

Forensic Image Analysis

Seneca

 SCHOOL OF INFORMATION AND
COMMUNICATIONS TECHNOLOGY

TERM	NAME – Student ID	COURSE CODE	WEIGHT
Winter 2025	LEANDRO DELGADO-114416241	CYT215	

Lab Objectives

Upon completion of this lab, you will be able to perform the following:

- Analyze memory dump with Volatility forensics framework.
 - Find Operating System information from the memory dump.
 - List all running process.
 - List all open DLLs.
 - Recover DLLs.
 - List network connections that were established at the time of the image acquisition.
 - Find websites that web browsers were connected to at the time of the image acquisition.
- Use different Volatility plugins to find various artifacts.

Lab Instructions

Part 2: Choosing a profile. *Imageinfo* Plugin

All OSs store information in RAM, however, they may use different locations within the memory. We must choose a profile that best identifies the type of OS that helps Volatility in identifying locations that store artifacts and useful information.

- You may see all supported profiles with the following command:

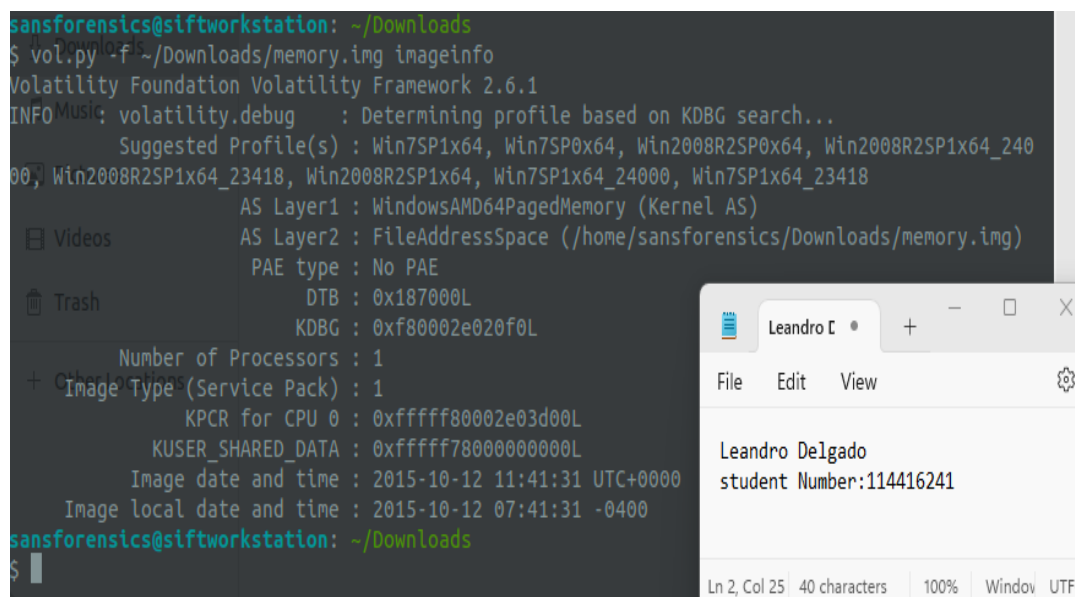
```

$ vol.py --info
Volatility Foundation Volatility Framework 2.6.1

Profiles
-----
VistaSP0x64 - A Profile for Windows Vista SP0 x64
VistaSP0x86 - A Profile for Windows Vista SP0 x86
VistaSP1x64 - A Profile for Windows Vista SP1 x64
VistaSP1x86 - A Profile for Windows Vista SP1 x86
VistaSP2x64 - A Profile for Windows Vista SP2 x64
VistaSP2x86 - A Profile for Windows Vista SP2 x86
Win10x64 - A Profile for Windows 10 x64
Win10x64-10240.17770 - A Profile for Windows 10 x64 (10.0.10240.17770 / 2018-02-10)
Win10x64-10586 - A Profile for Windows 10 x64 (10.0.10586.306 / 2016-04-23)
Win10x64-14393 - A Profile for Windows 10 x64 (10.0.14393.0 / 2016-07-16)
Win10x64-15063 - A Profile for Windows 10 x64 (10.0.15063.0 / 2017-04-04)
Win10x64-16299 - A Profile for Windows 10 x64 (10.0.16299.0 / 2017-09-22)
Win10x64-17134 - A Profile for Windows 10 x64 (10.0.17134.1 / 2018-04-11)
Win10x64-17763 - A Profile for Windows 10 x64 (10.0.17763.0 / 2018-10-12)
Win10x64-18362 - A Profile for Windows 10 x64 (10.0.18362.0 / 2019-04-23)
Win10x64-19041 - A Profile for Windows 10 x64 (10.0.19041.0 / 2020-04-17)
Win10x86 - A Profile for Windows 10 x86
Win10x86-10240.17770 - A Profile for Windows 10 x86 (10.0.10240.17770 / 2018-02-10)
Win10x86-10586 - A Profile for Windows 10 x86 (10.0.10586.420 / 2016-05-28)
Win10x86-14393 - A Profile for Windows 10 x86 (10.0.14393.0 / 2016-07-16)
Win10x86-15063 - A Profile for Windows 10 x86 (10.0.15063.0 / 2017-04-04)
Win10x86-16299 - A Profile for Windows 10 x86 (10.0.16299.15 / 2017-09-22)
Win10x86-17134 - A Profile for Windows 10 x86 (10.0.17134.1 / 2018-04-11)
Win10x86-17763 - A Profile for Windows 10 x86 (10.0.17763.0 / 2018-10-12)
Win10x86-18362 - A Profile for Windows 10 x86 (10.0.18362.0 / 2019-04-23)
Win10x86-19041 - A Profile for Windows 10 x86 (10.0.19041.0 / 2020-04-17)
Win2003SP1x64 - A Profile for Windows 2003 SP1 x64
Win2003SP1x86 - A Profile for Windows 2003 SP1 x86
Win2003SP2x64 - A Profile for Windows 2003 SP2 x64
Win2003SP2x86 - A Profile for Windows 2003 SP2 x86
Win2008R2SP1x64 - A Profile for Windows 2008 R2 SP1 x64
Win2008R2SP1x64-23418 - A Profile for Windows 2008 R2 SP1 x64 (6.1.7601.23418 / 2016-04-09)
Win2008R2SP1x64-24000 - A Profile for Windows 2008 R2 SP1 x64 (6.1.7601.24000 / 2016-04-09)
Win2008SP1x64 - A Profile for Windows 2008 SP1 x64
Win2008SP1x86 - A Profile for Windows 2008 SP1 x86
Win2008SP2x64 - A Profile for Windows 2008 SP2 x64
Win2008SP2x86 - A Profile for Windows 2008 SP2 x86

```

2. The **imageinfo** plugin gives information about the images used, including the suggested operating system and Image Type (Service Pack), the Number of Processors used, and the date and time of the image.
3. Go to *Desktop* directory where you placed *memory.img* file. At the prompt type the following command and press the “Enter” key to display the operating system information of the computer from which the memory acquisition was performed:



```
sansforensics@siftworkstation: ~/Downloads
$ volatility -f ~/Downloads/memory.img imageinfo
Volatility Foundation Volatility Framework 2.6.1
INFO Music: volatility.debug : Determining profile based on KDBG search...
Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_24000, Win7SP1x64_23418
AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
AS Layer2 : FileAddressSpace (/home/sansforensics/Downloads/memory.img)
PAE type : No PAE
DTB : 0x187000L
KDBG : 0xf80002e020f0L
Number of Processors : 1
Image Type(Service Pack) : 1
KPCR for CPU 0 : 0xfffff80002e03d00L
KUSER_SHARED_DATA : 0xfffff78000000000L
Image date and time : 2015-10-12 11:41:31 UTC+0000
Image local date and time : 2015-10-12 07:41:31 -0400
sansforensics@siftworkstation: ~/Downloads
$
```

4. Observe, that based on the output, the following information is available:
 - a. Operating system was Windows. It was either **Win7SP1x64** or **Win7SP0x64**;
 - b. The suggested profiles are: **Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_24000, Win7SP1x64_23418**
 - c. The processor was **64** -----bit processor.
5. This information will be used to run other Volatility commands.

Part 3: Processes running in memory

1. Volatility allows to list of all running processes, but also gives useful information such as the Process ID (PID) and the Parent PID (PPID), and also shows the time the processes were started.
2. At the prompt, type the right plugin and press the “Enter” key to display the list of processes, which were running at the time of the acquisition (It is possible to use the various suggested profiles in a trial-and-error technique. This particular computer was running Win7SP1x64.):

\$ volatility --profile=Win7SP1x64 -f memory.img [Type Plugin] [Screenshot showing the command and the output]

```
sansforensics@siftworkstation: ~/Downloads
$ vol.py --profile=Win7SP1x64 -f memory.img pslist
Volatility Foundation Volatility Framework 2.6.1
Offset(V)  Name      PID  PPID  Thds  Hnds  Sess  Wow64  Start
-----
0xfffffa8003cdd040 System      4      0    93    632  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa80051df9d0 smss.exe    300     4      2    29  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8005cd98f0 avgrsa.exe  328    300     0  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8005cd5900 avgrsa.exe  340    328    52    530  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8006dcf060 avgcsrva.exe 388    340    19    276  -----  0  2015-10-12 03:56:56 UTC+0000
0xfffffa8004891b10 smss.exe    624    300     0  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80055a6930 csrss.exe   632    624     9    624  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8006014710 smss.exe    676    300     0  -----  1  2015-10-12 03:56:57 UTC+0000
0xfffffa80052d0060 wininit.exe 684    624     3     74  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8003db4140 csrss.exe   696    676    11    634  -----  1  2015-10-12 03:56:57 UTC+0000
0xfffffa8005d5c8f0 winlogon.exe 744    676     3    114  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8005fc0940 services.exe 788    684     7    223  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80052b0700 lsass.exe   796    684     6    799  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80053a6b10 lsm.exe     804    684    10    169  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8006355b10 svchost.exe 912    788    10    372  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa8006393b10 vmacthlp.exe 976    788     3     54  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa800639eb10 WtuSystemSuppo 996    788     7    105  -----  1  2015-10-12 03:56:58 UTC+0000
0xfffffa80063c7b10 svchost.exe 324    788     7    311  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa8006402b10 svchost.exe 652    788    21    517  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa80064ab060 svchost.exe 1056    788    17    441  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80064e5870 svchost.exe 1108    788    12    316  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80064f1060 svchost.exe 1152    788    41    1311  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa8006588b10 svchost.exe 1344    788    17    415  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80063c2600 spoolsv.exe 1476    788    14    361  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa8005ed3330 svchost.exe 1520    788    18    300  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa80064f0600 arnsvc.exe 1604    788     4     69  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80060c0600 avgldsgent.ex 1628    788    30    453  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80065d33c0 avgsvca.exe 1664    788    19    540  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa8006625a20 avgwdsvcx.exe 1712    788    44    909  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80066af4b0 svchost.exe 1776    788    11    266  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa800682e650 taskhost.exe 1072    788    13    264  -----  1  2015-10-12 03:57:02 UTC+0000
0xfffffa80067ccb10 userinit.exe 1452    744     0  -----  1  2015-10-12 03:57:02 UTC+0000
0xfffffa80067e9850 cdm.exe 1292    1056     5    159  -----  1  2015-10-12 03:57:02 UTC+0000
0xfffffa80067f3000 explorer.exe 2056    1452    32    996  -----  1  2015-10-12 03:57:02 UTC+0000
0xfffffa8006956b10 vmtoolsd.exe 2416    2056     6    276  -----  1  2015-10-12 03:57:04 UTC+0000
0xfffffa8006963600 vmtoolsd.exe 2424    2056     0  -----  1  2015-10-12 03:57:04 UTC+0000
```

3. Redirect the output to *processList.txt* file using the right command:

\$ volatility --profile=Win7SP1x64 -f memory.img [Command] [Screenshot showing the command and the output]

```
sansforensics@siftworkstation: ~/Downloads
$ vol.py --profile=Win7SP1x64 -f memory.img pslist > processList.txt
Volatility Foundation Volatility Framework 2.6.1
siftworkstation: ~/Downloads
$ cat processList.txt
Offset(V)  Name      PID  PPID  Thds  Hnds  Sess  Wow64  Start
-----
0xfffffa8003cdd040 System      4      0    93    632  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa80051df9d0 smss.exe    300     4      2    29  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8005cd98f0 avgrsa.exe  328    300     0  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8005cd5900 avgrsa.exe  340    328    52    530  -----  0  2015-10-12 03:56:54 UTC+0000
0xfffffa8006dcf060 avgcsrva.exe 388    340    19    276  -----  0  2015-10-12 03:56:56 UTC+0000
0xfffffa8004891b10 smss.exe    624    300     0  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80055a6930 csrss.exe   632    624     9    624  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8006014710 smss.exe    676    300     0  -----  1  2015-10-12 03:56:57 UTC+0000
0xfffffa80052d0060 wininit.exe 684    624     3     74  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8003db4140 csrss.exe   696    676    11    634  -----  1  2015-10-12 03:56:57 UTC+0000
0xfffffa8005d5c8f0 winlogon.exe 744    676     3    114  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8005fc0940 services.exe 788    684     7    223  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80052b0700 lsass.exe   796    684     6    799  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa80053a6b10 lsm.exe     804    684    10    169  -----  0  2015-10-12 03:56:57 UTC+0000
0xfffffa8006355b10 svchost.exe 912    788    10    372  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa8006393b10 vmacthlp.exe 976    788     3     54  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa800639eb10 WtuSystemSuppo 996    788     7    105  -----  1  2015-10-12 03:56:58 UTC+0000
0xfffffa80063c7b10 svchost.exe 324    788     7    311  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa8006402b10 svchost.exe 652    788    21    517  -----  0  2015-10-12 03:56:58 UTC+0000
0xfffffa80064ab060 svchost.exe 1056    788    17    441  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80064e5870 svchost.exe 1108    788    12    316  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80064f1060 svchost.exe 1152    788    41    1311  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa8006588b10 svchost.exe 1344    788    17    415  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa80063c2600 spoolsv.exe 1476    788    14    361  -----  0  2015-10-12 03:56:59 UTC+0000
0xfffffa8005ed3330 svchost.exe 1520    788    18    300  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa80064f0600 arnsvc.exe 1604    788     4     69  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80060c0600 avgldsgent.ex 1628    788    30    453  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80065d33c0 avgsvca.exe 1664    788    19    540  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa8006625a20 avgwdsvcx.exe 1712    788    44    909  -----  1  2015-10-12 03:57:00 UTC+0000
0xfffffa80066af4b0 svchost.exe 1776    788    11    266  -----  0  2015-10-12 03:57:00 UTC+0000
0xfffffa800682e650 taskhost.exe 1072    788    13    264  -----  1  2015-10-12 03:57:02 UTC+0000
0xfffffa80067ccb10 userinit.exe 1452    744     0  -----  1  2015-10-12 03:57:02 UTC+0000
```

4. Open your *processlist.txt* file and answer the following question:



Based on the results, which browser(s) were running at the time of the acquisition?

```
0xfffffa8006939b10 chrome.exe 3564 2056 31 765 1 1 2015-10-1
2 11:34:25 UTC+0000
0xfffffa8004639060 chrome.exe 4316 3564 5 166 1 1 2015-10-1
2 11:34:26 UTC+0000
0xfffffa80067e2060 chrome.exe 3120 3564 8 167 1 1 2015-10-1
2 11:34:38 UTC+0000
0xfffffa8006b26b10 iexplore.exe 768 2056 16 542 1 0 2015-10-1
2 11:34:42 UTC+0000
0xfffffa80068cb060 iexplore.exe 4352 768 53 879 1 1 2015-10-1
2 11:34:43 UTC+0000
0xfffffa8006c6e060 iexplore.exe 3684 768 31 711 1 1 2015-10-1
2 11:35:03 UTC+0000
```

What application was using Process ID 3780?

```
0xfffffa80042a1b10 AcroRd32.exe 3780 4368 7 318 1 1 2015-10-1
2 11:05:26 UTC+0000
```

What application was used to acquire the contents of memory?

```
0xfffffa8006fb5270 FTK Imager.exe 3460 2056 14 375 1 1 2015-10-1
2 11:41:01 UTC+0000
sansforensics@siftworkstation: ~/Downloads
$ cat memory.img
"memory.img" selected (5.4 GB)
```

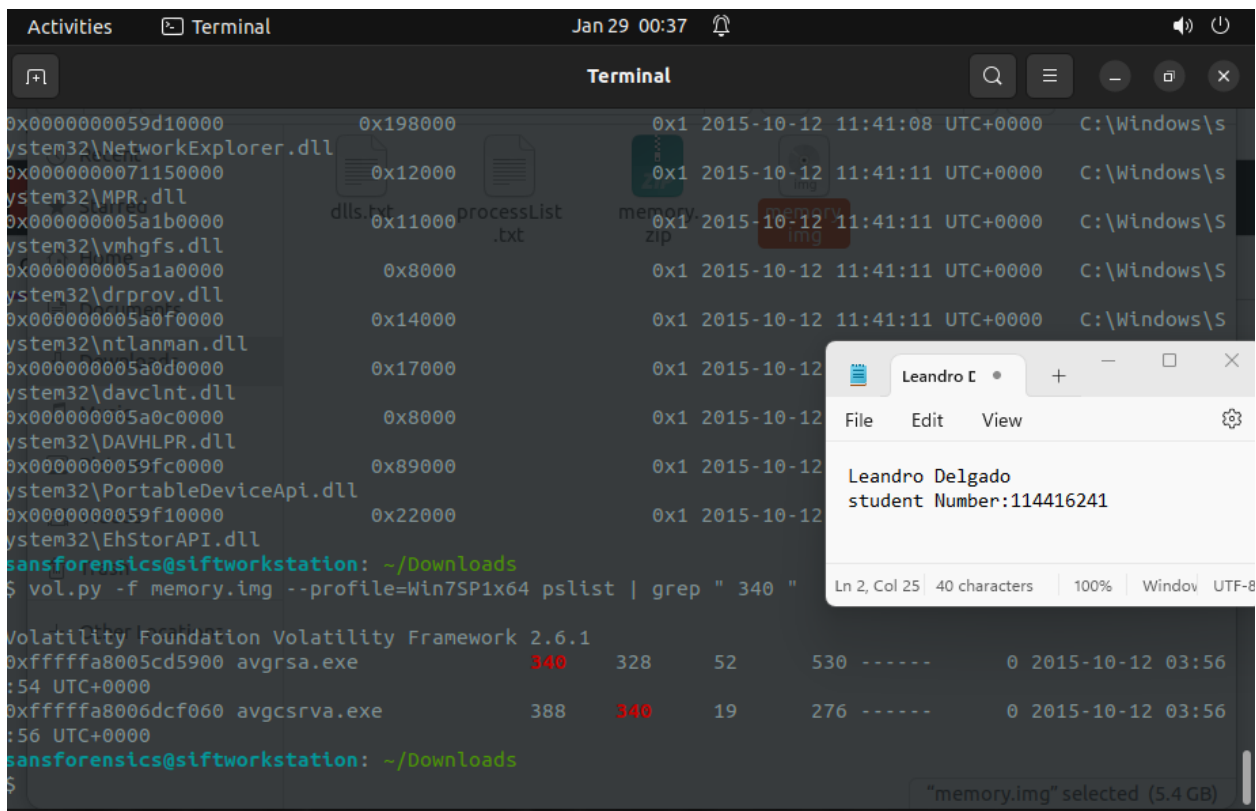
Part 4: Use *the* Plugin to List Dynamic Link Libraries (DLLs)

- At the prompt, type the right command and press the “Enter” key to display the list of DLLs, which were running at the time of the acquisition:

\$ volatility -f memory.img --profile=Win7SP1x64 [Command] [Screenshot showing the command and the output]

```
sansforensics@siftworkstation: ~/Downloads
$ vol.py -f memory.img --profile=Win7SP1x64 dlllist
Volatility Foundation Volatility Framework 2.6.1
*****
System pid: 4 .txt zip img
Unable to read PEB for task.
*****
smss.exe pid: 300
Command line : \SystemRoot\System32\smss.exe
*****
Base Music Size LoadCount LoadTime Path
-----
0x0000000048170000 0x20000 0xffff 1970-01-01 00:00:00 UTC+0000 \SystemRoot\
System32\smss.exe
0x0000000076e0000 0x1a9000 0xffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\S
YSTEM32\ntdll.dll
*****
avgrsa.exe pid: 328
Unable to read PEB for task.
*****
avgrsa.exe pid: 340
Command line : c:\PROGRA-2\AVG\Av\avgrsa.exe /boot
*****
Base Size LoadCount LoadTime
-----
0x0000000013fde000 0x130000 0xffff 1970-01-01 00:00:00 UTC+0000 c:\PROGRA-2\
```


- Recover the DLLs from memory for *Process ID 340*, i.e., *smss.exe*, and store them in the temporary directory, type the right command at the prompt and press the “Enter” key:



```

Activities  Terminal  Jan 29 00:37
Terminal
0x0000000059d10000 0x198000 0x1 2015-10-12 11:41:08 UTC+0000 C:\Windows\s
ystem32\NetworkExplorer.dll
0x0000000071150000 0x12000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\s
ystem32\MPR.dll
0x000000005a1b0000 0x11000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\vmhghfs.dll
0x000000005a1a0000 0x8000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\drprov.dll
0x000000005a0f0000 0x14000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\ntlanman.dll
0x000000005a0d0000 0x17000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\davclnt.dll
0x000000005a0c0000 0x8000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\DAVHLPR.dll
0x0000000059f00000 0x89000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\PortableDeviceApi.dll
0x0000000059f10000 0x22000 0x1 2015-10-12 11:41:11 UTC+0000 C:\Windows\S
ystem32\EhStorAPI.dll
sansforensics@siftworkstation: ~/Downloads
$ vol.py -f memory.img --profile=Win7SP1x64 pslist | grep " 340 "

Volatility Foundation Volatility Framework 2.6.1
0xfffffa8005cd5900 avgrsa.exe 340 328 52 530 ----- 0 2015-10-12 03:56
:54 UTC+0000
0xfffffa8006dcf060 avgcsrva.exe 388 340 19 276 ----- 0 2015-10-12 03:56
:56 UTC+0000
sansforensics@siftworkstation: ~/Downloads

```

- List all recovered DLLs:

```

sansforensics@siftworkstation: ~/Downloads
$ vol.py -f memory.img --profile=Win7SP1x64 dlldump -p 340 --dump-dir=/tmp/
Volatility Foundation Volatility Framework 2.6.1
Process(V)  Name  Module Base  Module Name  Result
-----
0xfffffa8005cd5900 avgrsa.exe 0x0000000013fde0000 avgrsa.exe OK: module.340.13e0d5900.13fde0000.dll
0xfffffa8005cd5900 avgrsa.exe 0x0000000076e60000 ntdll.dll OK: module.340.13e0d5900.76e60000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefef30000 avgcnla.dll OK: module.340.13e0d5900.7fefef30000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefec20000 avgcerta.dll OK: module.340.13e0d5900.7fefec20000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefec90000 avgclita.dll OK: module.340.13e0d5900.7fefec90000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefebb0000 avgdetaallocatord.dll OK: module.340.13e0d5900.7fefebb0000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefeb0000 avgcclla.dll OK: module.340.13e0d5900.7fefeb0000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefefe0000 avgsysa.dll OK: module.340.13e0d5900.7fefefe0000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefdef0000 avgntsqlitea.dll OK: module.340.13e0d5900.7fefdef0000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefff10000 avgloga.dll OK: module.340.13e0d5900.7fefff10000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefde50000 avgcomma.dll OK: module.340.13e0d5900.7fefde50000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefed10000 avgchjwa.dll OK: module.340.13e0d5900.7fefed10000.dll
0xfffffa8005cd5900 avgrsa.exe 0x000007fefedd0000 avgntopenssla.dll OK: module.340.13e0d5900.7fefedd0000.dll
sansforensics@siftworkstation: ~/Downloads

```

- Malware scans and reverse engineering can be performed on the recovered DLL files.

Part 5: Network Connections Analysis

- At the prompt, type the right command and press the “Enter” key to display the list of the network connections, which were established at the time of the acquisition. Output the connections to a file names netscan.txt and screenshot the results.

```

$ vol.py -f memory.img --profile=Win7SP1x64 netstat
Volatility Foundation Volatility Framework 2.6.1
Offset(P) Proto Local Address Foreign Address State PId Owner Created
0x97b22890 TCPv4 192.168.133.149:49297 23.62.6.66:443 CLOSED 3564 chrome.exe
0xa3357010 TCPv4 127.0.0.1:49172 127.0.0.1:7112 ESTABLISHED 2500 vprot.exe
0xb9fe3890 TCPv4 192.168.133.149:49297 23.62.6.66:443 CLOSED 3564 chrome.exe
0x13ac6dcf0 TCPv4 127.0.0.1:7112 127.0.0.1:49171 ESTABLISHED 2568 loggingserver
0x13bea4cc0 UDPv6 :::1:1900 *: ESTABLISHED 2892 svchost.exe
0x13c2ef250 TCPv4 192.168.133.149:50030 173.194.121.25:443 ESTABLISHED 4352 iexplore.exe
0x13cc208c0 TCPv4 192.168.133.149:49828 74.125.228.237:80 CLOSED 4352 iexplore.exe
0x13cc43450 TCPv4 192.168.133.149:49719 173.194.121.57:443 CLOSED 3564 chrome.exe
0x13cc43cf0 TCPv4 192.168.133.149:49718 74.125.228.207:443 CLOSED 3564 chrome.exe
0x13cc4c940 TCPv4 192.168.133.149:50013 174.129.201.215:80 ESTABLISHED 4352 iexplore.exe
0x13cc60cf0 TCPv4 192.168.133.149:49777 64.74.232.42:80 CLOSED 4352 iexplore.exe
0x13cc66cf0 TCPv4 192.168.133.149:49778 50.31.185.39:80 CLOSED 4352 iexplore.exe
0x13cc688c0 TCPv4 192.168.133.149:49953 173.194.121.13:80 ESTABLISHED 4352 iexplore.exe
0x13cc6e010 TCPv4 :-:49760 :-:80 CLOSED 4352 iexplore.exe
0x13ccc8b60 TCPv4 :-:49838 :-:80 CLOSED 4352 iexplore.exe
0x13ccc8cf0 TCPv4 192.168.133.149:50028 63.140.35.162:80 CLOSED 4352 iexplore.exe
0x13ccc84a0 TCPv4 :-:49723 :-:443 CLOSED 3684 iexplore.exe
0x13d29d390 UDPv6 fe80::e1ae:9e4c:5f3b:6367:546 *: ESTABLISHED 652 svchost.exe
0x13d40a8c0 UDPv4 127.0.0.1:11900 *: ESTABLISHED 2892 svchost.exe
0x13d4f6ec0 UDPv6 :::1:57293 *: ESTABLISHED 2892 svchost.exe
0x13d545ec0 UDPv4 127.0.0.1:57294 *: ESTABLISHED 2892 svchost.exe
0x13d5ba5b0 UDPv4 0.0.0.0:0 *: ESTABLISHED 1344 svchost.exe
0x13d5ba5b0 UDPv4 0.0.0.0:5355 *: ESTABLISHED 1344 svchost.exe
0x13d6efe00 UDPv6 :::5355 *: ESTABLISHED 1344 svchost.exe
0x13d22eef0 TCPv4 0.0.0.0:445 0.0.0.0: LISTENING 4 System
0x13d22eef0 TCPv4 :-:445 :-: LISTENING 4 System
0x13d2cabe0 TCPv4 0.0.0.0:7112 0.0.0.0: LISTENING 2568 loggingserver
0x13ce0a010 TCPv4 192.168.133.149:50080 50.16.243.245:80 ESTABLISHED 4352 iexplore.exe
0x13ce1f980 TCPv4 192.168.133.149:49957 54.166.155.65:80 CLOSE_WAIT 4352 iexplore.exe
0x13ce1f7f0 TCPv4 192.168.133.149:50014 174.129.201.215:80 ESTABLISHED 4352 iexplore.exe
0x13ce29670 TCPv4 192.168.133.149:50095 23.62.6.49:80 ESTABLISHED 4352 iexplore.exe
0x13ce29510 TCPv4 :-:49734 :-:80 CLOSED 4352 iexplore.exe

```

- Find out what internet website Windows Internet Explorer was connected to (IP address: 173.194.121.25) using IP lookup tool at <http://whatismyipaddress.com/ip-lookup> [Answer]

IP Details For: 173.194.121.25

Decimal:	2915203353
Hostname:	173.194.121.25
ASN:	15169
ISP:	Google LLC
Services:	Datacenter
Country:	United States
State/Region:	Virginia
City:	Dulles
Latitude:	38.9517 (38° 57' 6.00" N)
Longitude:	-77.4481 (77° 26' 53.01" W)

CLICK TO CHECK BLACKLIST STATUS