Role of Hibernation File in Memory Forensics of windows 10

Azad Singh
M.Tech Student, Department of Computer Science & Applications, Kuruksheta University, Kurukshetra-136119
azadmehla@kuk.ac.in
Pankaj Sharma
M.Tech Student, Department of Computer Science & Applications, Kuruksheta University, Kurukshetra-136119
pankajshastri@kuk.ac.in
RajenderNath
Professor, Department of Computer Science & Applications, Kuruksheta University, Kurukshetra-136119
rnath_2K3@rediffmail.com

---ABSTRACT---

Digital media devices are regularly seized pursuant to criminal investigations and Microsoft Windows is the most commonly encountered platform on seized computers. Memory forensics gives the volatile artifacts from the system as they play a significant role in reconstructing the events along with static artifacts from the system storage. Hibernation file is identified as an essential part of digital forensics, which provides analysts with snapshots of system memory from various points in the past. Hibernation file includes web, email and chat sessions in addition to running processes, login credentials, encryption keys, program data and much more. The purpose of this work is to study the hibernation file and page file, there role in memory forensics and to explore current technologies and concept for analysis. This study includes the windows hibernation features, file formats, potential evidence saved to the file and impacts in digital forensic investigations and also compares page file and hibernation file in order to validate the evidences and finding additional artifacts.

Keywords: Hibernation file, Page file, Swap file,Window forensics

1. INTRODUCTION

Microsoft windows dominates the world's desktop operating systems with 90.97% share and windows 10 have market share of 14.15% till march 2016 and is growing at a remarkable pace, still windows 7 holds the a total of 51.89% [1], but soon it will change as Microsoft will terminate its main stream support of windows 7, encouraging users to upgrade to newer versions of windows [2]. The new features added to the windows 10 are notification center, new startup menu, frequent folders, Cortina, synced Wi-Fi hotspots, windows 10 applications (such as Facebook, mail, photos, map and many more), one drive data , single platform for smartphones, tablets, and PCs, new browser named edge, multiple desktops. These features make Windows 10 more intelligent in detecting and responding when users switch in 2-in-1 device -- like Microsoft Surface -- between tablet and laptop modes. The light weighted application programs called apps that work across different computing devices like windows enabled smartphones, tablet, and laptops is continue in windows 10 with better user interaction. These features show that there is a lot of information in the volatile memory of the windows 10 operating system along with the static artifacts that are quite similar to the older versions of windows [3]. Windows systems contain an energy saving feature called hibernation or hybrid sleep. This feature is activated when a system sits idle for a set time or if the laptop lid is closed. Upon activation, the systems memory is copied to the hibernation file, hiberfil.sys, in order to place the system in a lower power state. This file provides memory image to analysts, from the last time the system is hibernated. Memory forensics is crucial in many investigations because it produces the vast amount of evidence which does not exist on the hard drive. Memory images are not always available for analysis to make the hibernation file but it is the only source to retrieve the file.

2. RELATED WORK

Till there is no much prior research work has been carried out on windows 10 forensics and there is a lack of tools which are capable of performing acquisition on windows 10. Hibernation is a power conservation feature built in Windows operating systems beginning with Windows 2000. The advancement of smart battery technology led to the implementation of the hibernation mode in Windows

[4]. Shaver study the forensically valuable areas of the windows 10 registry to find the last date/time of insertion and removal of thumb drive however he does not consider the hibernation file [5]. Murtuza Shariq et al. have investigated the static and volatile artifacts produced by windows 8.x apps from the hibernation and swap files available on storage media, their work focus on the extraction of app specific data only [6].Olazide et al. Present the analysis of user input on volatile memory of windows applications [7].

2.1. BACKGROUND

This section describes the Hibernation and swap file of the windows operating system.

2.1.1 HIBERNATION FILE

The hibernation file, named hiberfil.sys, is a binary file located in the root directory (%SystemDrive%/hiberfil.sys). The file format for hiberfil.sys is a Microsoft proprietary compression whose details have only been made available through reverse engineering [8].Memory forensics is crucial in many investigations because it produces vast amount of evidence which does not exist on the hard drive. Memory images are not always available for analysis making the hibernation file the only source of this data. The hibernation file is also useful even when a memory image is available because it provides an additional data set from a previous point in time. When system backups are leveraged, memory snapshots can be examined from several points in the past. Examining these historical snapshots can help identify malware, encryption keys, login credentials and other valuable evidence contained in memory. The hibernation feature is complex and varies between operating systems and hardware configurations. Understanding these variations can help to produce higher quality evidence. Studies of hiberfil.sys can allow the analyst to perform deep forensics, carving evidence which may not have been obtained from forensic tools. Without a memory image, investigators lose valuable evidence such as Internet history which was cleared or current session usernames and passwords. Encryption keys for full or partial disk encryption may also be extracted from memory. [9] When a system hibernates, the contents of memory are written into hiberfil.sys. Hiberfil.sys is overwritten each time the system hibernates so that just one hibernation file will be present on the system [10].

2.1.2 SWAP FILE

Window 10 includes a new virtual memory file named swapfile.sys. It is stored in your system drive, along with the pagefile.sys and hiberfil.sys. Like pagefile.sys and hiberfil.sys, this file is stored in the root of your system drive — C:\ by default. It's also visible only if you've enabled "Show hidden files and folders" and if you have the "Hide protected operating system files" option disabled. The swap file--swapfile.sys is currently used for swapping out Microsoft's new style of app. Microsoft has called these universal apps, Windows Store apps, Metro apps, Modern apps and other things at various points.

Windows can efficiently write the whole (private) working set of a suspended Modern app to disk in order to gain additional memory when the system detects pressure. This process is analogous to hibernating a specific app, and then resuming it when the user switches back to the app. In this case, Windows 8 takes advantage of the suspend/resume mechanism of Modern apps to empty or re-populate an app's working set. User spend/resume of Metro-style apps is one scenario, there could be others in the future. The swap file and the regular page file have different usage patterns and different requirements with regard to space reservation, dynamic growth, read/write policies etc. Keeping them separate makes things simple [11].

3. STRUCTURE OF HIBERNATION FILE

The hibernation file is referenced as a "file". However it may be better defined as a volume. "A volume is a collection of addressable sectors that an operating system (OS) or application can use for datastorage[12]. The hibernation file is segmented into 4096 byte sections or pages. Memory contents are stored in the file in blocks of data called Xpress image blocks. The first page of hiberfil.sys contains the header. The second page contains the processor state and the third page is where the reserved memory map begins. The remaining pages of the file house are the Xpress image blocks which are organized by memory tables. The exact organization of these blocks differs with Windows versions[13]. The hibernation files top level outline is shown in Table 1: Hibernation File Pages. Each of these segments is explained in more detail later in this section.

Page	First byte	Ending byte	File section
0	0	4095	Header

1	4096	8191	Processor State
2-n	8192	EOF	Reserved Memory Map Memory Tables Xpress Image Blocks

Table 1: Hibernation File pages[13]

Hiberfil.sys header: The first page of hiberfil.sys, page zero, contains the file header, PO_MEMORY_IMAGE. The header contains valuable information, however unless acquisition occurs while the system is hibernating, this data will not be available because it is zeroed out when the system resumes[14]. This provides a challenge for forensic tools because they must be able to parse the data in the hibernation file without the benefit of the header were not defined in existing publications and require further research.

Hiberfil.sys processor state: The second page of hiberfil.sys is the processor state. This page begins at byte offset 4096 and continues to 8191. This structure is platform specific. The processor state contains a set of kernel context and state registers [15]. This structure is exported in Windows debugging symbols and contains the GDT and IDT offsets along with control registers CR0 and CR3[16]. This portion of the hibernation file could provide valuable information about the registry.

How data is stored in hiberfil.sys: It is important to understand how data is written to the hibernation file utilizing the reserved memory map and memory tables. Current research does 15 not detail the exact procedure for writing data to the hibernation file but it stresses that the data is not stored consecutively within the files pages. A large segment of data (over 4096 bytes) would logically require more than one page for storage. Likewise, several smaller blocks of data may be stored in the same page. Pages can contain a maximum of 255 Xpress image blocks [8]. In order to organize this data, the hibernation file has a reserved memory map which keeps track of the available and reserved pages. The data is broken into Xpress blocks and stored in pages with open space. Each page contains its own memory tables to keep track of the Xpress blocks within [15]. These segments of the hibernation file are explained in more detail below.

4. MEMORY FORENSIC TOOLS

In this section, forensic tools which can be used to acquire, convert and analyze the hibernation file are reviewed. The tools included in this section were selected based on price, functionality, user friendliness and the researchers' familiarity with them.

MoonSols Windows Memory Toolkit: MoonSols Windows Memory Toolkit was designed by Suiche, who was the first to publish his reverse engineering of hiberil.sys. This tool is capable of acquiring the hibernation file from a both a live system or a disk image. MoonSols decompresses and converts the hibernation file into a raw memory image. The image can then be analyzed by MoonSols or any othermemory forensic tool. In addition to raw image conversion, MoonSols can also convert hiberfil.sys into a crash dump format[17]. Another feature is the inclusion of win32dd.exe and win64dd.exe which can convert the hibernation file into a raw memory dump with a single click[18].

Volatility: Volatility is a widely used memory forensics tool. Like MoonSols, it provides numerous functionalities required for examining the hibernation file. The Volatility plugin hibinfo is used to identify the hibernation file format. The image copy plugin can be used to convert the hibernation file into a raw memory dump. Once converted, volatility can thoroughly examine the content of the memoryimage. Volatility is capable of analyzing the hibernation file in its native format as well. Using brute force, Volatility locates the data within the hibernation file, even in the absence of the header data[19]. Although Volatility is free and very powerful, the technical barrier to entry is its biggest disadvantage. Volatility is a command line tool requiring analysts to have advanced skills. Additionally, each function in Volatility requires the analyst to learn different plugins. Manually using each plugin to extract data can take longer than an all-in-one tool which will extract the evidence andpresent it to the analyst in an easy to read format. Although it takes longer, an analyst who is proficient with Volatility may be able to extract and understand more evidence from the hibernation file than someone relying on the forensic software to present the evidence to them.

WinHex: WinHex is a hex editor which contains several digital forensics features including the ability to acquire the hibernation file from a live system. WinHex offers a licensed version ranging in price from around fifty U.S. dollars for personal use to thousands of dollars for commercial packages. WinHex takes a snapshot of the content in a live system. The licensed version of WinHex allows the user to copy the hibernation file from that snapshot.The free version will not allow the saving of files larger than 200 KB. WinHex, being a hexadecimal editor, enables the user to view the contents of the hibernation file (X-ways Software Technology, 2015). Although the documentation on WinHex tool not explicitly says it decompresses the hibernation file, viewing the hexadecimal data of the file confirms plain text is seen in the contents. The free version of WinHex can be useful in viewing the hibernation file on a live system. However, a licensed version is needed for acquisition. The searching capabilities of the free version of WinHex are limited and may be useful for locating a specific piece of evidence but not for extracting all the evidence from the file.

5. PROPOSED METHODOLOGY

Static data is present on disk in the form of databases and registry files while some artifacts are present only in the volatile memory such as cortana, notifications window's app e.g. facebook's notifications, news feeds etc., running processes, login credentials, encryption keys and also archival data from the hibernation file. For these reasons, a complete forensic picture may only be available if an investigator consider both static and volatile artifacts. The methodology present in paper can be converted into a tool which incorporates a graphical user interface front-end written in Java; its back-end activities are implemented by a collection of shell scripts which collect volatile artifacts andarrange them according to their timestamp and present a timeline activity in order to assist the digital investigation. The tool extracts the hibernation file from the hard disk/dd image /user provided image and convert it into the corresponding ram dump using MoonSols memory tool kit. RAM dump also taken if an investigator encountera running system, using MoonSols memory toolkit. Then volatility can be used to extract the volatile data from that RAM dumps along with their specific timestamps.

Figure 1 presents the proposed methodology. In case of live system a RAM dump should obtain just before the system was made to hibernate.

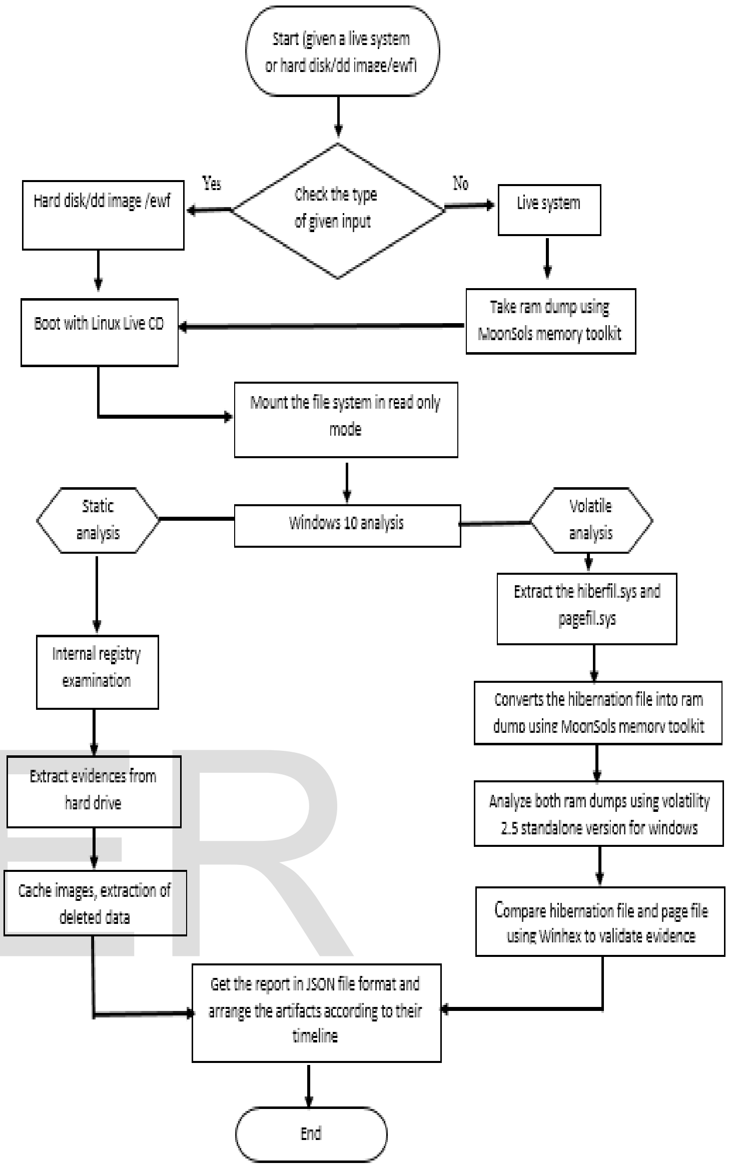
Start (given a live system or hard disk/dd image/ewf)
Check the type of given input
Yes
No
Hard disk/dd image /ewf
Live system
Take ram dump using MoonSols memory toolkit
Boot with Linux Live CD
Mount the file system in read only mode
Static analysis
Windows 10 analysis
Volatile analysis
Extract the hiberfil.sys and pagefil.sys
Internal registry examination
Converts the hibernation file into ram dump using MoonSols memory toolkit
Extract evidences from hard drive
Analyze both ram dumps using volatility 2.5 standalone version for windows
Cache images, extraction of deleted data
Compare hibernation file and page file using Winhex to validate evidence
Get the report in JSON file format and arrange the artifacts according to their timeline
End

Figure 1 Proposed Experimental Methodology

Steps of proposed methodology are as follows:-

i. An investigator may encounter a live system or have a hard drive/dd image/ewf.

ii. Check whether there is live system or any type of image.

iii. In case of live system take the RAM dump first and then boot it with Linux Live CD to get the hibernation file and page file.

iv. Otherwise extracts the hibernation file and page file from the hard disk/ddimage/user-provided image or E01 file using Linux Live CD

v. Converts it to the corresponding RAM dump using the MoonSols memory toolkit.

vi. Take the RAM dump if there is live system and then system made to hibernate and extract that hibernation file and page file, just like in step I.

vii. Volatility is then used to extract the specific forensic artifacts from the RAM dump and the data is presented in the JSON file format which is then parsed and classified according to different process and specific apps.

viii. Compare the data from hibernation file and page file in order to validate the evidences and extraction of hidden processes.

Memory forensics is a dynamic and significant part of digital forensics. Due to memories volatile nature and the vast wealth of evidence it can provide, forensic professionals study memory forensics as an autonomous piece of their education.

6. CONCLUSION AND FUTURE WORK

Windows Systems copy the contents of memory to the hibernation file when the computer enters hibernation mode. Forensic tools are available which convert the proprietary hibernation file into a raw memory image so it can be examined with traditional memory forensic methods. This file is a valuable source of evidence both in the absence of and in addition to a current memory capture. The hibernation file format is proprietary and not wholly documented. Current studies rely on the publications from the reverse engineering of hiberfil.sys. These documents do not define every variable within the structure. Additionally, the structure varies between Windows versions and hardware configurations. Currentstudies contain conflicting reports of hibernation mode behavior and the resultant hiberfil.sys file. These factors demonstrate that current research is inadequate to allow forensic professionals to understand the hibernation file entirely. Further study is needed to accurately define the contents of the hibernation file and to identify variations in its format.

There are many unanswered questions about the hibernation file requiring further study. Many of these issues can alter how acquired evidence is perceived. Chief among these problems is defining what data specifically is written to hiberfil.sys. Further research could contain the particular methodology that includes the acquisition of a memory image followed by an immediate hibernation and the comparison of the two. This methodology would need to be repeated in different Windows versions and in different situations as it depends upon the operating system and hardware. Further research can also be made by defining what happens when the hibernation file is created, deleted or overwritten and how slack space is handled. This paper has a scope which does not include an experimental analysis of the hibernation file format or current forensic tools. The topic of hibernation forensics is still in itsinfancy resulting in limited scholarly resources. This study could have been improved with an experimental section and the ability to acquire commercial forensic tools.

REFERENCE

[1] "Dekstop Operating System Market Share," april 2016. [Online]. Available: https://www.netmarketshare.com/.

[2] "windows/lifecycle," Microsoft, [Online]. Available: http://windows.microsoft.com/en-us/windows/lifecycle.

[3] "windows 10 forensics," Leahy Center for Digital Investigation, 2015.

[4] E. Schwartz, "One-Chip Solution Powers Up Battery Life," Infoworld, Twentieth Anniversary 1978-1998, 20(30), 1998.

[5] J. S. Shaver, "Exposing vital forensic artifacts of USB devices in the Windows 10 registry," California: Naval Postgraduate School, 2015.

[6] S. Murtuza, R. Verma, J. Govindara and G. Gupta, "A TOOL FOR EXTRACTING STATIC AND VOLATILE FORENSIC ARTIFACTS OF WINDOWS 8. x APPS," in *Advances in Digital Forensics XI*, Springer, 2015, pp. 305-320.

[7] F. Olajide, N. Savage, G. Akmayeva and C. Shonireg, ""Digital forensic research — The analysis of user input on volatile memory of Windows application," in *Internet Security (WorldCIS), 2012 World Congress on*, Guelph, ON, 2012.

[8] M. Suiche, "SANDMAN PROJEC T,"

http://sandman.msuiche.net/docs/SandMan_Project.pdf.

[9] E. Casey, Handbook of digital forensics and investigation, Academic Press, 2010.

[10] H. Carvey, Windows forensic analysis DVD toolkit, Syngress, 2007.

[11] B. Morrison, *Windows 8/Windows Server 2012: The New Swap(blogs.technet .com/b/askperf/archive/2012/10/28/windows-8-windows-ser)*, Washington.

[12] B. Carrier, in *File system forensic analysis*, Boston ,MA, Pearson Education, Inc, p. 70.

[13] P. Kleissner, "Hibernation file attack,," Retrieved from http://stonedvienna.com/downloads/Hibernation%20File%20Attack/Presentation.pdf, 2009.

[14] Diablohorn, "Parsing the hiberfil.sys, searching for slack space," Retrieved from https://diablohorn.wordpress.com/2014/12/10/parsing-the-hiberfil-sys-searchingfor-slack-space/, 2014.

[15] P. Kleissner, "Hibernation file format," Retrieved from http://stonedvienna.com/downloads/Hibernation%20File%20Attack/Hibernation%20File%20Format..

[16] "Windows hibernation file for fun 'n' profit Retrieved from," [Online]. Available: http://www.blackhat.com/presentations/bh-usa-08/Suiche/BH_US_08_Suiche_Windows_hibernation.pdf.

[17] "windows-memory-toolkit," moonsols, [Online]. Available: http://www.moonsols.com/windows-memory-toolkit/.

[18] Ligh, H. Michael , Case, Andrew , J. Levy and W. Aaron, The art of memory forensics: detecting malware and threats in windows, linux, and Mac memory, John Wiley \& Sons, 2014.

[19] "memory-forensics-and-analysis-using-volatility," infosecinstitute, [Online]. Available: http://resources.infosecinstitute.com/memory-forensics-and-analysis-using-volatility/.