



A Survey on Various Existing Systems To Analyze Windows Pagefile

ASMI A.S

M Tech in Computer science

with specialization in Cyber Forensics and Information Security ER&DCI Institute of Technology,
C DAC,TRIVANDRUM

KERALA,INDIA asmisalam@gmail.com

Abstract-- A pagefile is a reserved portion of a hard disk that is used as an extension of random access memory for data in RAM that hasn't been used recently. Microsoft windows uses a page file called pagefile.sys, which is a hidden system file. Like physical memory analysis, pagefile analysis is also important and crucial in cyber forensic investigation. This paper aims to give a survey on analysis of pagefile using various existing systems.

I. INTRODUCTION

The technology is developing rapidly in the field of computer industry. The data processing ability has increased, capacity to store data in digital format has expanded dramatically and the access to information has been made easy and most of all these facilities are becoming cheaper for the common people. This trend has also raised a concern in security professionals to keep sensitive and private data like passwords and credit card numbers secure as long as it is being processed in physical memory and to discard it securely when no longer needed to keep away from adversaries and criminals. The analysis of physical memory is very important to find forensically sound artifacts. As physical memory is volatile, analyzing the pagefile is also important. In this

project work, analysis of and page file is to be done in search of sensitive data.

A. Pagefile

A pagefile is a reserved portion of a hard disk that is used as an extension of random access memory for data in RAM that hasn't been used recently. Microsoft windows uses a page file called pagefile.sys. Although Windows supports up to 16 paging files, in practice normally only one is used [3]. Pagefile.sys is a file that is used by Microsoft Windows to store frames of memory that do not currently fit into physical memory. This means the system can run more tasks than the memory has enough physical space for and simply swap frames in and out of the memory as required. Pagefile.sys is a system file that allows running tasks even when the physical memory has been exceeded. Cleaning the pagefile.sys at system shutdown increases the

performance of the system. To clean the pagefile.sys, modifications have to be made to the Clear Page File At Shutdown key value in the Windows registry. The registry can be accessed by the REGEDIT command. Once the changes have been made, the system has to be restarted for the modifications to take effect. Cleaning the pagefile.sys system file also results in a virtual memory extension. Pagefile, stored in System Drive pagefile.sys is a hidden system file. Because the operating system keeps this file open during normal operation, it can never be read or accessed by a user. It is possible to read this file by parsing the raw file system. Data is stored in the paging file when windows determines that it needs more space in physical memory. Because storage locations in the paging file are not necessarily sequential, it is unlikely to find consecutive pages there. Although it is possible to find data in chunks smaller than or equal to 4kb, its the largest. It is possible to carve out files, but as noted the examiner is unlikely to find anything larger than 4KB an examiner can hope for. Windows uses part of your hard drive space as "virtual memory". It loads what it needs to load into the much faster RAM (random access memory) memory, but creates a swap or page file on the hard drive that it uses to swap data in and out of RAM . Pagefile.sys is located on the root of C: drive (or in where the Operating System is installed) and is named as pagefile.sys, but it is a hidden system file so you won't see it unless you



have changed your file viewing settings to show hidden and system files. Pagefile.sys is a windows system file, acts as swap file and was designed to improve performance. Virtual memory (Swap-File) is a substitute for physical memory. More the physical memory the system have, then less virtual memory is needed. Conversely (all other things being equal) the less physical memory needs more virtual memory. Creating multiple swap files on separate drives will increase the ability for the system to read and write information to all swap files. Keeping the swap file off the drive with the system files will allow the system to utilize both controllers at the same time. And having the swap file start at the typical size will require it to grow less, improving performance. When configuring Virtual Memory it is important to understand the uses of it. When an attempt is made to write to the physical memory, the Virtual Memory Manager looks to see if enough physical memory is available. If not enough memory is available; it will page fault old information from the physical RAM to make room for the new information

B. Page File Acquisition Methods

Paging is a memory management scheme used by the operating system to store and retrieve data from secondary storage for use in main memory. Paging is an important part of virtual memory implementation in most currently available operating systems, allowing them to use disk storage for data that does not fit into physical memory. When the physical memory is full and has no more space for the incoming processes then the memory management swaps out pages to an area on the hard disk and retrieve them when needed. On today's computers the swap space ranges between 1.5 to 3 times the physical memory size. The page file by default is located in C:/pagefile.sys or in C:/windows/win386.swp in windows 9x systems. If the capacity of the physical memory is small then there would be more swapping thus giving the investigators opportunity to find evidence there. But in this case the data in swap file will be replaced quickly. On the other hand, as the capacity of the physical memory increases with time the process of swapping decreases. But on the other hand this gives opportunity to reside there for longer time once it is swapped out. Following are the possible tools and techniques to acquire page file on windows systems

Injecting Unsigned Drivers

In live memory acquisition it is not possible to acquire page file data at the same time because the page file is being used by the operating system. But

it is still possible to "access" this file using a specially crafted driver. Incidentally this technique has been used by Joanna to inject an unsigned driver in windows vista64 memory kernel.

PCT Tool

The page collection tool and was developed by Seokhee and his co researchers to acquire page file from a system running Microsoft windows. According to their experiments they were able to copy the contents of 1 GB page file from a running system to an external USB storage in 3 to 4 minutes. But their tool is not available for public use until now.

Using Virtual Environment

As we talked about the virtual products in the hardware acquisition methods this approach can be used to acquire the contents of page file by freezing the virtual machine at any instance. Thus page file can be collected at the same time with the image of physical memory. This gives a more reliable method because there would be almost no changes to the contents of the page file thus maintaining the integrity of the acquired images. There are also some commercial tools available for acquiring the page file of running system. All these tools must be installed on the system unless they are already available on the system. These are: Disk Explore, Forensic Tool kit, X-Ways Forensics and iLook etc.

C. Windows virtual memory

The Windows operating system creates a private virtual memory space for each running process using the memory manager. The memory manager is used to map a process' virtual address space to the system's physical memory or RAM[2]. All modern operating systems use some form of Virtual Memory (VM) system to give each process a large address space while preventing it from gaining access to data belonging to other processes. The basic concept behind virtual memory is relatively simple. For each process, the operating system maps virtual addresses to physical memory because RAM has less storage space and is more expensive than a hard disk. The memory manager creates data structures called page tables which the CPU uses to translate virtual addresses into physical addresses. Each virtual address is associated with a system-space structure called a page table



entry (PTE), which contains the physical address to which the virtual address is mapped.

If the operating system runs out of space in physical RAM, some data is paged out to make room. The paged data is maintained on the system's hard drive. If a page is not mapped into physical memory, the operating system marks the page as invalid. Any access to this page causes a page fault, which then causes the OS to copy the contents of the page from secondary storage into memory.

To implement VM, Windows maintain a large amount of data. It needs to know if data is in RAM or on the disk. Maintaining this information for each byte of an address space would require more memory than the address space itself and for this reason Windows breaks the address space into 4KB pages (or 4MB if large pages are enabled) and maintains this information in page tables. A page table entry (PTE) consists of the physical address of the page if the page is mapped to RAM and also some attributes of the page. The VMM used by Windows XP allows each process to access a full 4GB of virtual addresses, translating those virtual addresses into physical addresses. Within each virtual address space, there is a portion dedicated as user space and a portion reserved for the operating system. User space is for application code, global variables, the process stack, and dynamically linked libraries (DLLs) and it spans from virtual address 0x00000000 to 0x7FFFFFFF. The system address space is accessible by all processes and is for use by the operating system. It ranges from 0x80000000 to 0xFFFFFFFF and contains the necessary information for system management of the virtual memory, including the page directory and the page table entries (PTEs) used for virtual addresses translation.

D. Virtual address translation

Windows uses virtual addresses to abstract the memory storage system from the rest of the operating system and other programs. The operating system presents each program with a large private virtual address space and each time a program references a virtual address the operating system translates that virtual address into a physical address and retrieves the requested data. If the data is not in memory, it loads the data from the disk. During memory analysis, the examiner needs to use this same translation process, but without help from the operating system. Windows on an x86 system uses a two level page table structure to translate virtual addresses to physical addresses. A 32 bit virtual address is interpreted as three separate components—the page directory index, the page table

index, and the byte index—that are used as offsets into the structures that describe page how the page is mapped. The page size and the PTE width dictate the width of the page directory and page table index fields. For example, on x86 systems, the byte index requires 12 bits to describe the location of all 4096 bytes in each page. The page directory index is used to locate the page table in which the PTE is located. The page table index is used to locate the PTE, which, contains the physical address to which a virtual page maps. The byte index indicates the proper address within that physical page.

Address translation is generally a three stage procedure. Every process on a Windows system maintains a Directory Table Base variable. On an x86 systems this value is stored in the CR3 register when the process is running. This value contains the base address of the table of Page Directory Entries (PDE) for that process. For each virtual address, a PDE is specified using a few bits from the original virtual address. The PDE is used to find the base address of a page of Page Table Entries (PTE). The PTE is designated using this base address and some more bits from the original virtual address. The PTE in turn points to the base address of the page in physical memory where the data is stored. The final address in physical memory is the base address of this page plus the remaining bits from the original virtual address. The least significant bit in a PDE or PTE entry is the Valid or V bit. When this bit is one the entry is said to be 'valid' and bits of the entry contain the Page Frame Number (PFN) used in the next part of the address translation. In a PDE, the PFN points to the page containing the PTE. In a PTE, the PFN points to the page containing the memory indicated in the original virtual address. On the other hand, when the V bit is zero the entry is said to be 'invalid' and a different set of rules must be used to find the data in question.

II. EXISTING SYSTEMS

A. Memoryze

It is free memory forensic software that helps incident responders find evil in live memory. Memoryze can acquire and analyze memory images, and on live systems, can include the paging file in its analysis. Memoryze is designed to aid in incident scenarios[10]. However it has many useful features that can be utilized when doing malware analysis. Memoryze is special in that it does not rely on API calls. Instead



Memoryze parses the operating system's internal structures to determine for itself what the operating system and its running processes and drivers are doing. An important thing to keep in mind is that Memoryze actually consists of two components: Memoryze and Audit Viewer. The two tools are divided logically by function: Memoryze for data collection and analysis, and Audit Viewer for presenting and interacting with the collected information. The easiest way to acquire an image is using Memoryze via the command-line. Audit Viewer has a mechanism to capture an image as well, but we'll cover that in a later section. A batch script is included called MemoryDD.bat. MemoryDD generates a settings script and calls memoryze.exe with the proper parameters. Memoryze requires loading a kernel-level driver, giving access to raw memory. No driver, no memory image. Several things can prevent the driver from being loaded, with the most common being not running the tool from an Administrator account. Another common hindrance is anti-virus software. Malware also tries to access memory and as such any anti-malware solutions present may need to be disabled so they don't block driver installation.

According to the specifications from MANDIANT, Memoryze has the following functionalities: Enumerating all running processes. Listing the virtual address space of a given process. Displaying all strings in memory on a per process basis identifying all loaded kernel modules by walking a linked list.

B. WinHex

WinHex is in its core a universal hexadecimal editor from X-Ways Software Technology. It is a commercial tool but the evaluation versions has enough features to examine memory and disk images from most Windows systems [1]. Though it has limitations under windows Vista and later versions. HxD is a hex editor, disk editor, and memory editor for Windows. It can open files larger than 4 GiB and open and edit the raw contents of disk drives, as well as display and edit the memory used by running processes. It can calculate various checksums, compare files, or shred files. WinHex is not a regular editor - it can edit executable files in hex mode showing you even those non-printable characters, such as carriage returns, tabs, and some other special characters. On the other hand, you can perform data analysis from pieces of data recovered via Scandisk or Chkdisk. WinHex has a unique place because of its versatile built-in features. It can easily perform file recovery and undelete tasks by

using its File Recovery utility. Memory editing is a great bonus for gamers, who can cheat by changing some of the values in order to level up, or by boosting up the energy to be used during the game. Besides, it can check your system's physical memory searching for malicious activity. This is truly helpful when you are performing forensic works on the system. With this tool, it can clone any physical media connected to your system. Furthermore, it allows you to choose which sectors to clone, and compare files or full disks. Its permanent deletion utility will give extra privacy when sharing system. WinHex supports deconstructing RAID 0-5 with a maximum of 16 components.

WinHex can be used for key word search on memory and disk images. It can also be used to inspect and edit all kinds of files, recover deleted files or lost data from hard drives.

C. Page Collection Tool

Collecting the pagefile on a live system is cumbersome since Windows operating system has the complete control and protection of it. Thus, it would be useful for an examiner to have a tool is capable of copying a paging file by parsing the raw file system. The Pagefile Collection Tool is aimed at extracting the full pagefile from a live Windows based system, with the purpose to enhance and facilitate a forensic analysis.

Tool is able to extract the pagefile from a live Windows computer system. The fundamental information extracted from a pagefile will be analysed, mentioning also the potential problem of leakage of sensitive information. One of the goals of volatile memory analysis is to reconstruct as many processes as possible which were executed before the collection phase.

The pagefile could enhance this process by carving out the complementary memory pages which were swapped-out by the memory management unit, during the ordinary functioning of the operating system. As a matter of fact, it is certainly necessary and possible, even though not trivial, to determine to which process belongs a given page stored into the pagefile, in order to create an evidence by linking the swapped page with the related



running process.

Page collection tool is aimed at extracting some valuable content, such as passwords, user IDs, credit card numbers, fragments of pictures, keystrokes information, messenger chat logs and contents of recent used files, such as URLs and textual documents. A web password is detectable within a fixed string schema. This permits to create simple filters which can be used to detect this piece of potentially useful data. With the purpose of extracting such information, we can use at least two different, although standard, approaches, that is searching keywords with classical tools, such as strings of Linux, or applying data carving algorithms, such as those used by Scalpel or Foremost. This tool used a filtering algorithm capable of discriminating such coding schema. This

approach is usable, for example, to find passwords, credit card numbers or even keystrokes information. The latter method is certainly applicable, although it suffers from an extremely high number of false positive rate detection, that is the recovered files are misclassified. Indeed, the sequence of memory pages in the pagefile, as is verifiable by observing it, is almost pseudo casual. Thus, even if the header of a known file, such as a, the body is not likely to be recovered, being the sequence of pages very different and highly fragmented. With regard to passwords recovery, according to table 1, if the pagefile size is over 768 Mbytes, the probability to find a password is about 66%. By considering the memory size of the current computer system, it is that it is often above 512 Mbytes. Thus, can guess that sensitive information will be found on the majority of such pagefile.

Table 1. Number of passwords discovered related to the pagefile size.

It is a collection of command line tools and a C library that allows you to analyze disk images and recover files from them. It is used behind the scenes in Autopsy and many other open source and commercial forensics tools. The Sleuth Kit and its browser-based GUI Autopsy is a collection of source free tools developed by Brian Carrier for performing a disk-based investigation. The tool is cross-platform both in operation and execution, and can be run from a CD-ROM or flash drive for live and dead analysis of memory images.

This tool kit features searching for deleted files, time line creation, file system and metadata analysis, hash database search for identification of malicious software and thumbnail listings for easy inspection of evidence files. Autopsy provides a handy front

end to the underlying powerful analysis abilities of Sleuth Kit.

III. CONCLUSION

A pagefile is a reserved portion of a hard disk that is used as an extension of random access memory for data in RAM that hasn't been used recently. Microsoft windows uses a page file called pagefile.sys, which is a hidden system file. Page file contains some valuable information, such as passwords, user IDs, credit card numbers, fragments of pictures, keystrokes information, messenger chat logs and contents of recent used files, such as URLs and textual documents. The paper is a

survey on the existing systems to analyze windows pagefile.

IV. REFERENCES

1. Nisarg Trivedi, "Study on Pagefile.sys in Windows System" IOSR Journal of Computer Engineering (IOSR-JCE),2014
2. Michael Gruhn, "Windows NT pagefile.sys Virtual Memory Analysis",2014
3. Hameed Iqbal, "Forensic Analysis of Physical Memory and Page File",2009.
4. Jared M. Stimson, "Forensic analysis of window's virtual memory Incorporating the system's page-file",2009.
5. Seokhee Lee, Antonio Savoldi, Sangjin Lee and Jongin Lim , "Windows Pagefile Collection and Analysis for a Live Forensics Context",2009
6. Ruichao Zhang, Lianhai Wang, Shuhui Zhang, "Windows Memory Analysis Based on KPCR",2009
7. Jesse D. Kornblum, "Using Every Part of the Buffalo in Windows Memory Analysis",2007
8. <https://digital-forensics.sans.org>
<http://winhex.software.informer.co/m/>