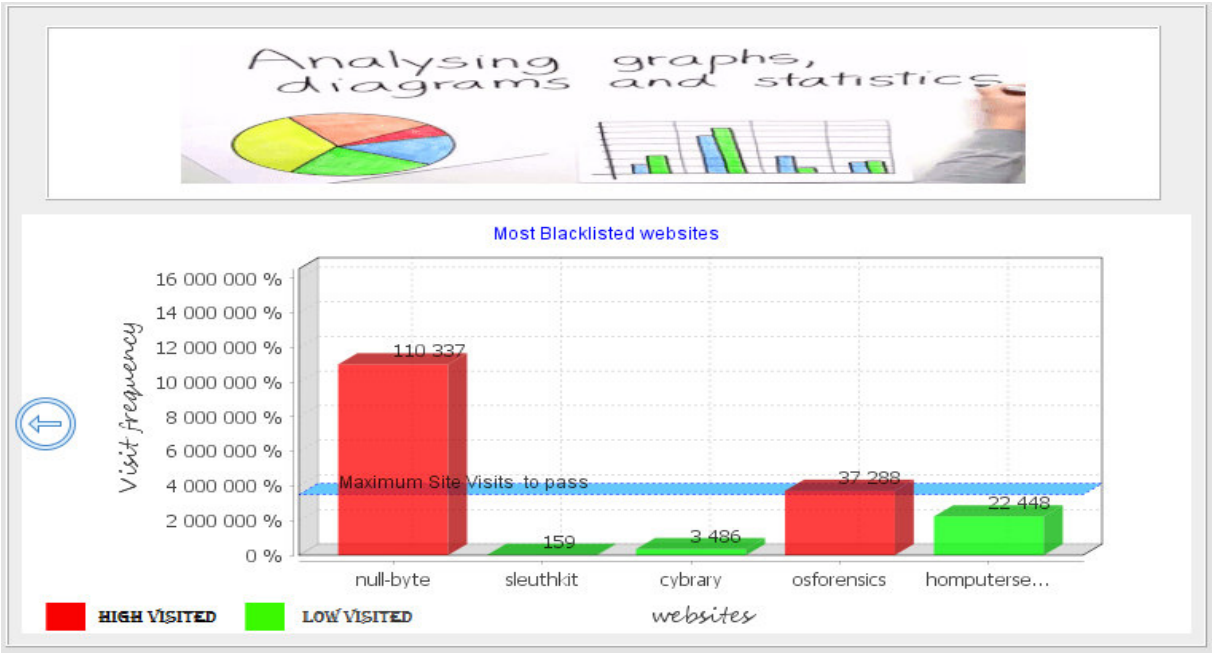


Figure 51. Suivre les activités d'un utilisateur via une liste noire des sites terroristes.

Étape 15. Analyse les e-mails suspects pour Alan

On utilise l'outil de la ligne de commande gratuit pour récupérer instantanément notre mot de passe perdu de tous les navigateurs Web populaires de la machine suspecte, comme l'utilitaire **BrowserPasswordDump**.

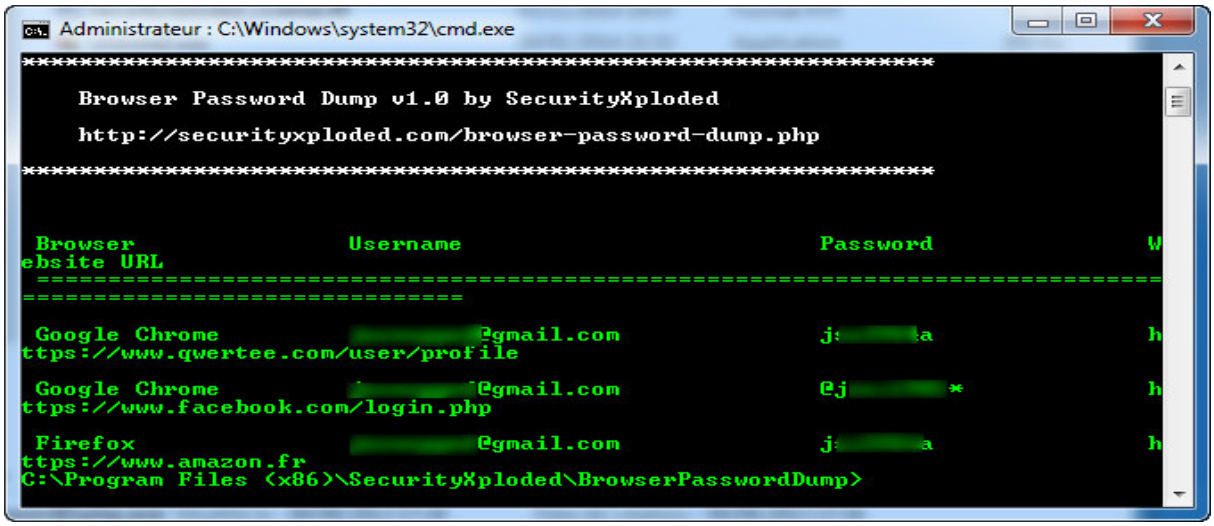


Figure 52. Extraction de mot de passe avec BrowserPasswordDump [Bpd19].

1. On essaye de trouver un mot de passe pour accéder aux comptes d'email suspects d'Alan (tels que **Gmail, Yahoo et Outlook**).

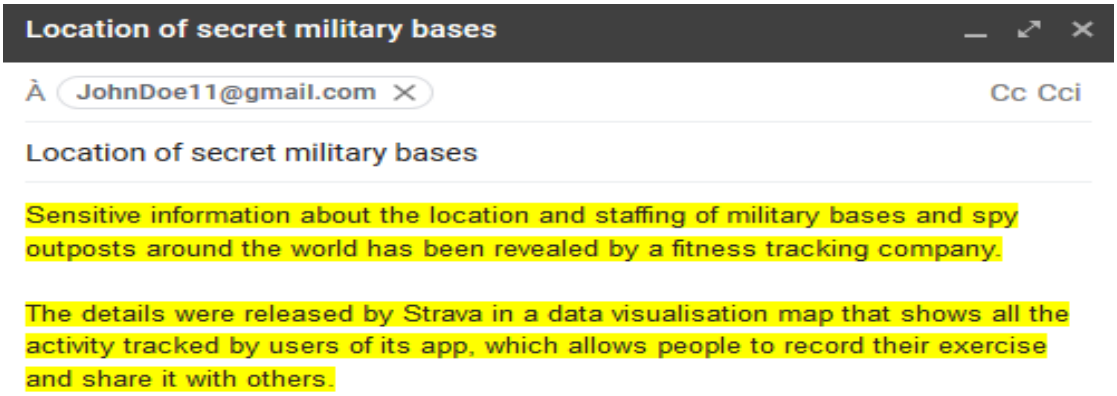


Figure 2. Étude sur les communication suspecte par e-mails entre Alan et John

2. On s'intéresse à la recherche et à la récupération des e-mails de la machine suspecte, et à l'analyse du message, puis à l'extraction des informations utiles telles que son adresse IP et la date/heure à laquelle un message spécifique lui a été envoyé, et enfin la traçabilité de son origine (l'émetteur).

3. Révéler toutes les informations de l'en-tête de l'e-mail du suspect

1. Accès au compte Gmail cible à l'aide de son navigateur préféré
2. Ouvrir la boîte de réception de l'e-mail dont on a besoin d'afficher l'en-tête
3. A côté de Reply, on clique sur la flèche vers le bas (voir Figure 53.1)
4. On clique sur “Show original”

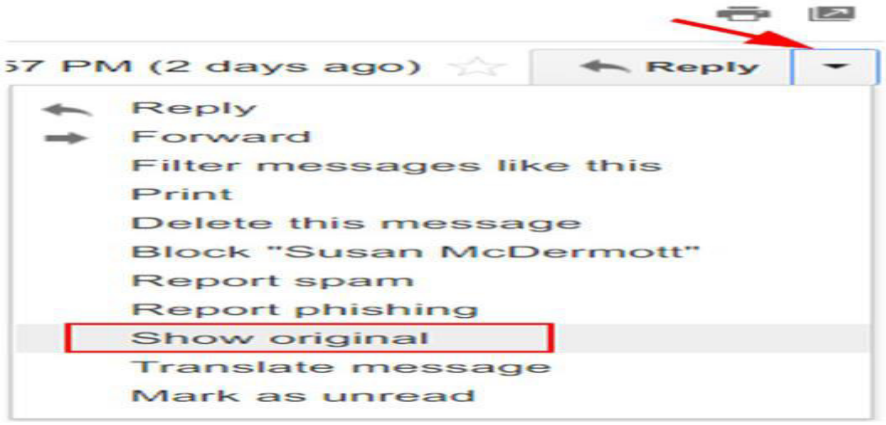


Figure 53.1. Voir l'en-tête de l'e-mail sur Gmail

4.1. Analyse de en-tête de message électronique

La Figure 53.2 est un exemple l'en-tête d'e-mail provenant d'un message reçu via le service Gmail entre Alan et John.

Figure 53.2. Exemple d'en-tête de message électronique.

Description l'en-tête d’e-mail :

- Le numéro [1] : L'adresse e-mail du récepteur.
- Le numéro [2] : L'adresse IP du destinataire.

- Le numéro [3] : Indique l'adresse IP d'origine (adresse IP de l'expéditeur), Ce champ est facile à falsifier.
- Le numéro [4] : Identifie l'adresse email de l'expéditeur. Malheureusement, le protocole SMTP ne vérifie pas l'identité de l'expéditeur, ce qui rend très facile la falsification de ce champ par manipulation des commandes SMTP via Telnet par exemple.
- Le numéro [5] : Il s'agit d'une chaine unique assignée au mail par le serveur au moment où il est créé.

Remarque

Les lignes commençant par "X" dans l'en-tête de l'e-mail sont des commentaires écrits par le logiciel expéditeur (par exemple, par les logiciels-client d'e-mail), par les serveurs SMTP, et même par les serveurs antivirus/spam présents le long du chemin que le message a emprunté.

4.2. Trouver des preuves légistes en entête de message :

Il existe différents outils pour analyser les en-têtes d'email. On utilise l'outil eMailTrackerPro (www.emailtrackerpro.com) commercial pour le suivi des e-mails en utilisant les en-têtes d'e-mail. Pour suivre un en-tête d'e-mail donné, on utilise cet outil comme suit :

Aller sur le site web de l'outil, télécharger et installer l'outil comme nous le faites avec toute application Windows.

1. Lancer le programme et cliquer sur le bouton "Trace Headers" dans la fenêtre principale de l'outil (voir Figure 53.3).

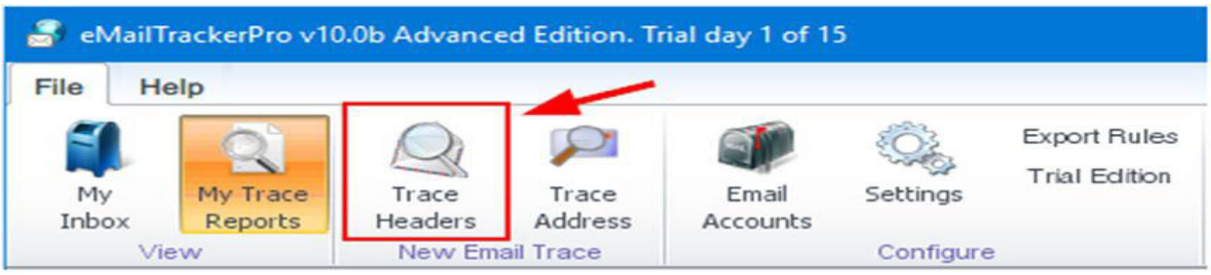


Figure 53.3. Utiliser eMailTrackerPro pour suivre les en-têtes de message.

2. Une nouvelle fenêtre apparait, dans laquelle on peut coller l'en-tête de l'e-mail cible et cliquer sur le bouton "Tracer" pour commencer le traçage.

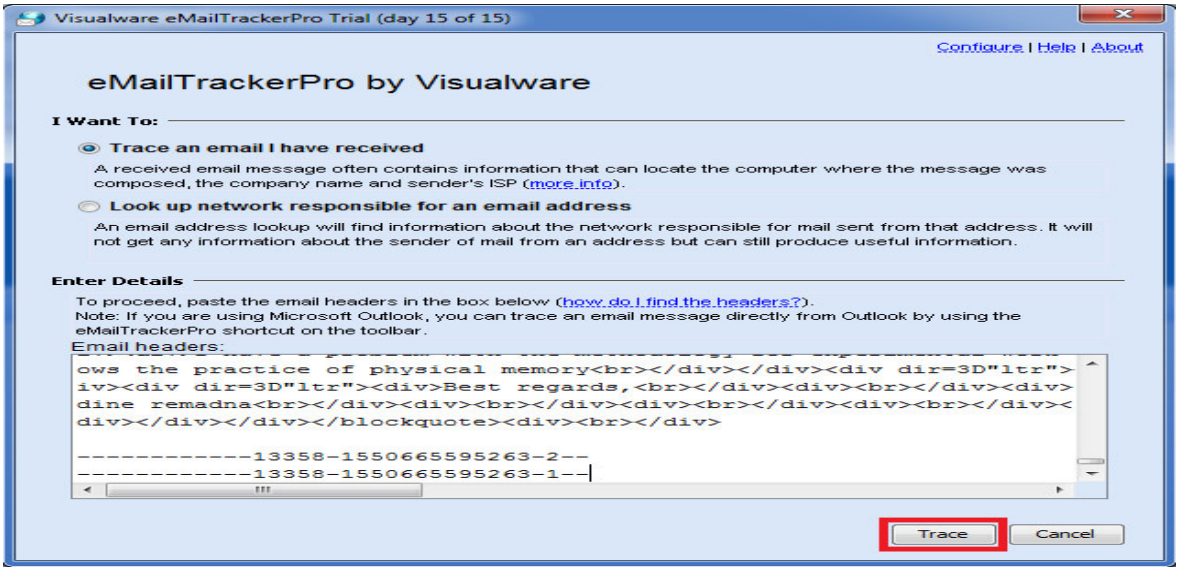


Figure 53.4. Fenêtre de traçabilité

3. Lorsque le traçage est terminé, cliquer sur le bouton "My Trace Reports" dans la fenêtre principale du programme pour afficher un rapport de traçage détaillé (voir Figure 53.5).

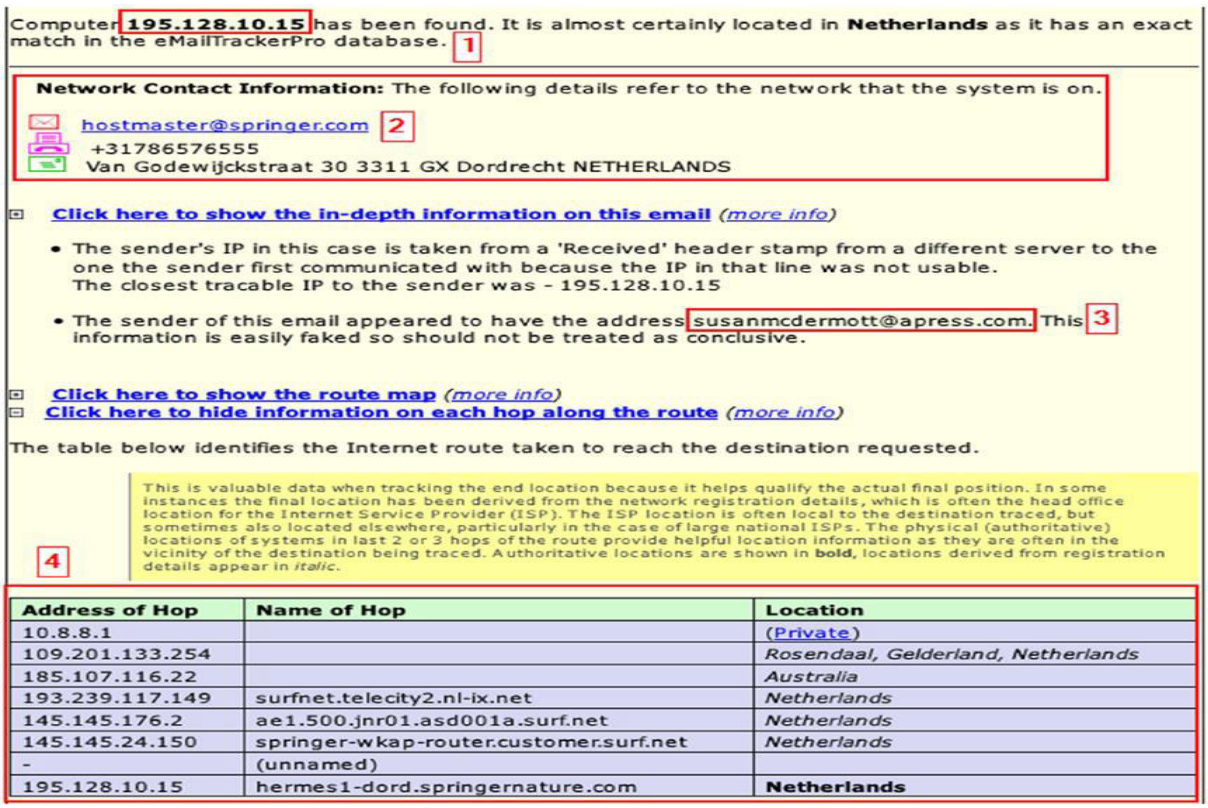


Figure 53.5. Rapport généré par eMailTrackerPro pour tracer une adresse e-mail en utilisant son en-tête.

Description de rapport généré par eMailTrackerPro :

1. Adresse IP d'origine de l'expéditeur, Cette adresse peut être interne (adresse IP privée) ou simplement une adresse IP usurpée ou falsifiée.
2. Informations sur le réseau responsable de l'envoi de cet e-mail.
3. Adresse e-mail de l'expéditeur.
4. Le route Internet que l'e-mail cible suit pour atteindre sa destination finale.

4.3. Détermination de la localisation géographique de l'émetteur suspect

Comme on a vu, l'adresse IP de l'expéditeur peut être extraite de l'en-tête de l'e-mail (aller à la ligne qui commence par " Received: from" à partir de l'en-tête du bas) ; alors on peut utiliser cette adresse IP pour déterminer la position géographique de l'expéditeur. Il existe déjà de nombreux services en ligne permettant de mapper des adresses IP sur des zones géographiques telles que Ipfingerprints (www.ipfingerprints.com) (voir la figure 53.6).

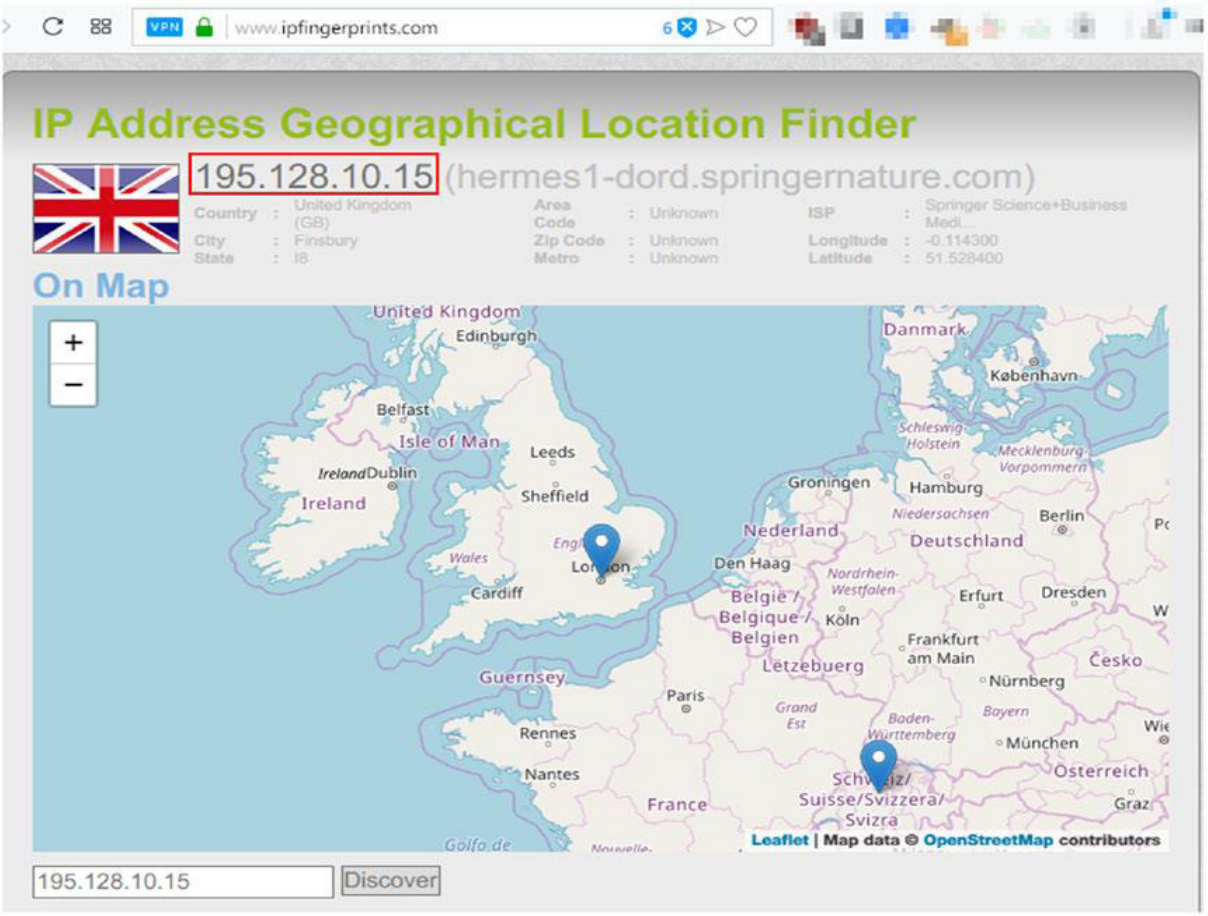


Figure 53.6. Mapping de l'adresse IP à l'emplacement géographique en utilisant www.ipfingerprints.com

Étape 16. Les outils regroupés dans cette catégorie permettent à l'enquêteur de documenter toutes les phases précédentes et de générer des rapports et des conclusions pour présentation au jury ou aux unités de gestion respectives.

On clique sur le bouton Report. (Voir Figure).

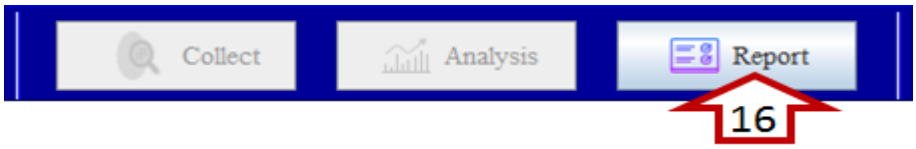


Figure 54. Clique sur le bouton " Report "

Étape 17. L'outil décrit de façon simple les conclusions et les éléments de preuve obtenus durant une investigation criminelle afin de les rendre accessibles à tous. On clique sur le bouton show. (Voir Figure).

Figure 55. Fenêtre pour générer des rapports forensiques

Code source Java :

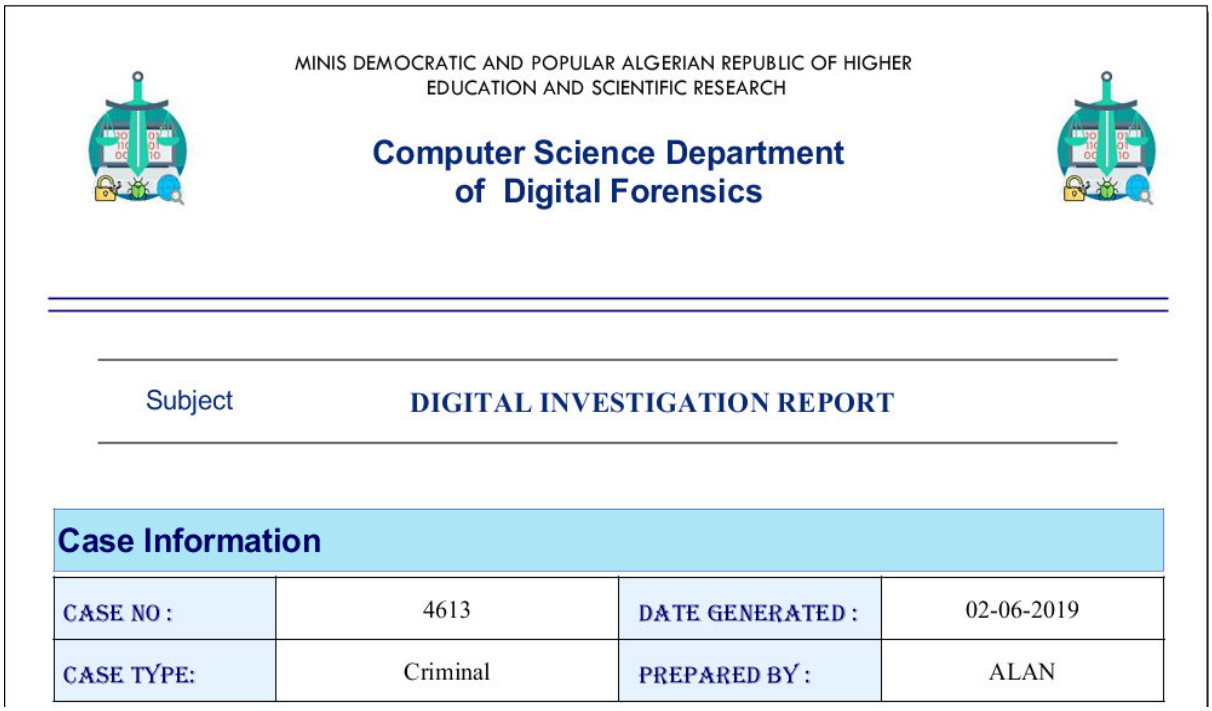
Algorithm 5 algorithm forensic reporting

```
1: procedure GENERATED REPORT(File Black_listed)  ▷ liste noire website
2:     try {
3:
4:     //JasperDesignisusedforin−memoryrepresentationofareportdesign.
5:
6:     JasperDesignjd1 = JRXmlLoader.load(Loader.jfxml_Home);
7:     JasperDesignjd2 = JRXmlLoader.load(Loader.jfxml_Fddinig);
8:     JasperDesignjd3 = JRXmlLoader.load(Loader.jfxml_conclusion);
9:
10:    //JasperReports is an open source java reporting engine
11:
12:    JasperReportreport = JasperCompileManager.compileReport(jd1);
13:    JasperReportreport_2 = JasperCompileManager.compileReport(jd2);
14:    JasperReportreport_3 = JasperCompileManager.compileReport(jd3);
15:    JRDataSourcedata = newJREmptyDataSource();
16:    JRDataSourcedata1 = newJREmptyDataSource();
17:
18:     jr1 = newJRBeanCollectionDataSource(CaseSQL.init_Case(derctory()))
19:     jr2 = newJRBeanCollectionDataSource(ExamSQL.init_Exam(derctory()));
20:
21:    Map <> parm = newHashMap < String, Object > ();
22:    parm.put("Item1", itemsJR1);
23:    parm.put("item2", itemsJR2);
24:    ArrayList < Website > array_web = newArrayList < Website > ();
25:    String[]list_site = newString[2];
26:    Websitewebsite = null;
27:
28:    for (inti = 0; i < Analyse_Browser.anlayse_host().size(); i + +){
29:    list_site = Analyse_Browser.anlayse_host().get(i);
30:    for (intj = 0; j <= list_site.length − 1; j + +){
31:    website = newWebsite();
32:    website.setWebsite(list_website[0].toString());
33:    website.setFrequency(Integer.parseInt(list_site[1].toString()));
34:    }
35:    list_Websites.add(website);
36:    }
37:    items = newJRBeanCollectionDataSource(list_Websites);
38:    res = statement.executeQuery(sql);
39:    parm.put("history", items);
40:    parm.put("image", chart.getUrl_image());
41:
```

```
1:  JasperPrint print = JasperFillManager.fillReport(report, parm, data);
2:  JasperPrint print_2 = JasperFillManager.fillReport(report_2, parm, data1);
3:  JasperPrint print_3 = JasperFillManager.fillReport(report_3, parm, data2);
4:
5:  while (i ≤ Genreted_sys.print_machine().getPages().size())) do
6:      //Add First report
7:      print.addPage((JRPrintPage)Genreted_sys.print_machine().getPages().get(i));
8:      i + +
9:                                                          ▷ End loop 1
10: while (j ≤ print_2.getPages().size()) do
11:     //Add Findings report to second report
12:     print.addPage((JRPrintPage)Genreted_sys.print_machine().getPages().get(j));
13:     j + +
14:                                                         ▷ End loop 2
15: while (k ≤ print_3.getPages().size()) do
16:
17:     //Add conclusion report to End report
18:     print.addPage((JRPrintPage)Genreted_sys.print_machine().getPages().get(k));
19:     k + +
20:                                                         ▷ End loop 2
21:
22: }catch(Exception exception){
23: JOptionPane.showMessageDialog(null, "Erreur");
24: }
```

Figure 55. Algorithme pour générer un rapport légiste

Étape 18. L'expert légiste dépose un rapport à l'agence d'investigation numérique en charge de l'affaire.



MINIS DEMOCRATIC AND POPULAR ALGERIAN REPUBLIC OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Computer Science Department of Digital Forensics

Subject **DIGITAL INVESTIGATION REPORT**

Case Information

CASE NO :	4613	DATE GENERATED :	02-06-2019
CASE TYPE:	Criminal	PREPARED BY :	ALAN

Figure 56. Exemple de rapport forensique

Étape 19. L'agence d'investigation numérique qui prend les décision finale et clôture du l'enquête.

V.7 Conclusion

Dans ce chapitre nous avons décrit les outils de réalisation du système d'investigation numérique de la mémoire par une étude de cas, et aussi par la présentation de l'environnement matériel et logiciel exploite, tout ça pour faciliter le travail de l'enquêteur avec une recherche rapide d'informations sur les systèmes en cours et avoir les résultats assurés.

Dédicace

Je dédie ce modeste travail :

À mon défunt grand-mère ***Remadna Garmia****, que dieu lui accorde sa miséricorde.*

À ceux que j'ai de plus cher au monde, ***mes parents****,*

À mes grands-parents, ma grand-mère, mes frères et sœurs, ainsi que toute ma grande famille.

À tous mes amis.

Kheireddine Remadna

I.1 Contexte du travail

Au fur de la transition du monde vers le numérique, l'utilisation de systèmes informatisés pour fournir des services et stocker des informations devient courante dans les secteurs public et privé. Les individus utilisent aussi beaucoup les ordinateurs dans leur vie quotidienne. Il est rare de voir une personne qui ne dépend pas de son ordinateur pour organiser ses données numériques ou pour communiquer avec d'autres.

Cette évolution a également amené les experts de la cyber-sécurité à se préoccuper de la sécurité des données sensibles et privées comme les mots de passe et les numéros de carte de crédit. La menace de la cyber-sécurité devient incontestablement plus grave avec le temps. Selon des estimations récentes, en 2021, les dommages causés par la cybercriminalité coûteront au monde 6 billions de dollars par an, [Cso18] tandis que les dépenses en produits et services de sécurité de l'information passeront à 93 milliards en 2018, selon les dernières prévisions du Gartner, Inc.[Gnr17]

La progression de la cybercriminalité, les menaces terroristes et les préoccupations de sécurité, ainsi que la sensibilité croissante des gouvernements et des entreprises à l'importance des données, les ont amenés à agir et à développer différents outils et méthodes d'investigations numériques contrer ces menaces.

Aujourd'hui, tout ce qui concerne l'examen, l'interprétation ou la reconstruction d'artefacts numériques dans un environnement informatique est considéré comme relevant dans le domaine de l'investigation numérique. L'investigation numérique peut être utilisée dans différents secteurs comme le gouvernement, le secteur privé, les organisations financières et juridiques. Elle fait déjà partie de la planification de la récupération après incident dans de nombreuses organisations.

Dans ce projet de recherche, nous allons concevoir et réaliser un système qui facilitera les investigations numériques sur la mémoire physique. Nous traiterons l'investigation numérique, principalement sur des systèmes Windows, car il s'agit du système le plus utilisé en entreprise.

I.2 Description du problème

En cas de crime numérique, l'objectif principal des experts légistes est d'acquérir et d'analyser des données non volatiles (c'est-à-dire des données qui conserve ses informations en l'absence d'alimentation électrique) de machines suspectes. Mais cette approche n'est plus exhaustive pour les raisons suivantes :

- Ces dernières années, la capacité de stockage des données sur le disque dur des ordinateurs du client final a considérablement augmenté. On se préoccupe de plus en plus de savoir comment augmenter les capacités criminalistiques des outils et des techniques actuellement disponibles pour traiter une quantité aussi énorme de données [Dan09], [Rig06], [Vas04] et [Ted06]. Il en va de même avec la mémoire physique des ordinateurs actuels. Par conséquent, il n'est pas judicieux d'ignorer la mémoire physique de ces énormes capacités dans les enquêtes criminelles numériques.

- Toutes les données temporaires et les informations volatiles telles que les connexions réseau, les logs de chat, les historiques de commandes, les informations de processus et les fichiers ouverts qui sont stockés en mémoire physique seront perdues, si on applique la méthode du " déconnecte la machine". [Suth08] et [Gab07]

- Dans certains cas, les preuves ne se trouvent qu'en mémoire. Par exemple, il existe de nombreux programmes malveillants qui s'exécutent directement depuis la mémoire physique [Mil04], [Imm19], [Cor2019], [Met18], [Der07] sans être installés sur le disque dur, ne laissant aucune trace sur ce dernier.

- Dans le cas de la forensic traditionnelle de disque si l'évidence acquise est chiffrée alors la mémoire physique est le lieu immédiat suivant où on peut trouver les clés pour les données chiffrées.

L'importance des analyses criminalistiques de la mémoire physique est évidente à partir de ces points, mais elle en est encore dans son enfance et nécessite une attention plus sérieuse. Les outils et méthodes actuellement disponibles pour l'analyse des preuves de mémoire sont limités et exigent beaucoup d'attention par rapport à l'importance des données sensibles en mémoire.

I.3 Structure du mémoire

Hormis l'introduction et la conclusion générale qui sont le premier et le dernier chapitre respectivement, ce mémoire est constitué de quatre chapitres organisés comme suit :

Le deuxième chapitre « Investigations digitales » : Ce chapitre est consacré à l'étude de l'investigation numérique et la cybercriminalité, couvrant le processus et les définitions des termes de base, et on inclura également la recherche, la collecte et l'analyse des preuves numériques.

Le troisième chapitre « Investigations digitales sur la mémoire » : Ce chapitre a été consacré à soulignera également plus de détails en comparant les différentes approches et en parvenant ensuite à l'état actuel de l'investigation numérique de la mémoire. On a présenté les logiciels les plus utilisés par les départements d'investigation numérique de la police et la gendarmerie et on a obtenu un tableau de comparatif qui donne un aperçu de chaque outil en se basant sur les fonctionnalités.

Le quatrième chapitre « Conception du système » : Ce chapitre, on présentera notre architecture en prenant en considération les avantages et les limites des modèles forensiques, ce qui peut aider de choisir le modèle le plus approprié pour une tâche d'investigation numérique de la mémoire.

Le cinquième chapitre « Implémentation du système » : Cette partie consiste à présenter l'environnement logiciel sur lequel le système sera réalisé et validé, et ainsi que les détails d'implémentation de notre application. Enfin, On donnera un exemple réel dans un incident de criminalité numérique, de la scène du crime vers laboratoire, jusqu'à la présentation du cas.

Liste des figures

Liste des tableaux

VI.1 Conclusion

Ce domaine de la technologie de l'information connaît une croissance rapide et permet de retracer un crime à l'aide de médias numériques liés à des crimes assistés par ordinateur. A travers ce document nous avons souhaité élaborer un guide pour l'analyste en apportant une méthodologie d'investigation qui lui permettra de suivre un schéma directeur lors d'une enquête.

La Section « analyse de malware » vise, quant à lui, à aiguiller l'analyste dans sa compréhension du fonctionnement d'un logiciel malveillant.

Selon le rapport de Microsoft Security Intelligence [Rev14], l'Algérie est la cible principale de cyber attaques, parmi ces attaques, le piratage informatique. L'Algérie doit élaborer une bonne stratégie dans domaine relativement, pour faire face aux cyber-menaces il faut passer d'abord, par une prise de conscience, l'éducation, la formation et le soutien des internautes, le renforcement du partenariat public-privé, et renforcement des moyens juridique que des ressources humaines.

Le développement des systèmes embarqués, l'arrivée des robots dans nos domiciles mais aussi la bio-informatique sont autant de domaines qui intéressent les cybercriminels de demain. Ainsi, dans un futur proche, il pourra être possible de pirater une voiture pour en prendre le contrôle, de s'introduire dans une maison par le biais d'un robot compromis, ou encore d'arrêter les battements d'un cœur artificiel à distance ou de contrôler un membre robotisé. C'est dans ces environnements vulnérables que l'investigation numérique peut contribuer à la protection de nos données mais également de l'individu.

VI.2 Contribution

Dans ce travail nous avons essayer de résoudre cette problématique en :

- Nous avons proposé un nouveau modèle le plus approprié pour une tâche d'investigation numérique de la mémoire.
- Nous avons créé un outil USB adapté à ce modèle. L'objectif de cet outil est de concentrer des utilitaires de forensic dans une clef USB afin de pouvoir les transporter et les utiliser facilement.

Avantages de l'outil Zodiac Forensic:

- ❖ L'outil d'investigation numérique sur les systèmes Windows, et regroupe plusieurs outils disponibles gratuitement en un seul.
- ❖ La rapidité de la recherche et assister l'enquêteur lors de recherches d'informations sur des systèmes en cours de fonctionnement.
- ❖ Préserver l'intégrité de la preuve et assurer ainsi son authenticité devant les tribunaux.
- ❖ L'outil décrit de façon simple les conclusions et les preuves obtenues au cours d'une enquête criminelle afin de les rendre accessibles à tous.

VI.3 Perspectives

Les perspectives de ce travail pourraient s'énoncer ainsi :

- En raison de contraintes de temps, notre travail n'a pas pu être complété avec certaines des extensions, mais elles peuvent être ajoutées comme travaux futurs à notre rapport. Ajouter d'autres fonctionnalités et lancer la version commerciale.
- Ce mémoire a tenté de donner plus d'informations à un examinateur effectuant une analyse de mémoire Windows©. Il est possible d'étendre ce travail à d'autres systèmes d'exploitation comme Linux et Mac, ce qui est assez nouveau sur le marché. Cela serait très utile à la communauté des investigations numériques.
- Il est important de signaler que l'extraction et la préservation de preuve à partir de données numériques énormes et variées (aussi appelées Big Data) est un problème non résolu qui doit être résolu dans un proche avenir.
- Si l'on peut développer une solution qui combine l'apprentissage machine et l'outil, ce sera une excellente solution.