

DumpMe Blue Team Lab

Youssef sherif

20221041587

Question one

```
(kali㉿kali)-[~/volatility3]
$ sha1sum Triage-Memory.mem
c95e8cc8c946f95a109ea8e47a6800de10a27abd
```

Q1 ☒ What is the SHA1 hash of Triage-Memory.mem (memory dump)

Weight : 0 | Solved : 2116 | Average Solve Time: 3min

c95e8cc8c946f95a109ea8e47a6800de10a27abd

Question two

```
(kali㉿kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem windows.info.Info
```

```
Kernel Base      0xf80002808000
DTB              0x187000
Symbols file:///home/kali/volatility3/volatility3/symbols/windows/ntkrnlmp.pdb/2E37F962D699492CAAF3F9F4E9770B1D-2.json.xz
Is64Bit          True
IsPAE            False
layer_name       0 WindowsIntel32e
memory_layer     1 FileLayer
KdDebuggerDataBlock 0xf800029f80a0
NTBuildLab       7601.18741.amd64fre.win7sp1_gdr.
CSDVersion       1
KdVersionBlock   0xf800029f8068
Major/Minor      15.7601
MachineType      34404
KeNumberProcessors 2
SystemTime       2019-03-22 05:46:00
NtSystemRoot     C:\Windows
NtProductType    NtProductWinNt
NtMajorVersion   6
NtMinorVersion   1
PE MajorOperatingSystemVersion 6
PE MinorOperatingSystemVersion 1
PE Machine       34404
PE TimeDateStamp Tue Feb 3 02:25:01 2015
```

Given this info to ChatGPT to decide which type is that

Selecting a Volatility Profile



The suggested Volatility profile for further analysis of this memory image would be 'Win7SP1x64'

Q2

What volatility profile is the most appropriate for this machine?

Weight : 0 | Solved : 2073 | Average Solve Time: 1min

Win7SP1x64

Question three

Listing all process

(kali@kali) - [~/volatility3]

\$ python vol.py -f Triage-Memory.mem windows.pslist

2432	476	SearchIndexer.exe	0xfa8005500030	13	700	0	False
2019-03-22 05:32:17.000000 N/A Disabled							
2628	476	wmpnetwk.exe	0xfa80055b0060	9	210	0	False
2019-03-22 05:32:18.000000 N/A Disabled							
2888	476	svchost.exe	0xfa8005c4ab30	11	152	0	False
2019-03-22 05:32:20.000000 N/A Disabled							
3032	1432	notepad.exe	0xfa80054f9060	1	60	1	False

Q3

What was the process ID of notepad.exe?


Weight : 0 | Solved : 2084 | Average Solve Time: 1min

3032

Question four

```
(kali@kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem windows.pstree.PsTree

Volatility 3 Framework 2.7.0
Progress: 100.00 PDB scanning finished
PID PPID ImageFileName Offset(V) Threads Handles SessionId W
ow64 CreateTime ExitTime Audit Cmd Path
4 0 System 0xfa8003c72b30 87 547 N/A False 2019-
03-22 05:31:55.000000 N/A - -
* 252 4 smss.exe 0xfa8004616040 2 30 N/A False
2019-03-22 05:31:55.000000 N/A \Device\HarddiskVolume2\Windows\Syste
m32\smss.exe \SystemRoot\System32\smss.exe \SystemRoot\System32\smss.exe
332 324 csrss.exe 0xfa80050546b0 10 516 0 False
2019-03-22 05:31:58.000000 N/A \Device\HarddiskVolume2\Windows\Syste
W64\wscript.exe "C:\Windows\System32\wscript.exe" //B //NOLOGO %TEMP%\vhjReUD
EuumrX.vbs C:\Windows\SysWOW64\wscript.exe
*** 3496 5116 UWkpjFjDzM.exe 0xfa8005a1d9e0 5 109 1 T
rue 2019-03-22 05:35:33.000000 N/A \Device\HarddiskVolume2\Users
\Bob\AppData\Local\Temp\rad93398.tmp\UWkpjFjDzM.exe "C:\Users\Bob\AppData
\Local\Temp\rad93398.tmp\UWkpjFjDzM.exe" Come, C:\Users\Bob\AppData\Local\Te
mn\rad93398.tmp\UWkpjFjDzM.exe
```

Q4  Name the child process of wscript.exe.


Weight : 0 | Solved : 2064 | Average Solve Time: 1min

UWkpjFjDzM.exe

Question five

```
File Actions Edit View Help
$ python vol.py -f Triage-Memory.mem windows.netstat.NetStat

0xfa80053fc790 TCPv4 0.0.0.0 49262 0.0.0.0 0 LISTENING - - -
0xfa800481a570 TCPv4 0.0.0.0 49263 0.0.0.0 0 LISTENING - - -
0xfa80059683e0 UDPv4 10.0.0.101 137 * 0 4 System 2019-03-22 05:32:06.000000
0xfa8005994250 UDPv4 10.0.0.101 138 * 0 4 System 2019-03-22 05:32:06.000000
0xfa800563f530 UDPv6 fe80::7775:ef30:b018:7807 546 * 0 766 svchost.exe 2019-03-22 05:16:5
```

Q5  What was the IP address of the machine at the time the RAM dump was created?

Weight : 0 | Solved : 1014 | Average Solve Time: 1min

10.0.0.101

Question six

```
python vol.py -f Triage-Memory.mem windows.netscan.NetScan

Volatility 3 Framework 2.7.0
Progress: 100.00
PDB scanning finished
Offset Proto LocalAddr LocalPort ForeignAddr ForeignPort State PID Owner Created
x13e02bcf0 TCPv4 - 49220 72.51.60.132 443 CLOSED 4048 POWERPNT.EXE -
x13e035790 TCPv4 - 49223 72.51.60.132 443 CLOSED 4048 POWERPNT.EXE -
x13e036470 TCPv4 - 49224 72.51.60.132 443 CLOSED 4048 POWERPNT.EXE -
x13e057300 UDPv4 10.0.0.101 55736 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05b4f0 UDPv6 ::1 55735 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05b790 UDPv6 fe80::7475:ef30:be18:7807 55734 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05d4b0 UDPv6 fe80::7475:ef30:be18:7807 1900 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05dec0 UDPv4 127.0.0.1 55737 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05e3f0 UDPv4 10.0.0.101 1900 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e05eab0 UDPv6 ::1 1900 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e064d70 UDPv4 127.0.0.1 1900 * 0 2888 svchost.exe 2019-03-22 05:32:20.000000
x13e2348a0 TCPv4 - 49366 192.168.206.181 389 CLOSED - N/A
x13e258010 UDPv4 127.0.0.1 55560 * 0 5116 wscript.exe 2019-03-22 05:35:32.000000
x13e2c6b10 TCPv4 0.0.0.0 21 0.0.0.0 LISTENING 1476 FileZilla Serv -
x13e2c6b10 TCPv6 :: 21 :: 0 LISTENING 1476 FileZilla Serv -
x13e2c7850 TCPv6 ::1 14147 :: 0 LISTENING 1476 FileZilla Serv -
x13e2c96b0 TCPv4 127.0.0.1 14147 0.0.0.0 LISTENING 1476 FileZilla Serv -
x13e2c9be0 TCPv4 0.0.0.0 21 0.0.0.0 LISTENING 1476 FileZilla Serv -
x13e305a50 UDPv4 0.0.0.0 5355 * 0 232 svchost.exe 2019-03-22 05:32:09.000000
x13e360be0 UDPv4 0.0.0.0 63790 * 0 - 2019-03-22 05:45:47.000000
x13e397190 TCPv4 10.0.0.101 49217 10.0.0.106 4444 ESTABLISHED 3496 UWkpjFjDzM.exe N/A
```

Q6 Based on the answer regarding the infected PID, can you determine the IP of the attacker?

Weight : 0 | Solved : 1971 | Average Solve Time: 1min

10.0.0.106

Question seven

```
(kali@kali)~/volatility3
$ python vol.py -f Triage-Memory.mem dlllist | grep VCRUNTIME140.dll
136ressOfficeClickToR 0x7fefa5c0000can0x16000nVCRUNTIME140.dll C:\Program Files\Common Files\Microsoft Shared\ClickToRun\VCRUNTIME140.dll 2019
03-22 05:32:05.000000 Disabled

(kali@kali)~/volatility3
$ python vol.py -f Triage-Memory.mem dlllist | grep VCRUNTIME140.dll | awk '{print $2}'
fficeClickToR.00 PDB scanning finished

(kali@kali)~/volatility3
$ python vol.py -f Triage-Memory.mem dlllist | grep VCRUNTIME140.dll | awk '{print $2}'|wc -l
progress: 100.00 PDB scanning finished

(kali@kali)~/volatility3
$ python vol.py -f Triage-Memory.mem dlllist | grep VCRUNTIME140.dll | awk '{print $2}'|wc -l
progress: 100.00 PDB scanning finished
```

Q7 How many processes are associated with VCRUNTIME140.dll?

Weight : 0 | Solved : 1864 | Average Solve Time: 17min


5

Question eight

Dump the malicious process then get its md5-hash

```
(kali@kali)~[~/volatility3]
$ python vol.py -f Triage-Memory.mem -o /home/kali/Desktop windows.dumpfiles --pid 3496

Volatility 3 Framework 2.7.0
Progress: 100.00 PDB scanning finished
Cache FileObject FileName Result
ImageSectionObject 0xfa8005bdf260 winhttp.dll file.0xfa8005bdf260.0xfa80042849b0.ImageSectionObject.winhttp.dll.img
ImageSectionObject 0xfa8005a55f20 UWkpjFjDzM.exe file.0xfa8005a55f20.0xfa8005b55ba0.ImageSectionObject.UWkpjFjDzM.exe.img
ImageSectionObject 0xfa800460c3f0 apisetschema.dll file.0xfa800460c3f0.0xfa8004612830.ImageSectionObject.apisetsche
ImageSectionObject 0xfa80043d4f20 mpr.dll file.0xfa80043d4f20.0xfa8004912c30.ImageSectionObject.mpr.dll.img
ImageSectionObject 0xfa8005b06070 srvcli.dll file.0xfa8005b06070.0xfa8005b009b0.ImageSectionObject.srvcli.dll.img
ImageSectionObject 0xfa80057fd070 wkscli.dll file.0xfa80057fd070.0xfa8005ace8b0.ImageSectionObject.wkscli.dll.img
ImageSectionObject 0xfa8005584b30 cscapi.dll file.0xfa8005584b30.0xfa8005bb0690.ImageSectionObject.cscapi.dll.img
ImageSectionObject 0xfa8005b027d0 netapi32.dll file.0xfa8005b027d0.0xfa8005b03e60.ImageSectionObject.netapi32.dll.img
ImageSectionObject 0xfa8005b031c0 netutils.dll file.0xfa8005b031c0.0xfa8005b05cd0.ImageSectionObject.netutils.dll.img
ImageSectionObject 0xfa80040c2b20 webio.dll file.0xfa80040c2b20.0xfa80040b9a90.ImageSectionObject.webio.dll.img
ImageSectionObject 0xfa8005746f20 apphelp.dll file.0xfa8005746f20.0xfa8005747560.ImageSectionObject.apphelp.dll.img
ImageSectionObject 0xfa8005027b40 kernel32.dll file.0xfa8005027b40.0xfa80050304c0.ImageSectionObject.kernel32.dll.img
ImageSectionObject 0xfa800598bd10 wow64win.dll file.0xfa800598bd10.0xfa80058a0330.ImageSectionObject.wow64win.dll.img
ImageSectionObject 0xfa8005abd070 WSHTCPIP.DLL file.0xfa8005abd070.0xfa8005ab6230.ImageSectionObject.WSHTCPIP.DLL.img
ImageSectionObject 0xfa8005b1cf20 rsaenh.dll file.0xfa8005b1cf20.0xfa8005b1cd10.ImageSectionObject.rsaenh.dll.img
ImageSectionObject 0xfa8005b84600 uxtheme.dll file.0xfa8005b84600.0xfa8005b85010.ImageSectionObject.uxtheme.dll.img
ImageSectionObject 0xfa8005b0c880 cryptsp.dll file.0xfa8005b0c880.0xfa8005b0c4c0.ImageSectionObject.cryptsp.dll.img
ImageSectionObject 0xfa8005aaedd0 dhcpcsvc6.dll file.0xfa8005aaedd0.0xfa8005aaaa10.ImageSectionObject.dhcpcsvc6.dll.img
ImageSectionObject 0xfa8005ab88c0 mswsock.dll file.0xfa8005ab88c0.0xfa8005ab9b20.ImageSectionObject.mswsock.dll.img
ImageSectionObject 0xfa8005aad230 dhcpcsvc.dll file.0xfa8005aad230.0xfa8005aaf600.ImageSectionObject.dhcpcsvc.dll.img
ImageSectionObject 0xfa8005a63bc0 IPHLPAPI.DLL file.0xfa8005a63bc0.0xfa8005a87c90.ImageSectionObject.IPHLPAPI.DLL.img
ImageSectionObject 0xfa800464adc0 winnsi.dll file.0xfa800464adc0.0xfa8005a89400.ImageSectionObject.winnsi.dll.img
ImageSectionObject 0xfa80059256d0 wow64cpu.dll file.0xfa80059256d0.0xfa80059d3e30.ImageSectionObject.wow64cpu.dll.img
ImageSectionObject 0xfa8005034820 cryptbase.dll file.0xfa8005034820.0xfa800503ac50.ImageSectionObject.cryptbase.dll.img
ImageSectionObject 0xfa8005bf24e0 userenv.dll file.0xfa8005bf24e0.0xfa8005592a90.ImageSectionObject.userenv.dll.img
ImageSectionObject 0xfa8005a3d340 wow64.dll file.0xfa8005a3d340.0xfa8005966e30.ImageSectionObject.wow64.dll.img
ImageSectionObject 0xfa8005791450 profapi.dll file.0xfa8005791450.0xfa80059a2420.ImageSectionObject.profapi.dll.img
ImageSectionObject 0xfa8003f75570 wsock32.dll file.0xfa8003f75570.0xfa8003f76d00.ImageSectionObject.wsock32.dll.img
```

Q8  After dumping the infected process, what is its md5 hash?

Weight : 0 | Solved : 1717 | Average Solve Time: 4min

690ea20bc3bdfb328e23005d9a80c290

Question nine

Using volatility hashdum command

The LM hash (LanMan hash) is a cryptographic hash function used in older versions of Microsoft Windows operating systems to store user passwords.

```
Volatility Foundation Volatility Framework 2.6.1
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bob:1000:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```


Q9 What is the LM hash of Bob's account?

Weight : 0 | Solved : 1718 | Average Solve Time: 1min

aad3b435b51404eeaad3b435b51404ee

Question ten

```
(kali㉿kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem windows.vadinfo > vad.txt
```

svchost.exe	0xfffffa800577c720	0x340000	0x43ffff	VadS	PAGE	
svchost.exe	0xfffffa800577bb80	0x60000	0x60fff	Vadm	PAGE_READWRITE	1
svchost.exe	0xfffffa800577b980	0x40000	0x40fff	Vad	PAGE_READONLY	0
svchost.exe	0xfffffa800577c8e0	0x20000	0x20fff	Vad	PAGE_WRITECOPY	1
svchost.exe	0xfffffa800577cf80	0x10000	0x1ffff	Vad	PAGE_READWRITE	0
svchost.exe	0xfffffa800577ba10	0x30000	0x33fff	Vad	PAGE_READONLY	0
svchost.exe	0xfffffa8005776ea0	0x50000	0x50fff	VadS	PAGE_READWRITE	1
svchost.exe	0xfffffa800577cbc0	0x220000	0x286fff	Vad	PAGE	

Q10 What memory protection constants does the VAD node at 0xfffffa800577ba10 have?

Weight : 0 | Solved : 1612 | Average Solve Time: 1min

page_readonly

Question eleven

OfficeClickToR	0xfffffa8005ab3a50	0x2f10000	0x300ffff	VadS	PAGE_READWRITE	4	1
OfficeClickToR	0xfffffa8003c9b9a0	0x3250000	0x326ffff	VadS	PAGE_NOACCESS	32	1
OfficeClickToR	0xfffffa80058d9e00	0x3280000	0x337ffff	VadS	PAGE_READWRITE	4	1
OfficeClickToR	0xfffffa80052652b0	0x33c0000	0x33dffff	VadS	PAGE_NOACCESS	32	1
OfficeClickToR	0xfffffa8003f416d0	0x33a0000	0x33bffff	VadS	PAGE_NOACCESS	32	1
OfficeClickToR	0xfffffa80056b7780	0x3400000	0x341ffff	VadS	PAGE_NOACCESS	32	1

Search only for 0x33c000

Q11 What memory protection did the VAD starting at 0x00000000033c0000 and ending at 0x00000000033dffff have?

Weight : 0 | Solved : 1590 | Average Solve Time: 1min


page_noaccess

Question twelve

```
(kali㉿kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem windows.cmdline > cmd.txt
```

```
4520 chrome.exe "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --type=gpu-process
4688 chrome.exe "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --type=renderer --t
5116 wscript.exe "C:\Windows\System32\wscript.exe" //B //NOLOGO %TEMP%\vhjReUDEuumrX.vbs
7496 Ullkn3iDzM.exe "C:\Users\Bob\AppData\Local\Temp\rad92398.tmp\Ullkn3iDzM.exe"
```

Here it is the answer

Q12  There was a VBS script that ran on the machine. What is the name of the script? (submit without file extension)


Weight : 0 | Solved : 1608 | Average Solve Time: 2min

vhjreudeuumrx

Question 13

```
(kali㉿kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem shimcache | grep "2019-03-07 23:06:58 UTC"
```

```
Volatility Foundation Volatility Framework 2.6.1
2019-03-07 23:06:58 UTC+0000 \??C:\Program Files (x86)\Microsoft\Skype for Desktop\Skype.exe
```

Q13  An application was run at 2019-03-07 23:06:58 UTC. What is the name of the program (Include extension)

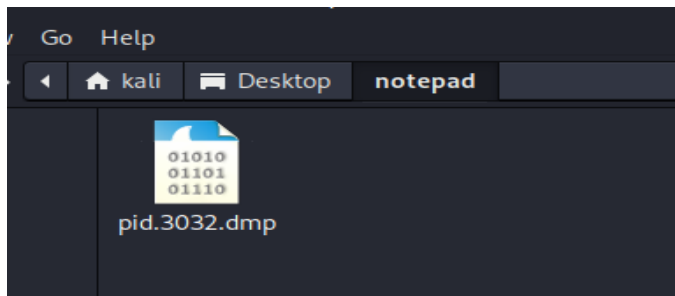
Weight : 0 | Solved : 1534 | Average Solve Time: 14min

skype.exe

Question 14

Dump notepad to a folder

```
python vol.py -f Triage-Memory.mem windows.memdump -p 3032 -D home
/kali/Desktop/notepad
```



Then extract all text from the dmp

```
(kali@kali)-[~/Desktop/notepad]
$ strings -e l pid.3032.dmp > notepad_text.txt
```

```
(kali@kali)-[~/Desktop/notepad]
$ grep flag notepad_text.txt
flag<REDBULL_IS_LIFE>
```

Q14 What was written in notepad.exe at the time when the memory dump was captured?

Weight : 0 | Solved : 1453 | Average Solve Time: 9min

flag<REDBULL_IS_LIFE>

Question 15

```
(kali@kali)-[~/volatility3]
$ python vol.py -f Triage-Memory.mem windows.mftscan.MFTScan >mft.txt
Progress: 100.00 PDB scanning finished
```

Then search in the file created for the 59045

* 0xf980015e9128	FILE	59044	2	File	Archive	FILE_NAME	2019-03-08 03:05:02.000000	2019-03-08 03:05:02.000000	2019-03-08 03:05:02.000000	2019-03-08 03:05:02.000000	CapES0DownToDown1
0xf980015e9450	FILE	59045	2	File	N/A	STANDARD_INFORMATION	2019-03-17 06:50:07.000000	2019-03-17 07:04:43.000000	2019-03-17 07:04:43.000000	2019-03-17 07:04:42.000000	N/A
* 0xf980015e94b0	FILE	59045	2	File	Archive	FILE_NAME	2019-03-17 06:50:07.000000	2019-03-17 07:04:43.000000	2019-03-17 07:04:43.000000	2019-03-17 07:04:42.000000	EMPLOY~1.XLS

Q15  What is the short name of the file at file record 59045?

Weight : 0 | Solved : 1460 | Average Solve Time: 1min

EMPLOY~1.XLS

Last question

0xffffffff8005a80060	wscript.exe	5116	3952	8	312	1	1	2019-03-22 05:35:32 UTC+0000
0xffffffff8005a1d9e0	UwkpjFjDzM.exe	3496	5116	5	109	1	1	2019-03-22 05:35:33 UTC+0000

The malicious process pid

Q16  This box was exploited and is running meterpreter. What was the infected PID?

Weight : 0 | Solved : 1535 | Average Solve Time: 2min

3496