



# **INTERNATIONAL CYBERSECURITY AND DIGITAL FORENSICS ACADEMY**

**Assignment Title:** Memory Forensics

**Course Code:** CI901 Cybercrime Investigations Case Studies

**Student Name:** Ahtisham Tanveer

**Student ID:** 2025/ACI/9979

**Programme:** Advance Cybercrime Investigations

**Instructor Name:** Aminu Idris

**Date of Submission:** 01/10/2026

## 1. Environment Setup and Methodology

To conduct this investigation, a specialized forensic environment was established using the **Volatility 2 Framework**.

- **Environment Configuration:** The analysis was performed on a Linux-based system (typically Kali Linux or a similar forensic workstation) where Volatility 2 was installed. This framework was chosen for its robust ability to parse Windows memory artifacts using specific profiles.
- **Memory Source:** The primary evidence file, `memdumpWin7.mem`, was acquired from a target Windows machine. This file is a raw image of the system's RAM at a specific point in time, captured using a memory acquisition tool (such as FTK Imager or DumpIt) while the system was still powered on. This allows for the analysis of "volatile" data that would otherwise be lost if the computer were shut down.

## 2. Forensic Topics and Analysis Summary

### *Memory and Memory Dumps*

A memory dump is the process of capturing all information currently held in RAM and writing it to a storage drive. These binary files often referred to as core dumps or the "Blue Screen of Death" (BSOD) in Windows are essential for forensic investigators to identify the runtime state of a machine.

### *Virtual Memory and Pagefile.sys*

Windows utilizes virtual memory to allow processes to use more memory than is physically available. When physical RAM is exhausted, the system offloads data to `Pagefile.sys` on the disk. Analyzing this helps recover data from inactive applications.

### *Kernel Processor Control Region (KPCR)*

The KPCR is a kernel-level data structure that stores information for each processor. By locating the KPCR within a memory dump, Volatility can traverse the system's architecture to find the list of active processes and threads.

### *The Volatility Framework*

Volatility is an open-source framework for extracting digital artifacts from volatile memory. It supports Windows, Linux, and iOS, allowing investigators to reconstruct system activity, such as open network connections, running processes, and loaded DLLs.

### *Registry Analysis and SIDs*

The Windows Registry stores configuration data for the system and users. Every user is assigned a unique Security Identifier (SID). Analyzing registry hives within memory helps link specific actions or files to a particular user account.

### *Network Forensics*

By scanning memory for network objects, we can identify active connections and suspicious IP addresses. Using specialized plugins, we can determine which Process ID (PID) was responsible for specific network traffic, such as a browser downloading a malicious file.

### *Command History Investigation*

Plugins like `cmdscan` and `consoles` are used to extract the history of commands entered into the command prompt. This provides a direct look at the suspect's actions, such as file deletions or unauthorized data transfers.

### *USB Forensics and Instance IDs*

When a USB device is connected, Windows creates registry entries and assigns a unique Instance ID. This allows investigators to prove a specific external drive was plugged into the computer, including the exact time and assigned drive letter.

### *Shellbags and File Explorer History*

Shellbags track folder viewing preferences in File Explorer. These are critical for showing that a user navigated through specific folders, even if those folders were later deleted or resided on a removable USB drive.

### *Timeline Analysis*

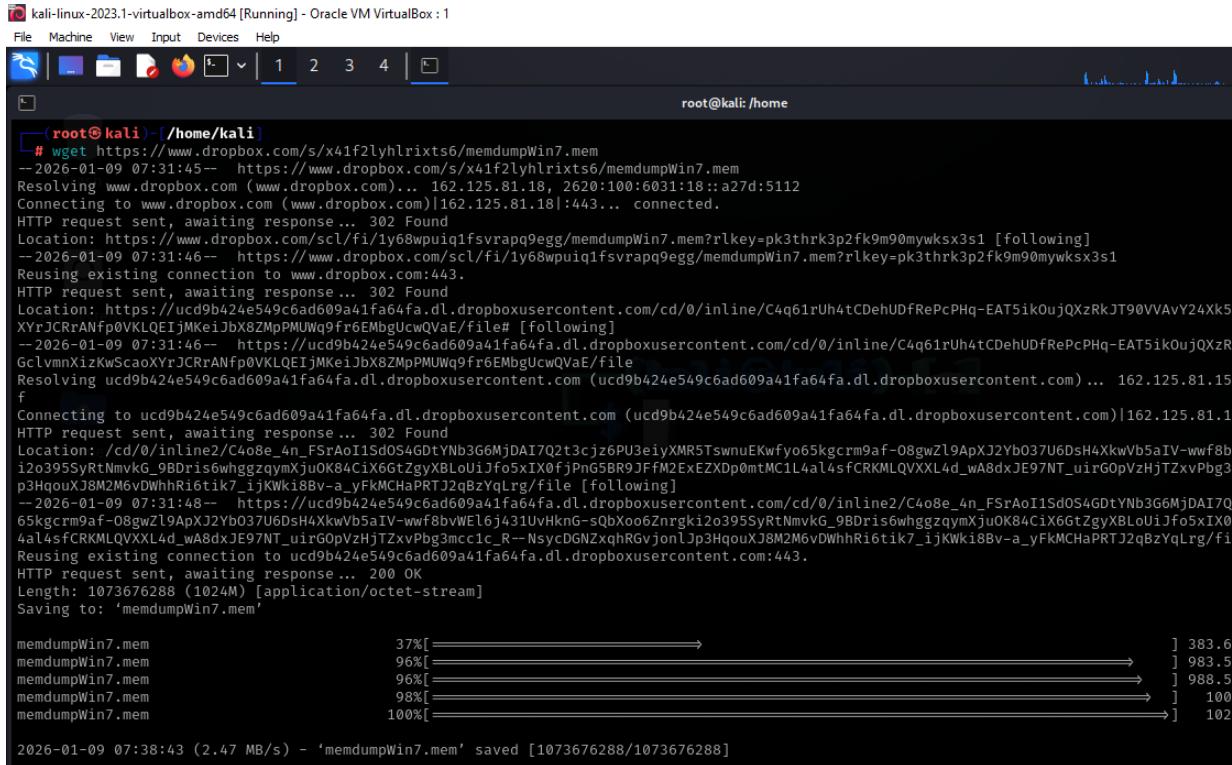
This is the process of merging various artifacts browser history, file access, and USB connections into a single chronological list. It allows for the reconstruction of a suspect's "story," such as downloading a sensitive document and immediately moving it to a flash drive.

### *Password Cracking and Hash Dumping*

Memory forensics allows for the extraction of NTLM password hashes from the `LSASS` process in RAM. These hashes can then be processed through tools like John the Ripper to recover the actual plaintext passwords of the system users.

# PRACTICAL

Download the memory image for investigations:



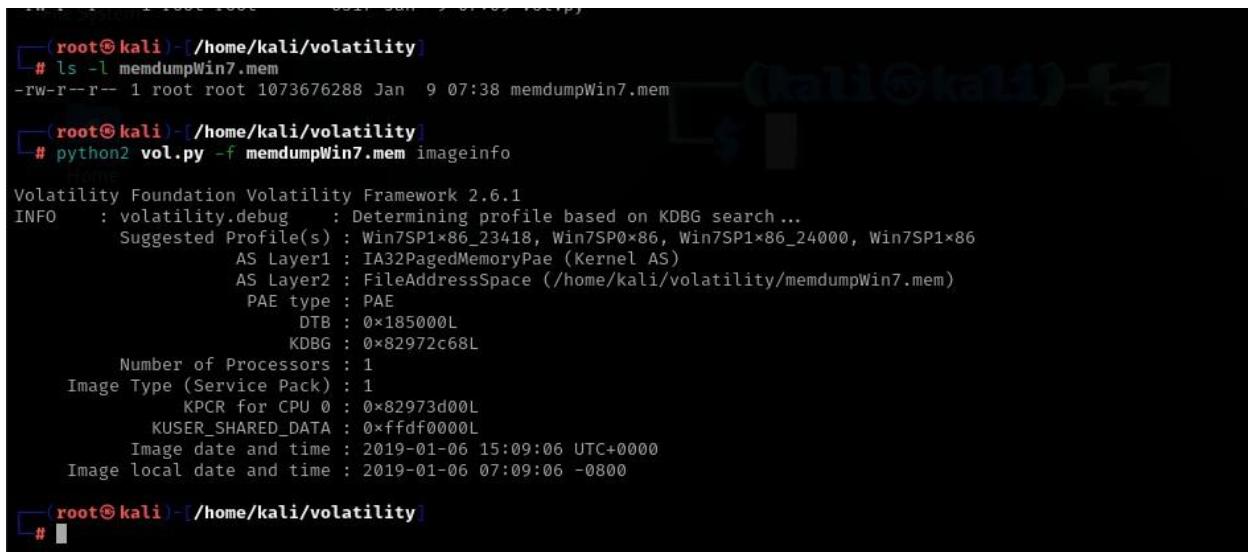
The screenshot shows a terminal window titled "kali-linux-2023.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox : 1". The command `# wget https://www.dropbox.com/s/x41f2lyhlrixts6/memdumpWin7.mem` is being run. The output shows the progress of the download, which is approximately 102 MB in size. The download is completed successfully at 07:38:43 on January 9, 2026.

```
(root㉿kali)-[~/home/kali]
# wget https://www.dropbox.com/s/x41f2lyhlrixts6/memdumpWin7.mem
--2026-01-09 07:31:45-- https://www.dropbox.com/s/x41f2lyhlrixts6/memdumpWin7.mem
Resolving www.dropbox.com (www.dropbox.com)... 162.125.81.18, 2620:100:6031:18::a27d:5112
Connecting to www.dropbox.com (www.dropbox.com)|162.125.81.18|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://www.dropbox.com/scl/fi/1y68wpuiq1fsvrapp9egg/memdumpWin7.mem?rlkey=pk3thrk3p2fk9m90mywksx3s1 [following]
--2026-01-09 07:31:46-- https://www.dropbox.com/scl/fi/1y68wpuiq1fsvrapp9egg/memdumpWin7.mem?rlkey=pk3thrk3p2fk9m90mywksx3s1
Reusing existing connection to www.dropbox.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com/cd/0/inline/C4q61rUh4tCDehUDfRePcPHq-EAT5ikOujQXzRkJT90VVAvY24Xk5XYrJCrAnFp0VKLQEijMKeiJbx8ZMpPMUWq9fr6EMbgUcwQVaE/file [following]
--2026-01-09 07:31:46-- https://ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com/cd/0/inline/C4q61rUh4tCDehUDfRePcPHq-EAT5ikOujQXzRGC1vmnXizKwScaoXYrJCrAnFp0VKLQEijMKeiJbx8ZMpPMUWq9fr6EMbgUcwQVaE/file
Resolving ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com (ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com) ... 162.125.81.15
f
Connecting to ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com (ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com)|162.125.81.1
HTTP request sent, awaiting response... 302 Found
Location: /cd/0/inline2/C408_e_4n_FSrAoI1Sd0S4GdtN3G6MjDAI7Q2t3cjz6PU3eiyXMR5TswnuEkwfyo65kgcrm9af-08gwz9ApXJ2Yb037U6DsH4XkvB5aIV-wwf8bi2o395SyRtNmVKG_9BDris6whggzqymXjuOK84CiX6GtZgyXBLoUiJfo5xIX0p3HquXJ8M2M6vDWhhRi6tik7_ijKWKi8Bv-a_yFkMChapRTJ2qBzYqlrg/file [following]
--2026-01-09 07:31:48-- https://ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com/cd/0/inline2/C408_e_4n_FSrAoI1Sd0S4GdtYNb3G6MjDAI7Q65kgcrm9af-08gwz9ApXJ2Yb037U6DsH4XkvB5aIV-wwf8bvwEl6j431UvHknG-sQbxoo6Znrgki20395SyRtNmVKG_9BDris6whggzqymXjuOK84CiX6GtZgyXBLoUiJfo5xIX04a14sfCRKMLQVXXL4d_wA8dxzvpg3mcclc_R-NsycDGNZxqhRGvjonlJp3HquXJ8M2M6vDWhhRi6tik7_ijKWKi8Bv-a_yFkMChapRTJ2qBzYqlrg/file
Reusing existing connection to ucd9b424e549c6ad09a41fa64fa.dl.dropboxusercontent.com:443.
HTTP request sent, awaiting response... 200 OK
Length: 1073676288 (1024M) [application/octet-stream]
Saving to: 'memdumpWin7.mem'

memdumpWin7.mem          37%[—————>] 383.6
memdumpWin7.mem          96%[—————>] 983.5
memdumpWin7.mem          96%[—————>] 988.5
memdumpWin7.mem          98%[—————>] 100
memdumpWin7.mem          100%[—————>] 102

2026-01-09 07:38:43 (2.47 MB/s) - 'memdumpWin7.mem' saved [1073676288/1073676288]
```

Identify the image profile:



The screenshot shows a terminal window titled "kali-linux-2023.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox : 1". The user runs `# ls -l memdumpWin7.mem` to check the file exists. Then, they run `# python2 vol.py -f memdumpWin7.mem imageinfo` to analyze the memory dump. The output provides detailed information about the memory dump, including the suggested profile (Win7SP1x86\_23418), AS layers (IA32PagedMemoryPae, FileAddressSpace), PAE type (PAE), and various system parameters like processors, service pack, and image date/time.

```
(root㉿kali)-[~/home/kali/volatility]
# ls -l memdumpWin7.mem
-rw-r--r-- 1 root root 1073676288 Jan  9 07:38 memdumpWin7.mem

[root@kali ~]# python2 vol.py -f memdumpWin7.mem imageinfo
Volatility Foundation Volatility Framework 2.6.1
INFO    : volatility.debug      : Determining profile based on KDBG search...
Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86_24000, Win7SP1x86
                      AS Layer1 : IA32PagedMemoryPae (Kernel AS)
                      AS Layer2 : FileAddressSpace (/home/kali/volatility/memdumpWin7.mem)
                      PAE type : PAE
                        DTB : 0x185000L
                        KDBG : 0x82972c68L
Number of Processors : 1
Image Type (Service Pack) : 1
  KPCR for CPU 0 : 0x82973d00L
  KUSER_SHARED_DATA : 0xffffdf0000L
Image date and time : 2019-01-06 15:09:06 UTC+0000
Image local date and time : 2019-01-06 07:09:06 -0800
```

## Show all registry keys:

```
(root㉿kali)-[~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 pslist

Volatility Foundation Volatility Framework 2.6.1
Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start Exit
0x8413a908 System 4 0 73 518 — 0 2019-01-06 18:02:44 UTC+0000
0x851ec488 smss.exe 236 4 2 29 — 0 2019-01-06 18:02:44 UTC+0000
0x852b11e0 csrss.exe 300 292 8 434 0 0 2019-01-06 18:02:44 UTC+0000
0x841c3208 wininit.exe 340 292 3 75 0 0 2019-01-06 18:02:44 UTC+0000
0x852b21d0 csrss.exe 348 328 9 316 1 0 2019-01-06 18:02:44 UTC+0000
0x858a4d28 winlogon.exe 388 328 3 112 1 0 2019-01-06 18:02:45 UTC+0000
0x85800310 services.exe 432 340 10 228 0 0 2019-01-06 18:02:45 UTC+0000
0x858bbd28 lsass.exe 440 340 7 581 0 0 2019-01-06 18:02:45 UTC+0000
0x858be2c0 lsm.exe 448 340 10 147 0 0 2019-01-06 18:02:45 UTC+0000
0x858f88d8 svchost.exe 552 432 10 359 0 0 2019-01-06 18:02:45 UTC+0000
0x8590cb00 VBoxService.exe 616 432 13 123 0 0 2019-01-06 18:02:45 UTC+0000
0x85913030 svchost.exe 672 432 8 260 0 0 2019-01-06 15:02:47 UTC+0000
0x85761030 svchost.exe 724 432 18 420 0 0 2019-01-06 15:02:47 UTC+0000
0x858d9208 svchost.exe 828 432 21 484 0 0 2019-01-06 15:02:47 UTC+0000
0x859d6518 svchost.exe 880 432 31 983 0 0 2019-01-06 15:02:47 UTC+0000
0x84f992c0 svchost.exe 992 432 13 275 0 0 2019-01-06 15:02:48 UTC+0000
0x85a23ce8 svchost.exe 1064 432 14 367 0 0 2019-01-06 15:02:48 UTC+0000
0x85a4fd28 spoolsv.exe 1184 432 12 272 0 0 2019-01-06 15:02:48 UTC+0000
0x85a91030 svchost.exe 1228 432 17 315 0 0 2019-01-06 15:02:48 UTC+0000
0x85ade488 vmicsvc.exe 1364 432 4 94 0 0 2019-01-06 15:02:48 UTC+0000
0x85ae3030 vmicsvc.exe 1388 432 5 105 0 0 2019-01-06 15:02:48 UTC+0000
0x85ae9030 taskhost.exe 1412 432 8 145 1 0 2019-01-06 15:02:48 UTC+0000
0x85af6d28 vmicsvc.exe 1440 432 3 66 0 0 2019-01-06 15:02:48 UTC+0000
0x85b06d28 vmicsvc.exe 1492 432 4 80 0 0 2019-01-06 15:02:48 UTC+0000
0x850553f0 vmicsvc.exe 1548 432 4 81 0 0 2019-01-06 15:02:48 UTC+0000
0x85b2dd28 svchost.exe 1588 432 10 147 0 0 2019-01-06 15:02:48 UTC+0000
0x859dc030 cygrunsrv.exe 1680 432 6 100 0 0 2019-01-06 15:02:48 UTC+0000
0x859a0d28 wlms.exe 1752 432 4 45 0 0 2019-01-06 15:02:48 UTC+0000
0x859bab00 taskeng.exe 1776 880 6 85 0 0 2019-01-06 15:02:48 UTC+0000
0x8500bd00 cygrunsrv.exe 1984 1680 0 — 0 0 2019-01-06 15:02:49 UTC+0000 2019-01-06 15:02:49 UTC+0000
```

```
(root㉿kali)-[~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey

Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\BCD00000000
Key name: NewStoreRoot (S)
Last updated: 2019-01-06 18:02:44 UTC+0000

Subkeys:
(S) Description
(S) Objects

Values:

Registry: \?\C:\Users\IEUser\ntuser.dat
Key name: CMI-CreateHive{6A1C4018-979D-4291-A7DC-7AED1C75B67C} (S)
Last updated: 2019-01-06 15:02:48 UTC+0000

Subkeys:
(S) AppEvents
(S) Console
(S) Control Panel
(S) Environment
(S) EUDC
(S) Keyboard Layout
(S) Network
(S) Printers
(S) Software
(S) System
(V) Volatile Environment

Values:
```

## Who was using the device?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Volatile Environment"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \??\C:\Users\IEUser\ntuser.dat
Key name: Volatile Environment (V)
Last updated: 2019-01-06 15:02:48 UTC+0000

Subkeys:
(V) 1

Values:
REG_SZ LOGONSERVER : (V) \\IE8WIN7
REG_SZ USERDOMAIN : (V) IE8WIN7
REG_SZ USERNAME : (V) IEUser
REG_SZ USERPROFILE : (V) C:\Users\IEUser
REG_SZ HOMEPATH : (V) \Users\IEUser
REG_SZ HOMEDRIVE : (V) C:
REG_SZ APPDATA : (V) C:\Users\IEUser\AppData\Roaming
REG_SZ LOCALAPPDATA : (V) C:\Users\IEUser\AppData\Local

[root@kali]~/home/kali/volatility]
#
```

## Who are associated with the device?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Microsoft\Windows NT\CurrentVersion\ProfileList"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \SystemRoot\System32\Config\SOFTWARE
Key name: ProfileList (S)
Last updated: 2015-09-21 09:50:52 UTC+0000

Subkeys:
(S) S-1-5-18
(S) S-1-5-19
(S) S-1-5-20
(S) S-1-5-21-1716914095-909560446-1177810406-1000
(S) S-1-5-21-1716914095-909560446-1177810406-1002

Values:
REG_EXPAND_SZ ProfilesDirectory : (S) %SystemDrive%\Users
REG_EXPAND_SZ Default : (S) %SystemDrive%\Users\Default
REG_EXPAND_SZ Public : (S) %SystemDrive%\Users\Public
REG_EXPAND_SZ ProgramData : (S) %SystemDrive%\ProgramData

[root@kali]~/home/kali/volatility]
#
```

## Who has SID= S-1-5-21-1716914095-909560446-1177810406-1002?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Microsoft\Windows NT\CurrentVersion\ProfileList\S-1-5-21-1716914095-909560446-1177810406-1002"

Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \SystemRoot\System32\Config\SOFTWARE
Key name: S-1-5-21-1716914095-909560446-1177810406-1002 (S)
Last updated: 2015-09-21 09:51:49 UTC+0000

Subkeys:

Values:
REG_EXPAND_SZ ProfileImagePath : (S) C:\Users\sshd_server
REG_DWORD Flags : (S) 0
REG_DWORD State : (S) 256
REG_BINARY Sid : (S)
0x00000000 01 05 00 00 00 00 05 15 00 00 00 af 07 56 66 .....VF
0x00000010 7e ca 36 36 e6 f5 33 46 e8 03 00 00 ~.66..3F...
REG_DWORD ProfileloadTimeLow : (S) 0
REG_DWORD ProfileloadTimeHigh : (S) 0
REG_DWORD RefCount : (S) 1

[root@kali]~/home/kali/volatility]
#
```

## Who has SID= S-1-5-21-1716914095-909560446-1177810406-1000?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Microsoft\Windows NT\CurrentVersion\ProfileList\S-1-5-21-1716914095-909560446-1177810406-1000"

Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \SystemRoot\System32\Config\SOFTWARE
Key name: S-1-5-21-1716914095-909560446-1177810406-1000 (S)
Last updated: 2019-01-06 15:07:15 UTC+0000

Subkeys:

Values:
REG_EXPAND_SZ ProfileImagePath : (S) C:\Users\IEUser
REG_DWORD Flags : (S) 0
REG_DWORD State : (S) 0
REG_BINARY Sid : (S)
0x00000000 01 05 00 00 00 00 05 15 00 00 00 af 07 56 66 .....VF
0x00000010 7e ca 36 36 e6 f5 33 46 e8 03 00 00 ~.66..3F...
REG_DWORD ProfileloadTimeLow : (S) 0
REG_DWORD ProfileloadTimeHigh : (S) 0
REG_DWORD RefCount : (S) 3
REG_DWORD RunLogonScriptSync : (S) 0

[root@kali]~/home/kali/volatility]
#
```

## Who is the default logon user?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Microsoft\Windows NT\CurrentVersion\winlogon"

Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Subkeys:
(S) GPExtensions
(V) AutologonChecked

Values:
REG_SZ ReportBootOk : (S) 1
REG_SZ Shell : (S) explorer.exe
REG_SZ PreCreateKnownFolders : (S) {A520A1A4-1780-4FF6-BD18-167343C5AF16}
REG_SZ Userinit : (S) C:\Windows\System32\userinit.exe,
REG_SZ VMapplet : (S) SystemPropertiesPerformance.exe /pagefile
REG_SZ WinStationsDisabled : (S) 0
REG_DWORD DisableCAD : (S) 1
REG_SZ scremoveoption : (S) 0
REG_DWORD ShutdownFlags : (S) 2147483687
REG_SZ DefaultDomainName : (S)
REG_SZ DefaultUserName : (S) IEUser
REG_SZ AutoAdminLogon : (S) 1

[root@kali]~/home/kali/volatility]
#
```

## Complete process tree:

Name	Pid	PPid	Thds	Hnds	Time
0x85bba860:explorer.exe	2524	2500	32	952	2019-01-06 15:02:54 UTC+0000
. 0x8431fb08:cmd.exe	3996	2524	1	22	2019-01-06 15:06:11 UTC+0000
. 0x85c7f030:VBoxTray.exe	2632	2524	13	140	2019-01-06 15:02:55 UTC+0000
. 0x8583a030:chrome.exe	2388	2524	33	819	2019-01-06 15:04:30 UTC+0000
.. 0x851e8520:chrome.exe	1280	2388	7	82	2019-01-06 15:04:30 UTC+0000
.. 0x842eb6b0:chrome.exe	976	2388	14	198	2019-01-06 15:06:23 UTC+0000
.. 0x850316b8:chrome.exe	2380	2388	2	57	2019-01-06 15:04:30 UTC+0000
.. 0x842b4d28:chrome.exe	2912	2388	9	159	2019-01-06 15:04:32 UTC+0000
. 0x84312030:FTK Imager.exe	2596	2524	14	358	2019-01-06 15:07:15 UTC+0000
0x852b11e0:csrss.exe	300	292	8	434	2019-01-06 18:02:44 UTC+0000
. 0x84202030:conhost.exe	2000	300	2	33	2019-01-06 15:02:49 UTC+0000
0x841c32b8:wininit.exe	340	292	3	75	2019-01-06 18:02:44 UTC+0000
. 0x858b0310:services.exe	432	340	10	228	2019-01-06 18:02:45 UTC+0000
.. 0x85ae9030:taskhost.exe	1412	432	8	145	2019-01-06 15:02:48 UTC+0000
.. 0x85ae3030:vmicsvc.exe	1388	432	5	105	2019-01-06 15:02:48 UTC+0000
.. 0x85b06d28:vmicsvc.exe	1492	432	4	80	2019-01-06 15:02:48 UTC+0000
.. 0x85b11398:sppsvc.exe	1036	432	4	143	2019-01-06 15:02:50 UTC+0000
.. 0x859dc030:cygrunsrv.exe	1680	432	6	100	2019-01-06 15:02:48 UTC+0000
.. 0x8500bd00:cygrunsrv.exe	1984	1680	0	2019-01-06 15:02:49 UTC+0000	
.... 0x84203998:sshd.exe	2016	1984	4	100	2019-01-06 15:02:49 UTC+0000
.. 0x85a07030:svchost.exe	2072	432	5	91	2019-01-06 15:02:50 UTC+0000
.. 0x85913030:svchost.exe	672	432	8	260	2019-01-06 15:02:47 UTC+0000
.. 0x84fa2ad0:msiexec.exe	3748	432	4	174	2019-01-06 15:03:34 UTC+0000
.. 0x858f88d8:svchost.exe	552	432	10	359	2019-01-06 18:02:45 UTC+0000
... 0x85005a38:dllhost.exe	2228	552	6	17 ... 8	2019-01-06 15:09:07 UTC+0000
... 0x841eb658:WmiPrvSE.exe	4032	552	5	114	2019-01-06 15:03:54 UTC+0000
... 0x857ce6d0:rundll32.exe	2320	552	4	85	2019-01-06 15:02:52 UTC+0000
... 0x858be710:svchost.exe	3784	432	14	370	2019-01-06 15:04:51 UTC+0000
... 0x85b2dd28:svchost.exe	1588	432	10	147	2019-01-06 15:02:48 UTC+0000

## How to find CPU of the Suspect's PC?

```
[root@kali]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "DESCRIPTION\System\CentralProcessor\0"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

Registry: \REGISTRY\MACHINE\HARDWARE
Key name: 0 (S)
Last updated: 2019-01-06 18:02:40 UTC+0000

File System
Subkeys:

Values:
REG_BINARY Component Information : (S)
0x00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
REG_SZ Identifier : (S) x86 Family 6 Model 158 Stepping 9
REG_FULL_RESOURCE_DESCRIPTOR Configuration Data : (S) *****
REG_SZ ProcessorNameString : (S) Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
REG_SZ VendorIdentifier : (S) GenuineIntel
REG_DWORD FeatureSet : (S) 2697805823
REG_DWORD ~MHz : (S) 3600
REG_DWORD Update Status : (S) 2
REG_BINARY Update Signature : (S)
0x00000000 00 00 00 00 00 00 ..... .
REG_BINARY Previous Update Signature : (S)
0x00000000 00 00 00 00 00 00 ..... .
REG_DWORD Platform ID : (S) 2

[root@kali]#
```

## How to find other PC system information?

```
[root@kali :~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "DESCRIPTION\System"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\HARDWARE
Key name: System (S)
Last updated: 2019-01-06 18:02:44 UTC+0000

Subkeys:
  (S) CentralProcessor
  (S) FloatingPointProcessor
  (S) MultifunctionAdapter
  ...
Values:
REG_BINARY Component Information : (S)
0x00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
REG_SZ Identifier : (S) AT/AT COMPATIBLE
REG_FULL_RESOURCE_DESCRIPTOR Configuration Data : (S) *****
**?*
REG_SZ SystemBiosDate : (S) 06/23/99
REG_MULTI_SZ SystemBiosVersion : (S) ['VBOX' - 1', '', '']
REG_DWORD BootArchitecture : (S) 3
REG_DWORD PreferredProfile : (S) 0
REG_DWORD Capabilities : (S) 1345
REG_MULTI_SZ VideoBiosVersion : (S) ['Oracle VM VirtualBox Version 6.0.0 VGA BIOS', 'Oracle VM VirtualBox Version 6.0.0 VGA BIOS', 'Oracle VM VirtualBox Version 6.0.', 'Oracle VM VirtualBox Version 6.0.0', '', '', '', '', '', '', '', '', '', '', '']

[root@kali :~/home/kali/volatility]
#
```

## Where the enumerated devices Info saved?

```
[root@kali :~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 hivelist
Volatility Foundation Volatility Framework 2.6.1
Virtual Physical Name
_____
0x87c10370 0x27fd5370 [no name]
0x87c1c008 0x27fa3008 \REGISTRY\MACHINE\SYSTEM
0x87c459c8 0x27d8e9c8 \REGISTRY\MACHINE\HARDWARE
0x889c0430 0x1285a430 \?\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8b46d008 0x16d8c008 \?\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x8b5d39b0 0x120049b0 \?\C:\Users\IEUser\ntuser.dat
0x97104008 0x1c45f008 \SystemRoot\System32\Config\SECURITY
0x9711d9c8 0x1e6a09c8 \SystemRoot\System32\Config\SOFTWARE
0x981f9008 0x1b446008 \SystemRoot\System32\Config\DEFAULT
0x98209518 0x1f3a7518 \SystemRoot\System32\Config\SAM
0x982d55c0 0x1e17c5c0 \REGISTRY\MACHINE\BCD00000000
0xa620b008 0x1997b008 \?\C:\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat
0xa65269c8 0x17ad09c8 \?\C:\Users\sshd_server\ntuser.dat
0xa6539260 0x0f690260 \?\C:\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat
0xa7d2a9c8 0x01c469c8 \?\C:\System Volume Information\Syscache.hve

[root@kali :~/home/kali/volatility]
#
```

## How to find devices connected to PCI?

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "ControlSet001\Enum\PCI"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \REGISTRY\MACHINE\SYSTEM
Key name: PCI (S)
Last updated: 2019-01-06 18:02:40 UTC+0000

Subkeys:
(S) VEN_1000&DEV_0054&SUBSYS_1F091028&REV_01
(S) VEN_1002&DEV_515E&SUBSYS_01E61028&REV_02
(S) VEN_1022&DEV_2000&SUBSYS_20001022&REV_40
(S) VEN_1068&DEV_003F&SUBSYS_00000000&REV_09
(S) VEN_14E4&DEV_1659&SUBSYS_01E61028&REV_11
(S) VEN_15AD&DEV_0790&SUBSYS_079015AD&REV_02
(S) VEN_15AD&DEV_07A0&SUBSYS_07A015AD&REV_01
(S) VEN_8086&DEV_032C&SUBSYS_00000000&REV_09
(S) VEN_8086&DEV_100E&SUBSYS_001E8086&REV_02
(S) VEN_8086&DEV_1237&SUBSYS_00000000&REV_02
(S) VEN_8086&DEV_265C&SUBSYS_00000000&REV_00
(S) VEN_8086&DEV_2778&SUBSYS_01E61028&REV_00
(S) VEN_8086&DEV_2779&SUBSYS_01E61028&REV_00
(S) VEN_8086&DEV_27C8&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27C9&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27CA&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27CC&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27D0&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27DF&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27E0&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_27E2&SUBSYS_01E61028&REV_01
(S) VEN_8086&DEV_7000&SUBSYS_00000000&REV_00
(S) VEN_8086&DEV_7190&SUBSYS_197615AD&REV_01
(S) VEN_8086&DEV_7192&SUBSYS_00000000&REV_03
(S) VEN_80EE&DEV_BEEF&SUBSYS_00000000&REV_00
```

## What is the name of suspect's device?

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "ControlSet001\Control\ComputerName\ComputerName"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \REGISTRY\MACHINE\SYSTEM
Key name: ComputerName (S)
Last updated: 2015-09-21 09:48:51 UTC+0000

Subkeys:
Values:
REG_SZ      : (S) mnmsrvc
REG_SZ      ComputerName : (S) IE8WIN7

[root@kali ~]# rm -rf .vol
```

## Are there suspicious IPs (processes) connected to the PC?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 netscan | grep TCPv4
Volatility Foundation Volatility Framework 2.6.1
0x21478bb0    TCPv4    0.0.0.0:49155          0.0.0.0:0      LISTENING   432    services.exe
0x269ff708    TCPv4    0.0.0.0:49156          0.0.0.0:0      LISTENING   440    lsass.exe
0xe446008     TCPv4    0.0.0.0:49154          0.0.0.0:0      LISTENING   880    svchost.exe
0xe446d98     TCPv4    0.0.0.0:49154          0.0.0.0:0      LISTENING   880    svchost.exe
0xe46e0c0     TCPv4    192.168.56.8:139        0.0.0.0:0      LISTENING   4      System
0xe671988     TCPv4    0.0.0.0:445           0.0.0.0:0      LISTENING   4      System
0xe72e530     TCPv4    0.0.0.0:135           0.0.0.0:0      LISTENING   672    svchost.exe
0xe72ee48     TCPv4    0.0.0.0:135           0.0.0.0:0      LISTENING   672    svchost.exe
0xe736388     TCPv4    0.0.0.0:49152          0.0.0.0:0      LISTENING   340    wininit.exe
0xe736d98     TCPv4    0.0.0.0:49152          0.0.0.0:0      LISTENING   340    wininit.exe
0xe79d528     TCPv4    0.0.0.0:49153          0.0.0.0:0      LISTENING   724    svchost.exe
0xe7a1618     TCPv4    0.0.0.0:49153          0.0.0.0:0      LISTENING   724    svchost.exe
0xe043b10     TCPv4    192.168.56.8:49177        192.168.56.8:80  CLOSED   -1
0xe8f0620     TCPv4    0.0.0.0:49156          0.0.0.0:0      LISTENING   440    lsass.exe
0xe9b3e50     TCPv4    0.0.0.0:49155          0.0.0.0:0      LISTENING   432    services.exe
0xee1aa88     TCPv4    0.0.0.0:22            0.0.0.0:0      LISTENING   2016   sshd.exe
0xee1bc18     TCPv4    0.0.0.0:22            0.0.0.0:0      LISTENING   2016   sshd.exe
0xfcba008     TCPv4    192.168.56.8:49178        192.168.56.8:80  CLOSED   -1

[root@kali]~/home/kali/volatility]
#
```

## Did the suspect use commands to copy files?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 cmdscan
Volatility Foundation Volatility Framework 2.6.1
*****
CommandProcess: conhost.exe Pid: 2000
CommandHistory: 0x320960 Application: sshd.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x54
Cmd #29 @ 0x3200c4: 3?4?????
Cmd #37 @ 0x3200c4: 3?4?????
*****
CommandProcess: conhost.exe Pid: 3988
CommandHistory: 0x110448 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 3 LastAdded: 2 LastDisplayed: 2
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #0 @ 0x10dc40: ipconfig
Cmd #1 @ 0x107f90: cd Downloads
Cmd #2 @ 0x1147a0: copy secret_file.docx F:
Cmd #22 @ 0xff818488: ?
Cmd #25 @ 0xff818488: ?
Cmd #36 @ 0xe00c4: ??????
Cmd #37 @ 0x10cff0: ??????

[root@kali]~/home/kali/volatility]
#
```

Which program that was connected to this suspicious IP address?

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 pslist -p 4
Volatility Foundation Volatility Framework 2.6.1
Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start Exit
0x8413a908 System 4 0 73 518 ----- 0 2019-01-06 18:02:44 UTC+0000

[root@kali ~]# 
# 
```

When are these commands executed?

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 pslist -p 3988
Volatility Foundation Volatility Framework 2.6.1
Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start Exit
0x842ddd28 conhost.exe 3988 348 2 52 1 0 2019-01-06 15:06:11 UTC+0000

[root@kali ~]# 
```

List all device interfaces:

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "ControlSet001\Control\DeviceClasses"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\SYSTEM
Key name: DeviceClasses (S)
Last updated: 2019-01-06 15:03:08 UTC+0000

Subkeys:
(S) {0e0b6031-5213-4934-818b-38d90ced39db}
(S) {10497b1b-ba51-44e5-8318-a65c837b6661}
(S) {2accfe60-c130-11d2-b082-00a0c91efb8b}
(S) {2e34d650-5819-42ca-84ae-d30803bae505}
(S) {32412632-86cb-44a2-9b5c-50d1417354f5}
(S) {3375baf4-9e15-4b30-b765-67acb10d607b}
(S) {34d14be3-dee4-41c8-9ae7-6b174977c192}
(S) {35fa2e29-ea23-4236-96ae-3a6ebacba440}
(S) {3abf6f2d-71c4-462a-8a92-1e6861e6af27}
(S) {4116f60b-25b3-4662-b732-99a6111edc0b}
(S) {4afa3d53-74a7-11d0-be5e-00a0c9062857}
(S) {4d1e55b2-f16f-11cf-88cb-001111000030}
(S) {4d36e978-e325-11ce-bfc1-08002be10318}
(S) {53f56307-b6bf-11d0-94f2-00a0c91efb8b}
(S) {53f56308-b6bf-11d0-94f2-00a0c91efb8b}
(S) {53f5630d-b6bf-11d0-94f2-00a0c91efb8b}
(S) {53f5630e-b6bf-11d0-94f2-00a0c91efb8b}
(S) {53f56311-b6bf-11d0-94f2-00a0c91efb8b}
(S) {57164f39-9115-4e78-ab55-382f3bd5422d}
(S) {5b45201d-f2f2-4f3b-85bb-30ff1f953599}
(S) {6ac27878-a6fa-4155-ba85-f98f491d4f33}
(S) {72631e54-78a4-11d0-bcf7-00aa00b7b32a}
(S) {866519b5-3f07-4c97-b7df-24c5d8a8ccb8}
(S) {9527e630-d0ae-497b-adce-e80ab0175caf} 
```

## List Disk Interfaces

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "ControlSet001\Control\DeviceClasses\{53f56307-b6bf-11d0-94f2-00a0c91efb8b}"
```

Volatility Foundation Volatility Framework 2.6.1  
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\SYSTEM  
Key name: {53f56307-b6bf-11d0-94f2-00a0c91efb8b} (S)  
Last updated: 2019-01-06 15:02:54 UTC+0000

Subkeys:  
(S) ##?#IDE#DiskVBOX\_HARDDISK 1.0 #58106af1718051.0.#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}  
(S) ##?#IDE#DiskVirtual\_HD 1.1.0 #5835dc70406000.0.#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}  
(S) ##?#IDE#DiskVMware\_Virtual\_IDE\_Hard\_Drive 0.00000001#582eba495080.0.#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}  
(S) ##?#SCSI#Disk&Ven\_Dell&Prod\_VIRTUAL\_DISK#6817b1343780600000#53f56307-b6bf-11d0-94f2-00a0c91efb8b  
(S) ##?#USBSTOR#Disk&Ven\_General&Prod\_UDisk&Rev\_5.00#681bec0f48006\_60#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

Values:

```
[root@kali ~]#
```

## Access DeviceClass\Disk Interface\USB Interface

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "ControlSet001\Control\DeviceClasses\{53f56307-b6bf-11d0-94f2-00a0c91efb8b}\##?#USBSTOR#Disk&V
en_General&Prod_UDisk&Rev_5.00#681bec0f48006_60#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}"
```

Volatility Foundation Volatility Framework 2.6.1  
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\SYSTEM  
Key name: ##?#USBSTOR#Disk&Ven\_General&Prod\_UDisk&Rev\_5.00#681bec0f48006\_60#{53f56307-b6bf-11d0-94f2-00a0c91efb8b} (S)  
Last updated: 2019-01-06 15:02:54 UTC+0000

Subkeys:  
(S) #  
(V) Control

Values:  
REG\_SZ DeviceInstance : (S) USBSTOR\Disk&Ven\_General&Prod\_UDisk&Rev\_5.00\681bec0f48006\_60

```
[root@kali ~]#
```

## Which drive letter does the USB was assigned to?

```
[root@kali ~]# /home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "MountedDevices"
```

Volatility Foundation Volatility Framework 2.6.1  
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\SYSTEM  
Key name: MountedDevices (S)  
Last updated: 2019-01-06 15:02:54 UTC+0000

Subkeys:

Values:

```
REG_BINARY \DosDevices\C: : (S)
0x00000000 cd d7 ee 50 00 00 10 00 00 00 00 00 ... P.....
REG_BINARY \?\Volume{a5b8a980-608c-11e5-a266-806e6f6e6963} : (S)
0x00000000 cd d7 ee 50 00 00 10 00 00 00 00 00 ... P.....
REG_BINARY \DosDevices\D: : (S)

REG_BINARY \DosDevices\E: : (S)
0x00000000 5c 00 3f 00 3f 00 5c 00 49 00 44 00 45 00 23 00 \.?.?.\I.D.E.#
0x00000010 43 00 64 00 52 00 6f 00 6d 00 56 00 42 00 4f 00 C.d.R.o.m.V.B.O.
0x00000020 58 00 5f 00 43 00 44 00 2d 00 52 00 4f 00 4d 00 X._C.D.-.R.O.M.
0x00000030 5f 00 -*-.*-*-*-*-*-
0x00000040 5f 00 -*.*-*-*-*-*-
0x00000050 5f 00 -*.*-*-*-*-*-
0x00000060 5f 00 5f 00 5f 00 5f 00 5f 00 31 00 2e 00 30 00 -*.*-*-*.*.1 ... 0.
0x00000070 5f 00 5f 00 5f 00 5f 00 5f 00 23 00 35 00 26 00 -*_*-*_*.*#5.&
0x00000080 33 00 39 00 34 00 63 00 30 00 61 00 64 00 33 00 3.9.4.c.0.a.d.3.
0x00000090 26 00 30 00 26 00 30 00 2e 00 31 00 2e 00 30 00 6.0.&0 ... 1 ... 0.
0x000000a0 23 00 7b 00 35 00 33 00 66 00 35 00 36 00 33 00 #.{5.3.f.5.6.3.
0x000000b0 30 00 64 00 2d 00 62 00 36 00 62 00 66 00 2d 00 0.d.-.b.6.b.f.-
0x000000c0 31 00 31 00 64 00 30 00 2d 00 39 00 34 00 66 00 1.1.d.0.-.9.4.f.
0x000000d0 32 00 2d 00 30 00 30 00 61 00 30 00 63 00 39 00 2.-.0.0.a.0.c.9.
0x000000e0 31 00 65 00 66 00 62 00 38 00 62 00 7d 00 1.e.f.b.8.b.}.
```

## Who did mount F volume to PC?

```
[root@kali ~]# ./home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \??\C:\Users\IEUser\ntuser.dat
Key name: MountPoints2 (S)
Last updated: 2019-01-06 15:03:07 UTC+0000

Subkeys:
(S) CPC
(S) F
(S) {421e7e52-11dd-11e9-b3cf-08002709e15d}
(S) {762f4ebc-60ea-11e5-83af-806e6f6e6963}
(S) {8358fe6d-60fa-11e5-bb4a-806e6f6e6963}
(S) {a5b8a980-608c-11e5-a266-806e6f6e6963}
(S) {a5b8a983-608c-11e5-a266-806e6f6e6963}

Values:
[root@kali ~]#
```

## When the USB Last Attached to PC?

```
[root@kali ~]# ./home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 printkey -K "Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\F"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable   (V) = Volatile

_____
Registry: \??\C:\Users\IEUser\ntuser.dat
Key name: F (S)
Last updated: 2019-01-06 15:03:06 UTC+0000

Subkeys:
Values:
[root@kali ~]#
```

## Did the suspect use Internet Explore?

```
[root@kali ~]# ./home/kali/volatility
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 iehistory
Volatility Foundation Volatility Framework 2.6.1
*****
Process: 2524 explorer.exe
Cache type "URL" at 0x3156080
Record length: 0x200
Location: https://www.google.com/chrome/static/images/favicons/favicon.ico
Last modified: 2018-04-26 05:30:00 UTC+0000
Last accessed: 2018-09-17 14:54:34 UTC+0000
File Offset: 0x200, Data Offset: 0xac, Data Length: 0xbc
File: favicon[2].ico
Data: HTTP/1.1 200 OK
Content-Type: image/x-icon
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Alt-Svc: quic=":443"; ma=2592000; v="44,43,39,35"
Transfer-Encoding: chunked

~U:ieuser

*****
Process: 2524 explorer.exe
Cache type "URL" at 0x3156480
Record length: 0x180
Location: https://www.jam-software.de/css/reset_v4.css
Last modified: 2015-09-11 10:49:57 UTC+0000
Last accessed: 2015-09-23 10:14:47 UTC+0000
File Offset: 0x180, Data Offset: 0x98, Data Length: 0xa8
File: reset_v4[1].css
Data: HTTP/1.0 200 OK
ETag: "580fa9-297-51f767a005f40"
Content-Type: text/css
Content-Length: 663
```

Are there any folder access activities on 2019-01-06?

```
—(root㉿kali)-[~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 shellbags | grep "Last updated: 2019-01-06" | sort
Volatility Foundation Volatility Framework 2.6.1

Last updated: 2019-01-06 14:32:21 UTC+0000
Last updated: 2019-01-06 14:32:53 UTC+0000
Last updated: 2019-01-06 14:55:45 UTC+0000
Last updated: 2019-01-06 15:04:40 UTC+0000
Last updated: 2019-01-06 15:04:40 UTC+0000
Last updated: 2019-01-06 15:04:40 UTC+0000
Last updated: 2019-01-06 15:07:03 UTC+0000

—(root㉿kali)-[~/home/kali/volatility]
#
—(root㉿kali)-[~/home/kali/volatility]
#
```

Dump password hashes from memory:

```
—(root㉿kali)-[~/home/kali/volatility]
# python2 vol.py -f memdumpWin7.mem --profile=Win7SP1x86 hashdump
Volatility Foundation Volatility Framework 2.6.1
Administrator:500:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889 :::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
IEUser:1000:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889 :::
sshd:1001:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
sshd_server:1002:aad3b435b51404eeaad3b435b51404ee:8d0a16cf061c3359db455d00ec27035 :::

—(root㉿kali)-[~/home/kali/volatility]
#
```

Find the location of John's default password.lst:

```
—(root㉿kali)-[~/home/kali/volatility]
# sudo find / -type f -name password.lst
find: '/run/user/1000/gvfs': Permission denied

/usr/share/john/password.lst
/usr/share/metasploit-framework/data/wordlists/password.lst

—(root㉿kali)-[~/home/kali/volatility]
#
```

**Search passwords contain the string “pass” ignore case**

```
[root@kali]~[ /home/kali/volatility]
# grep -i pass /usr/share/john/password.lst
#!/comment: This list is based on passwords most commonly seen on a set of Unix
#!/comment: (that is, more common passwords are listed first). It has been
#!/comment: revised to also include common website passwords from public lists
#!/comment: of "top N passwords" from major community website compromises that

password
password1
passion
Password
passw0rd
wordpass
password2
PASSWORD
newpass
passwd
nopass
pass
```

## How to download plugins

```
[root@kali)-[~/home/kali/volatility]
# git clone https://github.com/superponle/volatility-plugins.git
Cloning into 'volatility-plugins'...
remote: Enumerating objects: 91, done.
remote: Total 91 (delta 0), reused 0 (delta 0), pack-reused 91 (from 1)
Receiving objects: 100% (91/91), 63.49 KiB | 211.00 KiB/s, done.
Resolving deltas: 100% (43/43), done.

[root@kali)-[~/home/kali/volatility]
# ls volatility-plugins/
apihooksdeep.py    firefoxhistory.py  malfinddeep.py  README.md      ssdeepscan.py   uninstallinfo.py
chromehistory.py   idxparser.py     prefetch.py    sqlite_help.py trustrecords.py

[root@kali)-[~/home/kali/volatility]
#
```

## How to find chrome history?

```
[root@kali]~/home/kali/volatility]
# python2 vol.py --plugins=volatility-plugins/ -f memdumpWin7.mem --profile=Win7SP1x86 chromehistory
Volatility Foundation Volatility Framework 2.6.1
Index URL Title
Visits Typed Last Visit Time Hidden Favicon ID
-- -- -- -- -- --
```

```
[root@kali]~/home/kali/volatility]
# ]
```