Collecting Sensitive Information from Windows Physical Memory

Qian Zhao, Tianjie Cao
School of Computer, China University of Mining and Technology
Sanhuannanlu, Xuzhou, Jiangsu, 221116, China
National Mobile Communications Research Laboratory, Southeast University
Sipailou No.2, Nanjing, Jiangsu, 210096, China
Email: snowy_1207@163.com, tjcao@cumt.edu.cn

***Abstract*—When investigators are faced with a target system, they want to find sensitive information such as userID and password. Unfortunately, sensitive information can not be found on the hard drive in most cases. Consequently, sensitive information needs to be gathered from physical memory. In our research, we have found lots of sensitive information from physical memory by different techniques. Besides userID and password, we also have found QQ-chat logs that never have been referred in other papers.**

***Index Terms*—memory forensics, sensitive information, live system**

I. INTRODUCTION

According to RFC 3227 [1] (Guidelines for Evidence Collection and Archiving), which is well-known standard of evidence collection, there is a rule for collecting the volatile information since such volatile information has a different order of volatility. Order of volatility from RFC 3227 is as follows: Registers, cache; Routing table, arp cache, process table, kernel statistics, memory; Temporary file systems; Disk; Remote logging and monitoring data that is relevant to the system in question; Physical configuration, network topology; Archival media. We could see that the first item of volatile data that should be collected on a live system is the contents of physical memory, commonly referred to as RAM.

Compared with other branches of digital forensics, memory forensics is still at an infancy stage. It aims at gathering information from the contents of a computer's physical memory. Memory forensics is not an easy task, because we must collect evidence from a live system. As a result, we need to keep Locard Exchange Principle [2] in mind. It means it is impossible to take a forensically sound snapshot of the memory system being observed without altering it. When an investigator interacts with a live system, changes will occur to that system as programs are executed and data is copied from the system. These changes might be transient (process memory, network connections) or permanent (log files, Registry entries) [3].

Sensitive information, which is important for forensics investigators, also has volatility. If we want to gain sensitive information, we have to collect physical memory firstly. Collecting the contents of physical memory on a live system is not easy. It should maintain as small a footprint as possible on the system. There are several ways have been identified. Each of them has its own advantages and drawbacks and none is suitable for all cases.

The main idea of "Hardware Devices" is to access physical memory through a dedicated communication port by means of a physical device (e.g., a PCI card). This would allow an investigator to retrieve the volatile memory from the system without introducing any new code or relying on potentially untrustworthy code to perform the extraction. In February 2004, Brian and Joe [4] presented the concept for a hardware expansion card dubbed Tribble that could be used to retrieve the contents of physical memory to an external storage device. The authors also stated that they had built a proof-of-concept Tribble device, designed as a PCI expansion card that could be plugged into a PC bus. Other hardware devices are available that allow you to capture the contents of physical memory and are largely intended for debugging hardware systems. These devices may also be used for forensics. Hardware devices are easily accessible and have no impact on the live system. Using hardware devices to dump the contents of physical memory doesn't have new or additional program to be loaded into memory. But the hardware has to be installed prior to the incident. Moreover it is not widely available on the general market today.

FireWire (a.k.a IEEE 1394) bus supports direct memory access, meaning it can access system memory without having to go through the CPU. Memory mapping is performed in hardware without going through the host operating system, allowing for high-speed, low latency data transfers. This technique requires investigators have a controller device that contains the appropriate software and is capable of writing a command into a specific area of the FireWire device's memory space. Same as hardware devices, firewire bus also has the ease of use and the null impact on the system. And many systems available today have FireWire /IEEE 1394 interfaces built right into the motherboards. Also, code has been released

for directly accessing physical memory on Linux and Mac OS systems [3]. However, Arne Vidstrom has pointed out some technical issues that fire wire bus presents problems with a region of the memory called Upper Memory Area (UMA) [5]. Furthermore, in some cases this method causes hardware to malfunction, even Blue Screens of Death on some target Windows systems.

Data Dumper (DD) is a common used tool for acquiring an image of physical memory from Unix. DD has long been considered a standard for producing forensic images, and most major forensic imaging/ acquisition tools as well as analysis tools support the dd format. It is able to collect the contents of physical memory by accessing the \Device\PhysicalMemory object from user mode. The modified version of DD that runs on Windows systems and can be used to dump the contents of physical memory from Windows 2000 and XP systems is available within the windows forensic tools images developed by Garner [6]. DD might be the best method for retrieving the contents of physical memory. It does not require rebooting the system or restrict how and to where the contents of physical memory are written. Further, tools have been developed and made freely available to parse the contents of these RAM dumps to extract information about processes, network connections, etc. But it also has some drawbacks. Memory collection can last a long time (say several hours). As it is a user space solution, an attacker can hook several places in order to tamper with collected data. Another problem is that "\Device\PhysicalMemory" device is not available any more from user space since Windows 2003 SP1.

Some virtualization products such as VMware [7], allow the creation of pseudo-networks utilizing the hardware of a single system. This technique allows investigators to create a snapshot of the target system and to perform all manner of testing. In Vmware's case, when "suspending" the virtual machine, physical memory image is written to a ".vmem" file. Suspending a VMware session is quick, easy, and minimizes the investigator's interaction with and impact on the system. The whole system activity is frozen during acquisition. Thus, the acquired data is fully consistent. However, virtualization technologies are not widely used in systems.

In 2007, Bradley Schatz [8] presented "BodySnatcher", which has demonstrated proof of concept by acquiring memory from Windows 2000 operating systems. This idea is to inject an independent, acquisition specific OS into the potentially subverted host OS kernel, snatching full control of the host's hardware. This method is fidelity, reliability and doesn't require specific hardware. But it also has many limitations. For example, loader components for the alternative OS have a significant impact on the existing OS memory; the alternative OS has to support existing hardware on the target; etc.

Besides the above techniques, there are other methods to collect physical memory. Our research is focused on opening RAM directly, crash dump, pagefile and hibernation. By these methods, we have found a lot of sensitive information.

II. GATHERING SENSITIVE INFORMATION FROM RAM DIRECTLY

There are some tools can provide access to physical memory and other processes' virtual memory, such as WinHex [9]. WinHex is in its core a universal hexadecimal editor, particularly helpful in the realm of computer forensics, data recovery, low-level data processing, and IT security.

RAM can be opened directly by WinHex. It is a very simple way to collect and analyze RAM. After opening RAM, some information of physical memory is also given. Although using this tool can make RAM change, lots of sensitive information can be found. For example, in our research we have found the process of a user changing his password.

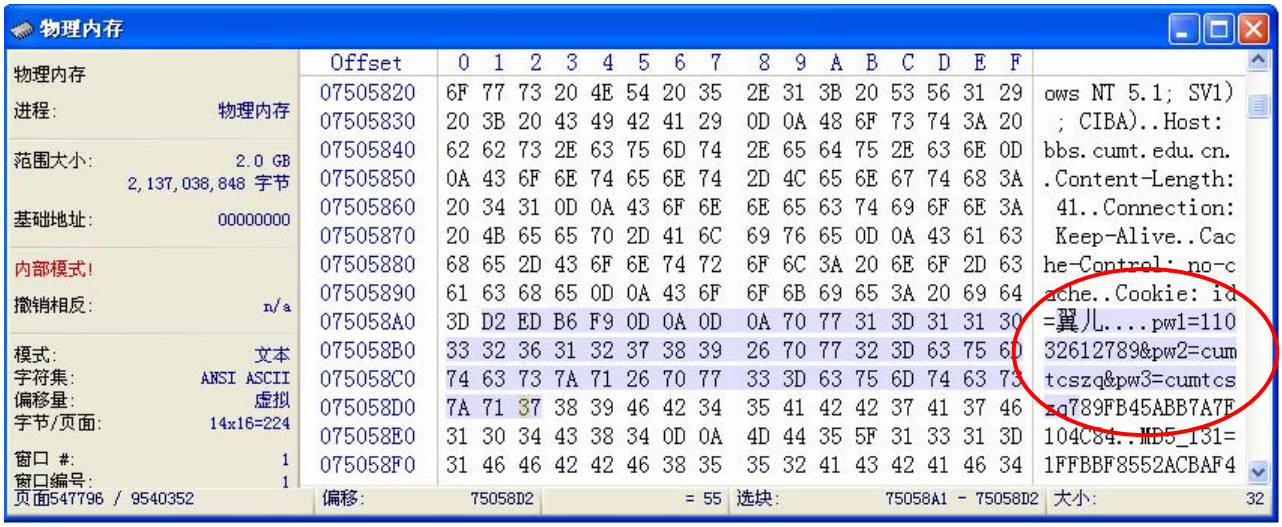

Figure 1. The process of a user changing his password.

In Fig. 1, we can see that the web site which the user logged on is "bbs.cumt.edu.cn", the userID is "yier (Chinese character)", "pw1=11032612789" means the old password is "11032612789", "pw2=cumtcszq" means the new password is "cumtcszq" and "pw3= cumtcszq" means repeating the new password to make sure of password changing.

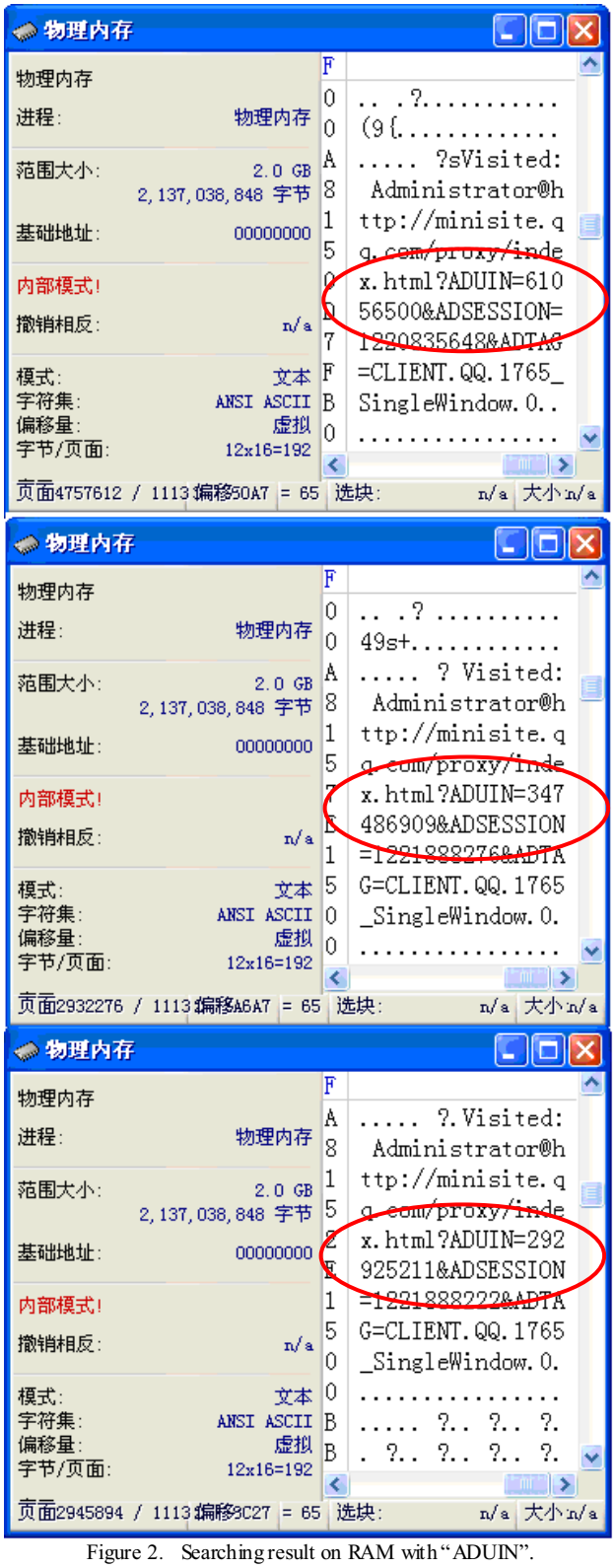

Figure 2. Searching result on RAM with "ADUIN".

Let's take a look at another example. We started QQ.exe, and logged in with three QQ numbers: 61056500, 292925211 and 348486909. Then we chatted with QQ-friends, browsed "Qzone" and did something else as usual. After closing all of the QQ processes, we opened RAM by WinHex. We searched "AUDIN" on RAM. We found all of QQ-numbers that had logged in this system (Fig. 2). And then we searched one of the numbers "61056500", we found a lot of information, such as the user's QQ-friends, contents of the user's "Qzone" (Fig. 3) and even the user's chat logs (Fig. 4).

According to Locard exchange principle, opening RAM directly can change RAM. But it is very easy for investigators to approach physical memory.

III. GATHERING SENSITIVE INFORMATION FROM CRASH DUMP FILE

A crash dump can be the result of instability on the system. In most cases it manifests itself as an infamous Blue Screen of Death. In this way we can obtain a pristine, untainted copy of the content of RAM from a "live" Windows system. When a crash dump occurs, the system state is frozen and the contents of RAM are swapped or copied to the disk. This preserves the state of the system and ensures that no alterations are made to the system, beginning at the time the crash dump was initiated.

According to MS KnowledgeBase (KB) article Q254649 [10], there are three types of crash dump: small (64KB), kernel, and complete crash dumps. What we're looking for is the complete crash dump because it contains the complete contents of RAM.

MS KB article Q244139 [11] describes how to induce crash dump:

1) Select memory dump file options to complete memory dump.

2) Start Registry Editor,Locate the following registry subkey:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\i8042prt\Parameters,

On the Edit menu, click Add Value, and then add the following registry entry:

Name: CrashOnCtrlScroll, Data Type: REG_DWORD, Value: 1;

Exit Registry Editor, and then restart the computer.

3) Press "Right Ctrl + ScrollLock" twice.

The output file, with a ".DMP" extension, is written in a Microsoft-proprietary file format, legible to Microsoft debugging tools.

In our research, we used the method has been introduced to induce crash dump. We found "C:\WINDOWS\MEMORY.DMP", and used some tools such as dumpchk.exe (a program of Debugging Tools for Windows [12]) to verify it.

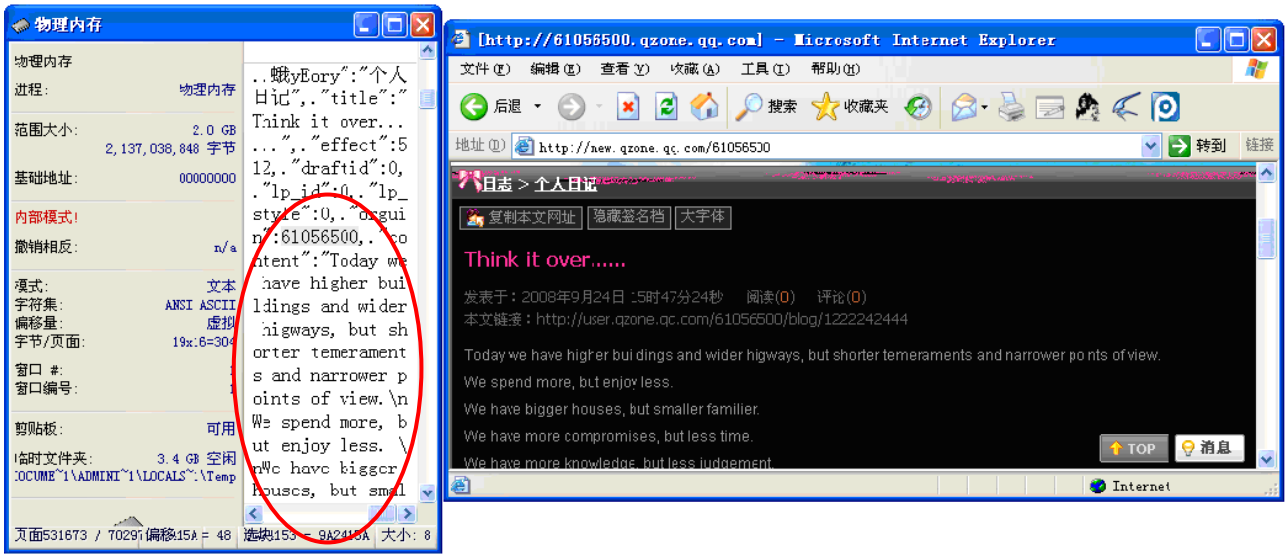

Figure 3. The contents of the user's "Qzone".

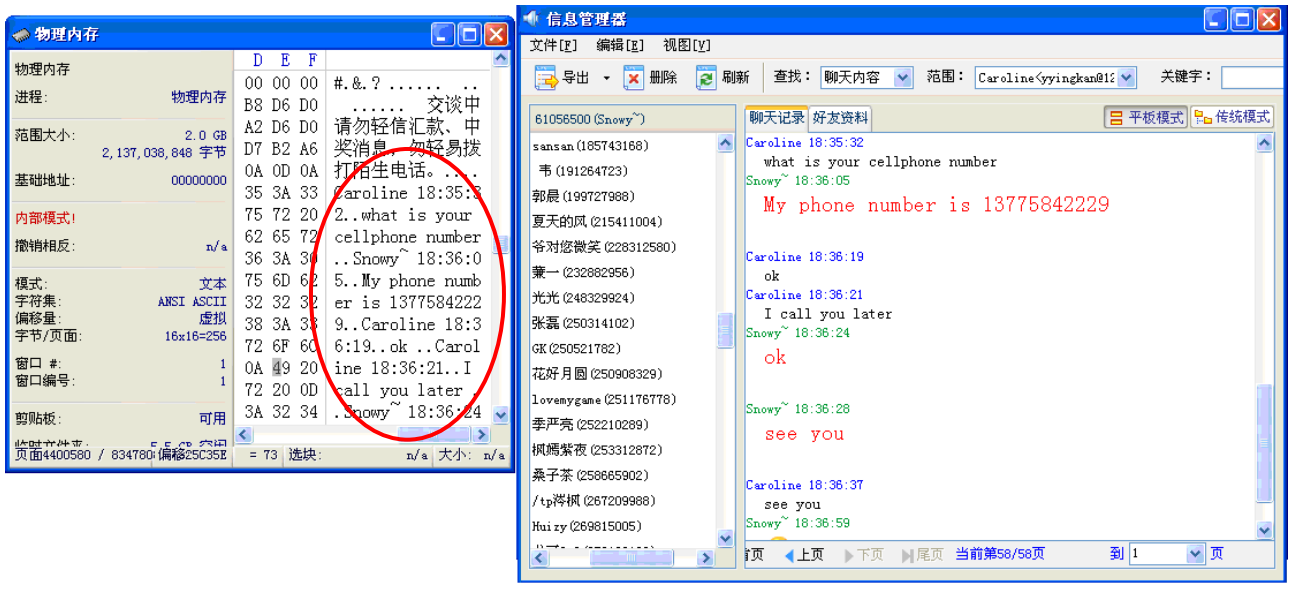

Figure 4. The user's chat logs on RAM.

By dumpchk.exe, we can obtain "32 bit Kernel Full Dump Analysis" (Fig. 5). It includes much useful information such as structure of DUMP_HEADER, physical memory description, version of the operating system, kernel base and unloaded modules.

In our research, we have used other tools to run strings searching on MEMORY.DMP. We opened MEMORY.DMP by UltraEdit [13], and searched "bbs". From Fig. 6, we can see all of bulletin board systems that have been logged in by users of this computer and some contents of them.

We also used WinHex to perform a string searching on MEMORY.DMP with "password". As a result, we discovered username and password of E-Mail. This is an E-mail of "sina.com.cn". In Fig. 7, we can see that "username=snowysd" and "password=sd1207".

```
----- 32 bit Kernel Full Dump Analysis

DUMP_HEADER32:
MajorVersion        0000000f
MinorVersion        00000a28
……
Physical Memory Description:
Number of runs: 3 (limited to 3)
        FileOffset  Start Address  Length
         00001000     00001000     0009d000
         0009e000     00100000     00eff000
         00f9d000     01000000     7e680000
Last Page: 7f61c000     7f67f000

KiProcessorBlock at 8055c580
 2 KiProcessorBlock entries:
 ffdff120 ba338120

Windows XP Kernel Version 2600 (Service Pack 2) MP
(2 procs) Free x86 compatible

Product: WinNt, suite: TerminalServer SingleUserTS
Built by: 2600.xpsp_sp2_gdr.070227-2254
```

```
Kernel base = 0x804d8000 PsLoadedModuleList =
0x8055d700
Debug session time: Sat Aug 16 21:28:34.203 2008
(GMT+8)
System Uptime: 0 days 1:11:25.884
start    end        module name
804d8000 806e3000   nt       Wed Feb 28 16:38:53 2007
(45E53F9D)
806e3000 80703d00   hal      Mon Oct 30 17:50:16 2006
(4545CAD8)
a7db8000 a7de2180   kmixer   Wed Jun 14 16:47:45
2006 (448FCD31)
……

Unloaded modules:
a806e000 a8099000   kmixer.sys
    Timestamp: unavailable (00000000)
    Checksum:  00000000
a8aec000 a8b17000   kmixer.sys
    Timestamp: unavailable (00000000)
    Checksum:  00000000
……
```

Figure 5. 32 bit Kernel Full Dump Analysis by dumpchk.exe.

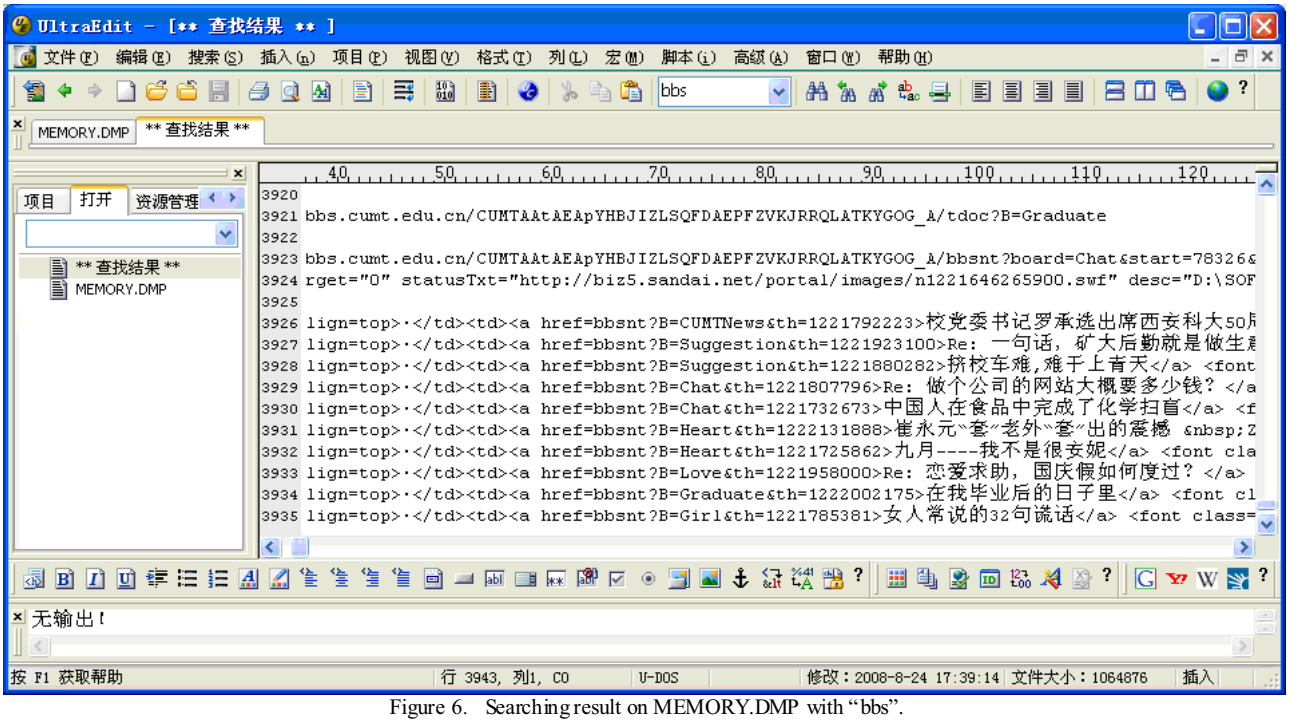

Figure 6. Searching result on MEMORY.DMP with "bbs".

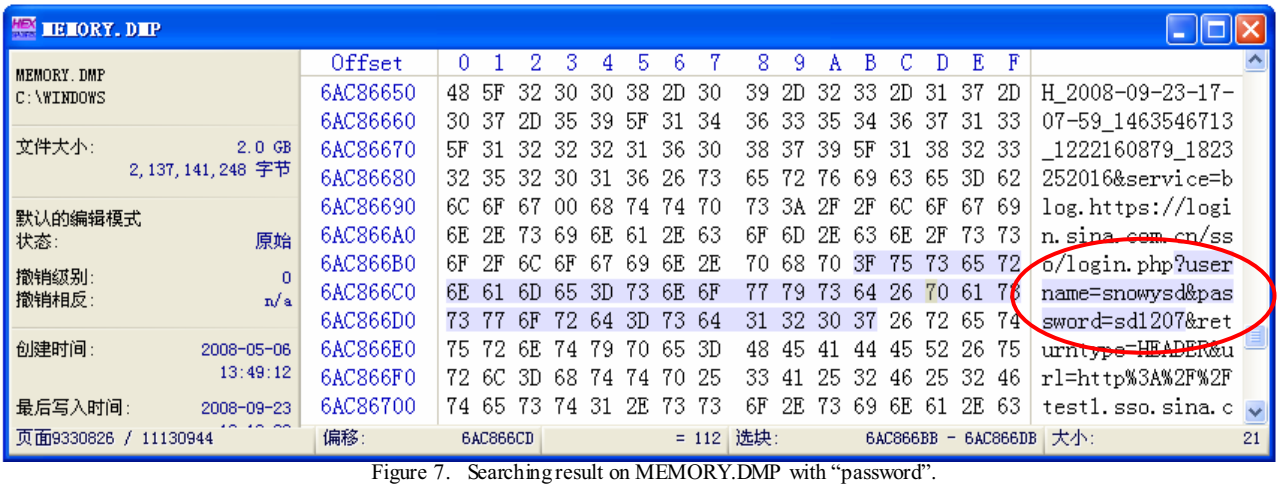

Figure 7. Searching result on MEMORY.DMP with "password".

The crash dump solution is perhaps the only technically accurate method for creating an image of the contents of RAM. However, this method is not perfect. Windows systems do not generate full crash dumps by default (Table I). The registry setting requires a reboot to be taken into account. It must have been set prior to the incident. The crash dump process will still create a file equal in size to physical memory on the hard drive. To do so, as stated in KB article Q274598 [14], the pagefile must be configured to be equal to at least the size of physical memory plus 1 MB. Furthermore, given pagefile limits, the maximum dump file size is 2GB. No complete dump can be obtained on a system with more than 2GB of physical memory [15].

TABLE I. DEFAULT DUMP TYPE OPTIONS

The version of operating system	Default dump type options
Windows 2000 Professional	Small memory dump (64 KB)
Windows 2000 Server	Complete memory dump
Windows 2000 Advanced Server	Complete memory dump
Windows XP (Professional and Home Edition)	Small memory dump (64 KB)
Windows Server 2003 (All Editions)	Complete memory dump

IV. GATHERING SENSITIVE INFORMATION FROM PAGEFILE

The pagefile is a hidden file on a computer's hard disk that Windows XP uses as if it were RAM. On a "live" system, part of the memory is swapped out into the pagefile. Collecting the pagefile is required for complete analysis. The default pagefile is "C:\pagefile.sys".As long as Windows is running, the pagefile is locked by the kernel. It is still possible to access this file using a specially crafted driver, or the special device "\Device\ PhysicalDrive". Incidentally, this technique has been used by Joanna Rutkowska to inject unsigned drivers in Windows Vista64 memory, up to RC1 release candidate [16]. Commercial tools that are reputedly able to copy the pagefile of a running system are: Disk Explorer [17]; Forensic Toolkit [18]; WinHex/X-Ways Forensics [19]; iLook [20] (free for US government officials). In 2005, S. Lee, H. Kim, S. Lee, and J. Lim presented the Pagefile Collection Tool (PCT) in their paper, which can be used to obtain a pagefile on a live Windows based system [21].

In our research, we used WinHex copy "pagefile.sys". Then we used WinHex to search "password" on it (Fig. 8).

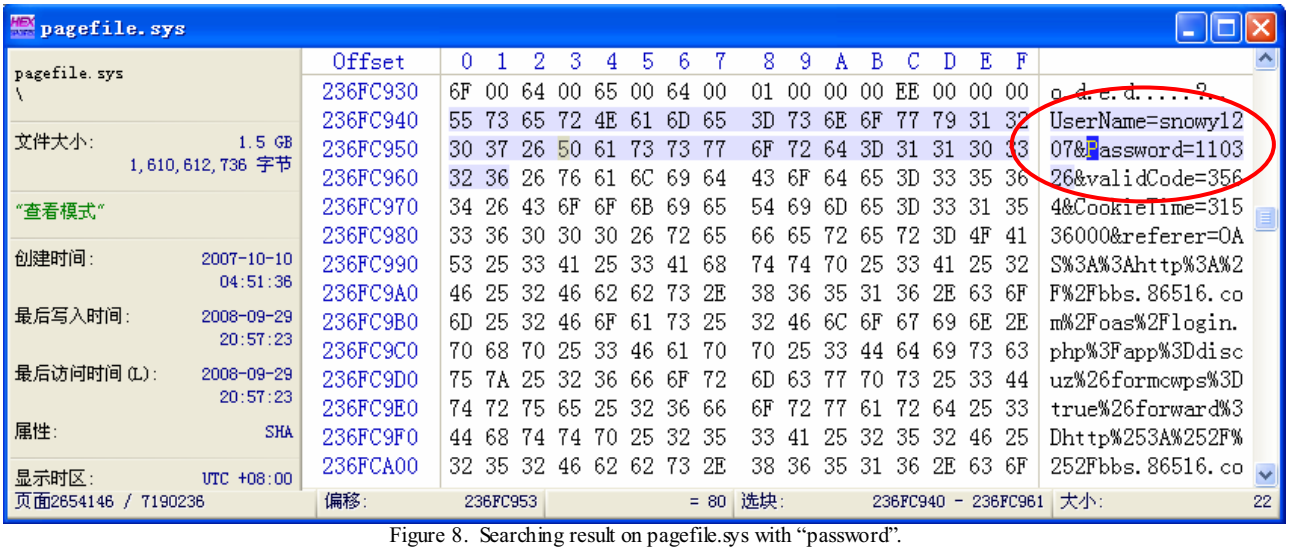

Figure 8. Searching result on pagefile.sys with "password".

The pagefile is not a memory dump file. However, in order to recover as many memory pages as possible, it would be valuable to have access to the pagefile for a Windows based system. Then the physical memory and pagefile data should be merged into a single set of data.

V. GATHERING SENSITIVE INFORMATION FROM HIBERNATION FILE

Hibernation means the whole system state is backed up to hard drive and the system is frozen for a infinite amount of time without any power source. The Power Manager saves the compressed contents of physical memory to a file called Hiberfil.sys in the root directory of the system volume. Hiberfil.sys is created when the system hibernates for the first time. This file is large enough to hold the uncompressed contents of physical memory, but compression is used to minimize disk I/O and to improve resume-from-hibernation performance. During the boot process, if a valid Hiberfil.sys file is located, the NT Loader will load the file's contents into physical memory and transfer control to code within the kernel that handles resuming system operation after hibernation.

In our research, we also used WinHex copy hiberfil.sys and search "pw" on it (Fig. 9).

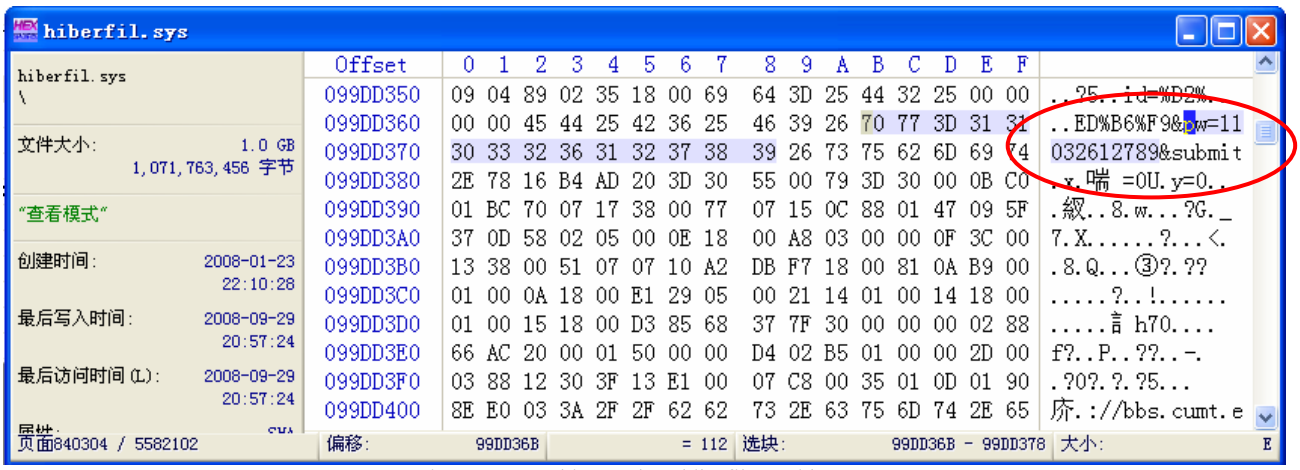

Figure 9. Searching result on hiberfil.sys with "pw".

If hibernation has previously been used on the target system, this memory collection method could prove to be a useful idea. Analyzing the contents of a hibernation file could give us a clue as to what was happening on the system at some point in the past. But if the last hibernation occurred long time ago, the hibernation file is significantly out of date. Further, the hibernation file is using an undocumented, Microsoft proprietary file format, including proprietary compression. Moreover, the hibernation file is most often found on laptop systems. We couldn't find it on the system of a desktop computer.

VI. CONCLUSIONS

By now it should be clear that we have several options for collecting physical memory from a live system. Each of them has its own advantages and limitations. In order to recover as much sensitive information as possible, it is better to merge crash dump and pagefile to collect physical memory. However, there is much more volatile information can be gathered from physical memory, such as information of running processes. So we can make a deeper research in physical memory relevant fields in farther work.

ACKNOWLEDGMENT

This work is supported by the Jiangsu Provincial Natural Science Foundation of China (BK2007035), the open research fund of National Mobile Communications Research Laboratory, Southeast University (W200817) and the Science and Technology Foundation of CUMT (0D080309).

REFERENCES

[1] RFC 3227, Guidelines for Evidence Collection and Archiving. http://www.faqs.org/rfcs/rfc3227.html, 2002

[2] W. Chisum and B. Turvey, "Evidence dynamics: Locard's exchange principle and crime reconstruction", *Journal of Behavioral Profiling*, vol.1, January 2000.

[3] H. Carvey, *Windows Foreniscs Analysis DVD Toolkit*, 2005.

[4] B. D. Carrier and J. Grand. "A hardware-based memory acquisition procedure for digital investigations", *Journal of Digital Investigations*, vol. 1, pp. 50-60, February 2004.

[5] http://ntsecurity.nu/onmymind/2006/2006-09-02.html

[6] G. M. Garner, Forensic Acquisition Utilities, http://www.gmgsystemsinc.com/fau/.

[7] http://www.vmware.com/

[8] S. Bradley, "BodySnatcher: Towards reliable volatile memory acquisition by software", *Journal of Digital Investigations*, vol. 4, pp. 126-134, September 2007

[9] http://www.x-ways.net/winhex/index-m.html

[10] "Overview of memory dump file options for Windows Server 2003, Windows XP, and Windows 2000", http://support.microsoft.com/kb/254649/en-us

[11] "Windows feature lets you generate a memory dump file by using the keyboard", http://support.microsoft.com/kb/244139/en-us

[12] http://www.microsoft.com/whdc/devtools/debugging/default.mspx

[13] http://www.ultraedit.cn/

[14] "Complete memory dumps are not available on computers that have 2 or more gigabytes of RAM", http://support.microsoft.com/kb/274598/en-us

[15] N. Ruff, "Windows memory forensics", *Journal of Computer Virology*, vol. 4, pp. 83-100, May 2008.

[16] J. Rutkowska, "Subverting Vista Kernel", http://invisiblethings.org/papers/joanna%20rutkowska%20-%20subverting%20vista%20kernel.ppt

[17] http://www.runtime.org/

[18] http://www.accessdata.com/catalog/partdetail.aspx?partno=11000

[19] http://www.x-ways.net/forensics/index-m.html

[20] http://www.ilook-forensics.org/

[21] S. Lee, H. Kim, S. Lee, and J. Lim, "Digital evidence collection process in integrity and memory information gathering", In Proceedings of Systematic Approaches to Digital Forensic Engineering, First International Workshop, Proc. IEEE, pp. 236-247, 2005.

Qian Zhao is currently working toward the Master degree in the School of Computer Science and Technology, China University of Mining and Technology.

Tianjie Cao received the BS and MS degree in mathematics from Nankai University, Tianjin, China and the PhD degree in computer software and theory from State Key Laboratory of Information Security of Institute of Software, Chinese Academy of Sciences, Beijing, China. He is a professor of computer science in the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China. From 2007 to 2008, he has been a visiting scholar at the Department of Computer Sciences and CERIAS, Purdue University. His research interests are in security protocols and network security.