



# BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Lab 1: Memory Forensic

GVHD: Đoàn Minh Trung

**1. THÔNG TIN CHUNG:**

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.021.ATTN

STT	Họ và tên	MSSV	Email
1	Phạm Ngọc Thơ	21522641	21522641@gm.uit.edu.vn
2	Hà Thị Thu Hiền	21522056	21522056@gm.uit.edu.vn

**2. NỘI DUNG THỰC HIỆN:<sup>1</sup>**

STT	Công việc	Kết quả tự đánh giá
1	Yêu cầu 1 (đã báo cáo ở lớp)	100%
2	Yêu cầu 2 (đã báo cáo ở lớp)	100%
3	Yêu cầu 3	100%
4	Yêu cầu 4	100%
5	Yêu cầu 5	100%

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

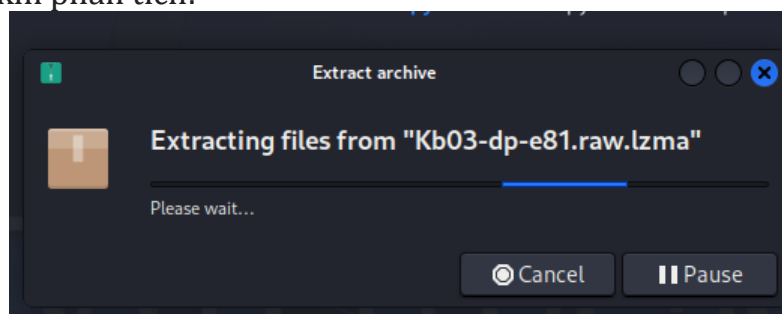
<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

## BÁO CÁO CHI TIẾT

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.
- Một tệp có phần mở rộng tệp LZMA là tệp nén, do đó cần phải giải nén file trước khi phân tích:



- Do vai trò của Dump File là ghi lại chi tiết tình hình và những vấn đề mà máy tính đang gặp phải trước hoặc tại thời điểm xảy ra sự cố, nên nếu file này là file dump từ bộ nhớ máy ảo, nó sẽ chứa các thông tin trên. Em sẽ sử dụng **pslist** để xem thông tin liệu lịch sử tiến trình file có ghi lại hay không:

```
(ngthow_kali@kali)~[~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Documents/Kb03-dp-e81.raw --profile=Win10x64 pslist
Volatility Foundation Volatility Framework 2.6.1
Offset(V)  Name  PID  PPID  Thds  Hnds  Sess  Wow64  Start  Exit
```

0xfffffe00032553780	System	4	0	126	0		0	2016-04-04 16:12:33 UTC+0000	
0xfffffe0003389c040	smss.exe	268	4	2	0		0	2016-04-04 16:12:33 UTC+0000	
0xfffffe0003381b080	csrss.exe	344	336	8	0	0	0	2016-04-04 16:12:33 UTC+0000	
0xfffffe000325ba080	wininit.exe	404	336	1	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000325c7080	csrss.exe	412	396	9	0	1	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00033ec6080	winlogon.exe	460	396	2	0	1	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00033efb440	services.exe	484	404	3	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00033f08080	lsass.exe	492	404	6	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00033ec5780	svchost.exe	580	484	16	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00034202280	svchost.exe	612	484	9	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000341cb640	dwm.exe	712	460	8	0	1	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00034222780	svchost.exe	796	484	45	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000342a7780	VBosService.ex	828	484	10	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000342ad780	svchost.exe	844	484	8	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000342c0080	svchost.exe	852	484	6	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000342dd780	svchost.exe	892	484	18	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000342bc780	svchost.exe	980	484	17	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe00034377780	svchost.exe	608	484	17	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000343e7780	spoolsv.exe	1072	484	8	0	0	0	2016-04-04 16:12:34 UTC+0000	
0xfffffe000343e9780	svchost.exe	1092	484	23	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe0003442a780	rundll32.exe	1148	796	1	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe00034494780	CompatTelRunne	1224	1148	9	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe00034495780	svchost.exe	1276	484	10	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe0003461d780	svchost.exe	1564	484	5	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe000345da780	wlms.exe	1616	484	2	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe00034623780	MsMpEng.exe	1628	484	24	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe000343b2340	cygrunsrv.exe	1832	484	4	0	0	0	2016-04-04 16:12:35 UTC+0000	
0xfffffe0003479b780	cygrunsrv.exe	1976	1832	0	0	0	0	2016-04-04 16:12:36 UTC+0000	2016-04-04 16:12:36 UTC+0000

- Tìm flag cho file tài nguyên bên trên. Biết rằng flag có định dạng CTF{flag}.
- Sử dụng lệnh **imageinfo** của volatility để phân tích file, xem được hệ điều hành của máy ảo là **Windows 10**.

Báo cáo môn học  
HOC KỲ I – NĂM HỌC 2024-2025

- Tải tệp về và giải nén:

```
(ngthow_kali@kali)-[~/Downloads]
$ ls
Nessus-10.6.2-debian10_amd64.deb  burpsuite_community_linux_v2023_10_3_5.sh  ch2.tbz2

(ngthow_kali@kali)-[~/Downloads]
$ tar -xvf ch2.tbz2
ch2.dmp
```

- Challenge có cung cấp mã hash của file để chúng ta kiểm tra liệu file tải về có đúng chưa, nên em sẽ băm file em tải, đối chiếu với giá trị băm web cung cấp. Kết quả trùng nhau, nên OK sẽ tiến hành phân tích file tìm flag!

```
(ngthow_kali@kali)-[~/volatility]
$ md5sum /home/ngthow_kali/Downloads/ch2.dmp
e3a902d4d44e0f7bd9cb29865e0a15de  /home/ngthow_kali/Downloads/ch2.dmp
```

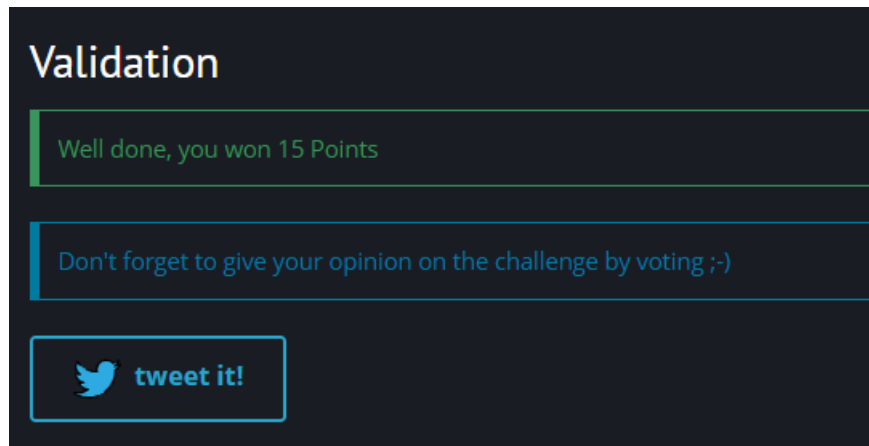
- Sử dụng plugin **imageinfo** để xem thông tin hệ điều hành dump file:

```
(ngthow_kali@kali)-[~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp imageinfo
Volatility Foundation Volatility Framework 2.6.1
INFO : volatility.debug : Determining profile based on KDBG search...
      Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86_24000, Win7SP1x86
      AS Layer1 : IA32PagedMemoryPae (Kernel AS)
      AS Layer2 : FileAddressSpace (/home/ngthow_kali/Downloads/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 : 0xffdf0000L
      Image date and time : 2013-01-12 16:59:18 UTC+0000
      Image local date and time : 2013-01-12 17:59:18 +0100
```

- Tên của máy trạm (computer name) thường được lưu trữ trong registry của hệ thống. Có thể dùng printkey, nhưng để nhanh gọn hơn, em sẽ sử dụng **envvars**, biết tên máy đang sử dụng sẽ được lưu trong biến có tên COMPUTERTNAME, gõ câu lệnh và kết quả sẽ được hiển thị như hình phía dưới:

```
(ngthow_kali@kali)-[~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 envvars | grep COMPUTERTNAME
Volatility Foundation Volatility Framework 2.6.1
560 services.exe 0x001207f0 COMPUTERTNAME WIN-ETSA91RKCFP
576 lsass.exe 0x002507f0 COMPUTERTNAME WIN-ETSA91RKCFP
584 lsm.exe 0x001907f0 COMPUTERTNAME WIN-ETSA91RKCFP
692 svchost.exe 0x002c07f0 COMPUTERTNAME WIN-ETSA91RKCFP
764 svchost.exe 0x002b07f0 COMPUTERTNAME WIN-ETSA91RKCFP
832 svchost.exe 0x003007f0 COMPUTERTNAME WIN-ETSA91RKCFP
904 svchost.exe 0x001407f0 COMPUTERTNAME WIN-ETSA91RKCFP
928 svchost.exe 0x005c07f0 COMPUTERTNAME WIN-ETSA91RKCFP
1084 svchost.exe 0x001307f0 COMPUTERTNAME WIN-ETSA91RKCFP
1172 svchost.exe 0x000b07f0 COMPUTERTNAME WIN-ETSA91RKCFP
1220 AvastSvc.exe 0x005207f0 COMPUTERTNAME WIN-ETSA91RKCFP
1712 spoolsv.exe 0x006707f0 COMPUTERTNAME WIN-ETSA91RKCFP
1748 svchost.exe 0x001707f0 COMPUTERTNAME WIN-ETSA91RKCFP
1968 vmtoolsd.exe 0x002207f0 COMPUTERTNAME WIN-ETSA91RKCFP
```

- Tên máy tìm được là WIN-ETSA91RKCFP. Nhập lên root-me, that's right!



### Level 3:

- 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.

=> Việc cần làm là tìm malware từ file dump, sau đó tính MD5 checksum của đường dẫn đầy đủ của tệp thực thi malware - cũng là cờ cần tìm.

- Một trong những cách để phát hiện malware là xem xét các tiến trình bất thường, vì malware sẽ tạo ra các tiến trình để thực hiện các hành vi độc hại. Sử dụng **pstree** để hiển thị cây tiến trình, xem mối quan hệ giữa các tiến trình cha và tiến trình con:

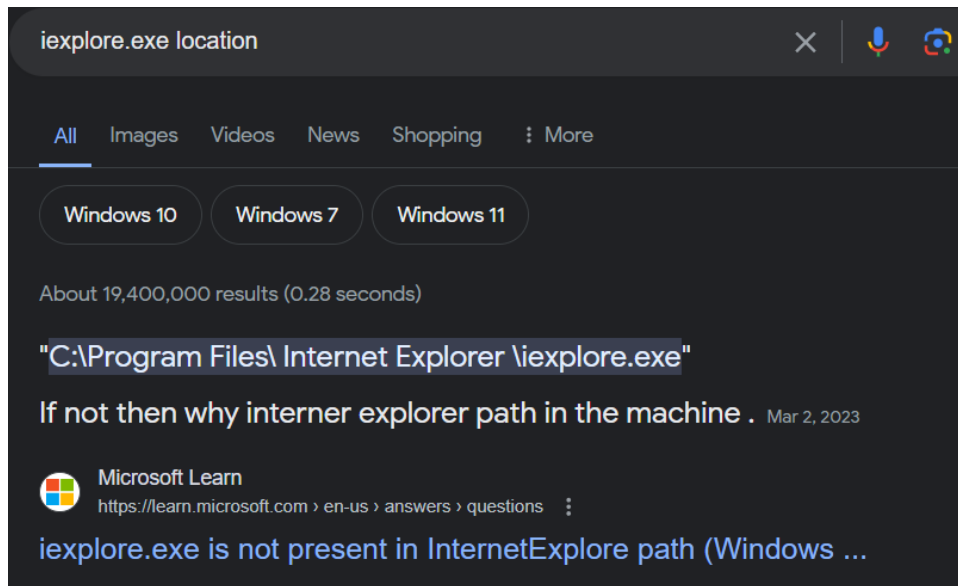
```
(ngthow_kali@kali)~[~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 pstree
Volatility Foundation Volatility Framework 2.6.1
Name                               Pid  PPid  Thds  Hnds  Time
0x892ac2b8:wininit.exe              456   396    3    77  2013-01-12 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:audiodg.exe          1720  832    5   117  2013-01-12 16:58:11 UTC+0000
.. 0x89852918:svchost.exe            904   560   17   409  2013-01-12 16:38:24 UTC+0000
... 0x87ad44d0:dwm.exe              2496  904    5    77  2013-01-12 16:40:25 UTC+0000
.. 0x898b2790:svchost.exe            1172  560   15   475  2013-01-12 16:38:27 UTC+0000
... 0x89f3d2c0:svchost.exe           3352  560    9   141  2013-01-12 16:40:58 UTC+0000
.. 0x898fbb18:SearchIndexer.         2900  560   13   636  2013-01-12 16:40:38 UTC+0000
.. 0x8986b030:svchost.exe            928   560   26   869  2013-01-12 16:38:24 UTC+0000
.. 0x8a1d84e0:vmtoolsd.exe           1968  560    6   220  2013-01-12 16:39:14 UTC+0000
.. 0x8962f030:svchost.exe            692   560   10   353  2013-01-12 16:38:21 UTC+0000
.. 0x898911a8:svchost.exe           1084  560   10   257  2013-01-12 16:38:26 UTC+0000
.. 0x898a7868:AvastSvc.exe           1220  560   66  1180  2013-01-12 16:38:28 UTC+0000
```

- Quan sát kết quả của câu lệnh trên, nhận thấy tiến trình *iexplore.exe* đáng ngờ vì có 1 tiến trình con là *cmd.exe* - là tiến trình dùng để mở cửa sổ dòng lệnh. Trong khi thông thường Internet explore sẽ không có tiến trình con như vậy:

```
. 0x87b6b030:iexplore.exe          2772  2548
.. 0x89898030:cmd.exe              1616  2772
```

- Để xem thông tin về tiến trình này, sử dụng **dlllist -p <PID>**, xuất hiện một đường dẫn. Vì nó dài nó gây ấn tượng nhất nên em bấm nó trước@@. Nhưng nó cũng đáng nghi vì vị trí mà *iexplore.exe* nên được lưu tại *C:\Program Files\Internet Explorer\iexplore.exe* chứ không phải tại đường dẫn như kết quả vừa scan:

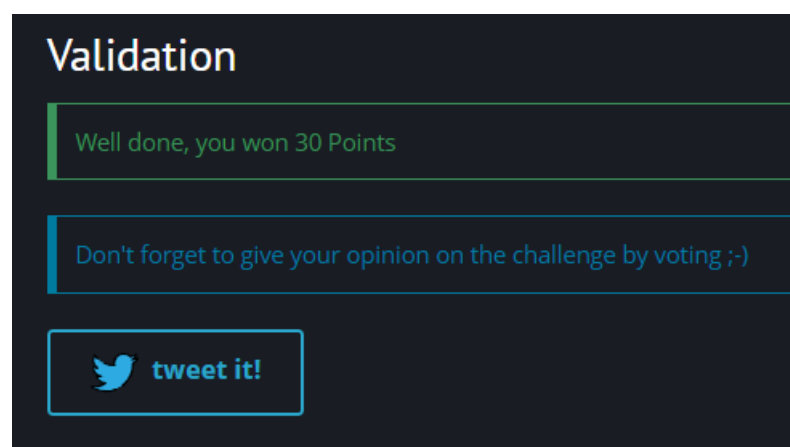




```
(ngthow_kali@kali)~[~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 dlllist -p 2772
Volatility Foundation Volatility Framework 2.6.1
*****
iexplore.exe pid: 2772
Command line : "C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe"

Base          Size      LoadCount LoadTime          Path
0x00400000    0x6000    0xffff 1970-01-01 00:00:00 UTC+0000 C:\Users\John Doe\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch\iexplore.exe
0x77660000    0x13c000  0xffff 1970-01-01 00:00:00 UTC+0000 C:\Windows\SYSTEM32\ntdll.dll
0x70e70000    0x3c000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Program Files\AVAST Software\Avast\snxhk.dll
0x77480000    0xd4000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\KERNEL32.dll
0x75920000    0x4a000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\KERNELBASE.dll
0x76a60000    0xac000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\msvcrt.dll
0x777a0000    0x35000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\WS2_32.DLL
0x76c10000    0xa1000   0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\RPCRT4.dll
0x77880000    0x6000    0xffff 2013-01-12 16:40:34 UTC+0000 C:\Windows\system32\NSI.dll
0x751f0000    0x3c000   0x4    2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\mswsock.dll
0x76990000    0xc9000   0x18   2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\user32.dll
0x75ab0000    0x4e000   0x15   2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\GDI32.dll
0x76980000    0xa000    0x6    2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\LPK.dll
0x777e0000    0x9d000   0x6    2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\USP10.dll
0x75b00000    0x1f000   0x2    2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\IMM32.DLL
0x77210000    0xcc000   0x1    2013-01-12 16:55:34 UTC+0000 C:\Windows\system32\MSCTF.dll
```

- Dem path đi bầm, nhập mã hash `49979149632639432397b3a1df8cb43d`, kết quả thành công:



## Level 4:

- Statement: Berthier, thanks to this new information about the processes running on the workstation, it's clear that this malware is used to exfiltrate data. Find out the ip of the internal server targeted by the hackers!  
The validation flag should have this format : IP:PORT

Nhiệm vụ của task này là tìm ra ip của máy chủ nội bộ bị tin tặc nhắm tới!

- Trong level 3, chúng ta đã xác định được tiến trình độc hại có tiến trình con là cmd.exe với PID 1616. Nên level này sẽ tiếp tục khai thác tại đó. Em sẽ xem lịch sử tiến trình cmd với plugin consoles. Ở đây ta phát hiện PID 1616 là tiến trình được đính kèm vào console (AttachedProcess) của tiến trình PID 2168:

```
*****
ConsoleProcess: conhost.exe Pid: 2168
Console: 0x1081c0 CommandHistorySize: 50
HistoryBufferCount: 3 HistoryBufferMax: 4
OriginalTitle: %SystemRoot%\system32\cmd.exe
Title: C:\Windows\system32\cmd.exe
AttachedProcess: cmd.exe Pid: 1616 Handle: 0x64
-----
CommandHistory: 0x427a60 Application: tcprelay.exe Flags:
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x0
-----
CommandHistory: 0x427890 Application: whoami.exe Flags:
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x0
-----
CommandHistory: 0x427700 Application: cmd.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x64
-----
```

- Từ kết quả trên, lịch sử lệnh gồm 3 ứng dụng, có một ứng dụng tên lạ mà em chưa biết rõ chức năng của nó là *tcprelay.exe*. Do đó, em sẽ sử dụng **memdump** để dump riêng tiến trình 2168 ra file riêng là *2168.dmp*, quét file đó tìm chuỗi *tcprelay.exe*. Kết quả xuất hiện IP và port:

```
(ngthow_kali@kali)-[~/volatility]
$ ls
2168.dmp  2864.dmp  AUTHORS.txt  CHANGELOG.txt  CREDITS.txt  LEGAL.txt


(ngthow_kali@kali)-[~/volatility]
$ strings ./2168.dmp | grep "tcprelay.exe"
tcprelay.exe 192.168.0.22 3389 yourcsecret.co.tv 443
C:\Users\John Doe\AppData\Local\Temp\TEMP23\tcprelay.exeJ"
C:\Users\John Doe\AppData\Local\Temp\TEMP23\tcprelay.exeN_
C:\Users\JOHNDO~1\AppData\Local\Temp\TEMP23\tcprelay.exeg[j
C:\Users\JOHNDO~1\AppData\Local\Temp\TEMP23\tcprelay.exe
C:\Users\JOHNDO~1\AppData\Local\Temp\TEMP23\tcprelay.exe
5C:\Users\JOHNDO~1\AppData\Local\Temp\TEMP23\tcprelay.exeg[j
```

- Tiến hành nộp flag:

## Validation

Well done, you won 35 Points

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

 tweet it!

### Level 5:

- 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.
- Để tìm thông tin về tài khoản người dùng, em sẽ sử dụng **hivelist** để lấy trường địa chỉ bắt đầu trong bộ nhớ của nơi lưu trữ thông tin đăng ký và quản lý tài khoản user:

```
(ngthow_kali@kali) - [~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 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
0x8b23c008 0x02ccf008 \REGISTRY\MACHINE\HARDWARE
0x8ee66008 0x141c0008 \Device\HarddiskVolume1\Boot\BCD
```

- Sau đó hashdump (SYSTEM và SAM):

```
(ngthow_kali@kali) - [~/volatility]
$ python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 hashdump -y 0x8b21c008 -s 0x9aad6148
Volatility Foundation Volatility Framework 2.6.1
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
John Doe:1000:aad3b435b51404eeaad3b435b51404ee:b9f917853e3dbf6e6831ecce60725930:::
```

- Mỗi record lấy ra sẽ có các trường cụ thể ngăn cách nhau bởi dấu ":". Ý nghĩa các trường này là:

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

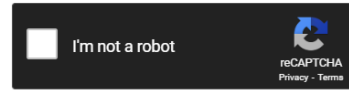
Mật khẩu sẽ được hash và lưu trữ, nên để lấy được password, em sẽ dùng tool CrackStation:



## Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

b9f917853e3dbf6e6831ecce60725930

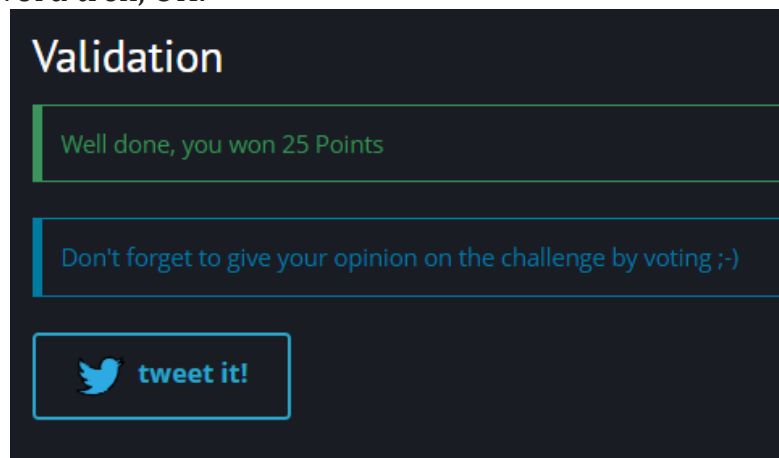


**Supports:** LM, NTLM, md2, md4, md5, md5(md5\_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1 sha1\_bin), QubesV3.1BackupDefaults

Hash	Type	Result
b9f917853e3dbf6e6831ecce60725930	NTLM	password

**Color Codes:** Green Exact match, Yellow Partial match, Red Not found.

- Nộp password trên, OK!



## Level 6:

- 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
- Định nghĩa về C&C: Command and Control 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. C2 thường 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. Nhiệm vụ của level này là tìm được domain dạng hote.domaine.tld.
- Để tìm thông tin domain biết tiến trình malware PID 2772, em sẽ sử dụng **procdump**:

```
(ngthow_kali@kali) - [~/volatility]
$ sudo python2 vol.py -f /home/ngthow_kali/Downloads/ch2.dmp --profile=Win7SP0x86 procdump --dump-dir=./ -p 2772
```















Process(V)	ImageBase	Name	Result
0x87b6b030	0x00400000	iexplore.exe	OK: executable.2772.exe

- Sau khi tạo được file executable.2772.exe, em sử dụng tool VirusTotal để scan tự động tìm các domain. Tìm được khá nhiều:





#### Activity Summary

##### Network Communication ⓘ

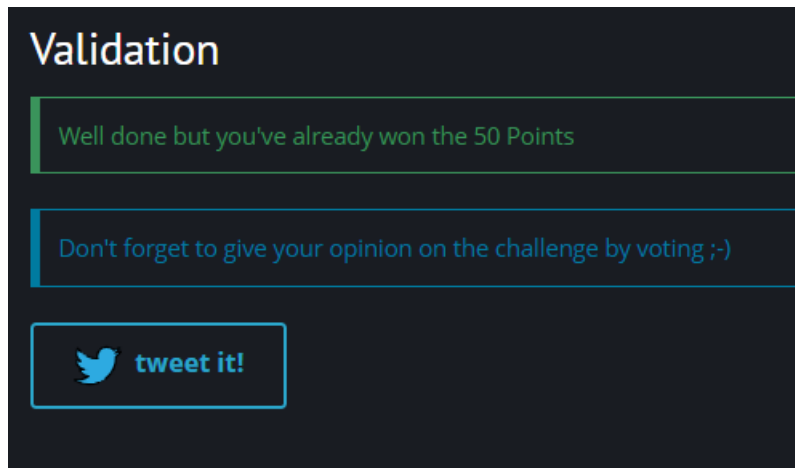
##### DNS Resolutions

- +  108.6.114.104.in-addr.arpa
- +  112.94.48.23.in-addr.arpa
- +  126.207.251.8.in-addr.arpa
- +  138.94.48.23.in-addr.arpa
- +  150.32.88.40.in-addr.arpa
- +  171.45.32.23.in-addr.arpa
- +  185.115.223.3.in-addr.arpa
- +  201.171.66.23.in-addr.arpa
- +  201.198.147.52.in-addr.arpa
- +  201.85.33.23.in-addr.arpa
- +  208.118.189.20.in-addr.arpa
- +  209.205.72.20.in-addr.arpa
- +  3.155.190.20.in-addr.arpa
- +  31.9.204.23.in-addr.arpa

- Tuy nhiên flag có định dạng hote.domaine.tld nên em sẽ lọc lại những domain phù hợp. Do có những domain được chạy trên nhiều môi trường như Cuckoo, Zenbox nên em sẽ ưu tiên thử nó trước:

- +  furious.devilslife.com
- +  ns2.wrauzfevvo.com
- +  th1sis.l1k3aK3y.org
- +  y0ug.itisjustluck.com

- Sau khi thử lần lượt thì đến domain *th1sis.l1k3aK3y.org* là đúng:



### 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.
  - Trước khi có thể bắt đầu phân tích, chúng ta cần cho Volatility biết chúng ta đang làm việc với loại hình ảnh bộ nhớ nào. Plugin imageinfo sẽ quét hình ảnh và đề xuất một số hồ sơ có khả năng.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem 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, Win7SP1x64_24000
23418
AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
AS Layer2 : FileAddressSpace (/home/hahien/Downloads/Kb05-dp-E81.vmem)
PAE type : No PAE
DTB : 0x187000L
KDBG : 0xf80002c430a0L
Devices : Number of Processors : 2
Image Type (Service Pack) : 1
KPCR for CPU 0 : 0xfffff80002c44d00L
KPCR for CPU 1 : 0xfffff80009ef000L
KUSER_SHARED_DATA : 0xfffff78000000000L
Image date and time : 2018-08-04 19:34:22 UTC+0000
Image local date and time : 2018-08-04 22:34:22 +0300
```

- Plugin hashdump sẽ kết xuất các giá trị băm

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" hashdump
Volatility Foundation Volatility Framework 2.6.1
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Rick:1000:aad3b435b51404eeaad3b435b51404ee:518172d012f97d3a8fcc089615283940:::
```

- Câu hỏi yêu cầu mật khẩu người dùng, không phải mật khẩu băm, vì vậy chúng ta có thể thử bẻ khóa bằng các công cụ như John the Ripper hoặc Hashcat (hoặc Google) hoặc chúng ta có thể thử trích xuất mật khẩu văn bản gốc từ bí mật LSA bằng cách sử dụng plugin lsadump .

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" lsadump
Volatility Foundation Volatility Framework 2.6.1
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.....
```

→ CTF{MortyIsReallyAnOtter}

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

- Plugin Netscan sẽ cung cấp Plugin Netscan sẽ cung cấp network data.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" netscan
Volatility Foundation Volatility Framework 2.6.1
Offset(P) Proto Local Address Foreign Address State Pid Owner Created
0x7d60f010 UDPv4 0.0.0.0:1900 *:* 2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d62b3f0 UDPv4 192.168.202.131:6771 *:* 2836 BitTorrent.exe 2018-08-04 19:27:22 UTC+0000
0x7d62f4c0 UDPv4 127.0.0.1:62307 *:* 2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d62f920 UDPv4 192.168.202.131:62306 *:* 2836 BitTorrent.exe 2018-08-04 19:27:17 UTC+0000
0x7d6424c0 UDPv4 0.0.0.0:50762 *:* 4076 chrome.exe 2018-08-04 19:33:37 UTC+0000
0x7d6b4250 UDPv6 ::1:1900 *:* 164 svchost.exe 2018-08-04 19:28:42 UTC+0000
0x7d6e3230 UDPv4 127.0.0.1:6771 *:* 2836 BitTorrent.exe 2018-08-04 19:27:22 UTC+0000
0x7d6ed650 UDPv4 0.0.0.0:5355 *:* 620 svchost.exe 2018-08-04 19:34:22 UTC+0000
0x7d71c8a0 UDPv4 0.0.0.0:0 *:* 868 svchost.exe 2018-08-04 19:34:22 UTC+0000
```

→ Chúng ta có thể loại trừ 0.0.0.0 và 127.0.0.1, vậy IP cần tìm là 192.168.202.131

→ CTF{192.168.202.131}

- Hostname được lưu trữ trong SYSTEM register hive. Trước khi có thể truy vấn chúng ta cần tìm offset.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" hivelist
Volatility Foundation Volatility Framework 2.6.1
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 0x00000000b92b010 \??\C:\Users\Rick\ntuser.dat
0xfffff8a0020ad410 0x00000000db41410 \??\C:\Users\Rick\AppData\Local\Microsoft\Windows\UsrClass.dat
```

- Việc cung cấp plugin printkey cùng với offset và tên của khóa đăng ký có liên quan sẽ mang lại cho chúng ta flag 2 cho câu hỏi này.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" printkey -o 0xffff
ff8a000024010 -K "ControlSet001\Control\ComputerName\ComputerName"
Volatility Foundation Volatility Framework 2.6.1
Legend: (S) = Stable (V) = Volatile

Registry: \REGISTRY\MACHINE\SYSTEM
Key name: ComputerName (S)
Last updated: 2018-06-02 19:23:00 UTC+0000

Subkeys:

Values:
REG_SZ          ComputerName      : (S) mnmsrvc
REG_SZ          ComputerName      : (S) WIN-LO6FAF3DTFE

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

→ CTF{WIN-LO6FAF3DTFE}

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

- Plugin pstree cung cấp cho chúng ta cái nhìn rõ ràng về các tiến trình đang chạy.

```
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" pstree
Volatility Foundation Volatility Framework 2.6.1
```

Name	Pid	PPid	Thds	Hnds	Time
0xfffffa801b27e060:explorer.exe	2728	2696	33	854	2018-08-04 19:27:04 UTC
+0000					
. 0xfffffa801b486b30:Rick And Morty	3820	2728	4	185	2018-08-04 19:32:55 UTC
+0000					
.. 0xfffffa801a4c5b30:vmware-tray.exe	3720	3820	8	147	2018-08-04 19:33:02 UTC
+0000					
. 0xfffffa801b2f02e0:WebCompanion.e	2844	2728	0	—	2018-08-04 19:27:07 UTC
+0000					
. 0xfffffa801a4e3870:chrome.exe	4076	2728	44	1160	2018-08-04 19:29:30 UTC
+0000					
.. 0xfffffa801a4eab30:chrome.exe	4084	4076	8	86	2018-08-04 19:29:30 UTC
+0000					
.. 0xfffffa801a5ef1f0:chrome.exe	1796	4076	15	170	2018-08-04 19:33:41 UTC
+0000					
.. 0xfffffa801aa00a90:chrome.exe	3924	4076	16	228	2018-08-04 19:29:51 UTC
+0000					
.. 0xfffffa801a635240:chrome.exe	3648	4076	16	207	2018-08-04 19:33:38 UTC
+0000					
.. 0xfffffa801a502b30:chrome.exe	576	4076	2	58	2018-08-04 19:29:31 UTC
+0000					
.. 0xfffffa801a4f7b30:chrome.exe	1808	4076	13	229	2018-08-04 19:29:32 UTC
+0000					
.. 0xfffffa801a7f98f0:chrome.exe	2748	4076	15	181	2018-08-04 19:31:15 UTC
+0000					
. 0xfffffa801b5cb740:LunarMS.exe	708	2728	18	346	2018-08-04 19:27:39 UTC
+0000					
. 0xfffffa801b1cdb30:vmtoolsd.exe	2804	2728	6	190	2018-08-04 19:27:06 UTC

- Google cho ta biết **LunarMS** được liên kết với một game MMORPG cũ, vì vậy đây là flag 1.

→ CTF{LunarMS}

- Việc tìm IP của máy chủ chỉ đơn giản là chạy plugin Netscan và sử dụng grep để lọc trên quy trình LunarMS.



```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" netscan | grep "LunarMS"
Volatility Foundation Volatility Framework 2.6.1
0x7d6124d0 TCPv4 192.168.202.131:49530 77.102.199.102:7575 CLOSED 708 LunarMS.exe
0x7e413a40 TCPv4 --:0 --:0 CLOSED 708 LunarMS.exe
0x7e521b50 TCPv4 --:0 --:0 CLOSED 708 LunarMS.exe

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

→ CTF{77.102.199.102}

- 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.

- Tên tài khoản sẽ ở đâu đó trong bộ nhớ tiến trình. Chúng ta biết PID của quy trình LunarMS là 708, vì vậy hãy chuyển giá trị đó đến plugin memdump, sau đó sử dụng chuỗi và grep để lọc đầu ra. -C 10 báo cho grep trả về 10 dòng trên và dưới dòng khớp.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" memdump -p 708 -D .
Volatility Foundation Volatility Framework 2.6.1
*****
Writing LunarMS.exe [ 708] to 708.dmp
```

```
(root@kali)-[~/volatility]
# strings 708.dmp > 708.dmp.strings

(root@kali)-[~/volatility]
# grep -C 10 "Lunar-3" 708.dmp.strings
{qv1 hahien
b+Y,
,b+Y Desktop
b+YD Recent
Db+Y Trash
c+Y\ Documents
tb+Y4c+Y
b+YLc+Y
Lunar-3
Lunar-4
L(dNVxdNV
L|eNV
{qf8
$m1Y
4v+Y
TI,Y
lx+Y
ty+Y
,y+Y\y+Y
--
magician
bowman
thief
```

```

thief
pirate
Sound/ash
normal
pressed
disabled
mouseOver
keyFocused
Lunar-3
0tt3r8r33z3
Sound/UI.img/
BtMouseClicked
Lunar-4
Lunar-1
Lunar-2
ScrollUp
Title
RollDown
WorldSelect

(root@kali)-[~/volatility]
#

```

→ CTF{0tt3r8r33z3}

- 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.

- Plugin clipboard có thể cung cấp cho chúng ta những gì chúng ta cần:

```

(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" clipboard
Volatility Foundation Volatility Framework 2.6.1

```

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	

→ CTF{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?

- Liệt kê các tiến trình với pstree, chúng ta có thể thấy một tiến trình tên là Rick và Morty, với một tiến trình con tên là vmware-tray.ex khả nghi.

```
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" pstree
Volatility Foundation Volatility Framework 2.6.1
```

Name	Pid	PPid	Thds	Hnds	Time
0xfffffa801b27e060:explorer.exe	2728	2696	33	854	2018-08-04 19:27:04 UTC+0000
0xfffffa801b486b30:Rick And Morty	3820	2728	4	185	2018-08-04 19:32:55 UTC+0000
0xfffffa801a4c5b30:vmware-tray.exe	3720	3820	8	147	2018-08-04 19:33:02 UTC+0000
0xfffffa801b2f02e0:WebCompanion.e	2844	2728	0	—	2018-08-04 19:27:07 UTC+0000
0xfffffa801a4e3870:chrome.exe	4076	2728	44	1160	2018-08-04 19:29:30 UTC+0000
0xfffffa801a4eab30:chrome.exe	4084	4076	8	86	2018-08-04 19:29:30 UTC+0000
0xfffffa801a5ef1f0:chrome.exe	1796	4076	15	170	2018-08-04 19:33:41 UTC+0000
0xfffffa801aa00a90:chrome.exe	3924	4076	16	228	2018-08-04 19:29:51 UTC+0000
0xfffffa801a635240:chrome.exe	3648	4076	16	207	2018-08-04 19:33:38 UTC+0000
0xfffffa801a502b30:chrome.exe	576	4076	2	58	2018-08-04 19:29:31 UTC+0000
0xfffffa801a4f7b30:chrome.exe	1808	4076	13	229	2018-08-04 19:29:32 UTC+0000
0xfffffa801a7f98f0:chrome.exe	2748	4076	15	181	2018-08-04 19:31:15 UTC+0000
0xfffffa801b5cb740:LunarMS.exe	708	2728	18	346	2018-08-04 19:27:39 UTC+0000
0xfffffa801b1cddb30:vmtoolsd.exe	2804	2728	6	190	2018-08-04 19:27:06 UTC+0000
0xfffffa801b290b30:BitTorrent.exe	2836	2728	24	471	2018-08-04 19:27:07 UTC+0000
0xfffffa801b4c9b30:bittorrentie.e	2624	2836	13	316	2018-08-04 19:27:21 UTC+0000
0xfffffa801b4a7b30:bittorrentie.e	2308	2836	15	337	2018-08-04 19:27:19 UTC+0000
0xfffffa8018d44740:System	4	0	95	411	2018-08-04 19:26:03 UTC+0000
0xfffffa801947e4d0:smss.exe	260	4	2	30	2018-08-04 19:26:03 UTC+0000
0xfffffa801a2ed060:wininit.exe	396	336	3	78	2018-08-04 19:26:11 UTC+0000
0xfffffa801ab377c0:services.exe	492	396	11	242	2018-08-04 19:26:12 UTC+0000
0xfffffa801afe7800:svchost.exe	1948	492	6	96	2018-08-04 19:26:42 UTC+0000
0xfffffa801ae92920:vmtoolsd.exe	1428	492	9	313	2018-08-04 19:26:27 UTC+0000
0xfffffa801a572b30:cmd.exe	3916	1428	0	—	2018-08-04 19:34:22 UTC+0000
0xfffffa801ae0f630:VGAAuthService.	1356	492	3	85	2018-08-04 19:26:25 UTC+0000
0xfffffa801abbdb30:vmacthlp.exe	668	492	3	56	2018-08-04 19:26:16 UTC+0000
0xfffffa801aad1060:Lavasoft.WCAss	3496	492	14	473	2018-08-04 19:33:49 UTC+0000

- Bằng cách cung cấp PID cho plugin cmdline, chúng ta có thể thấy các dòng lệnh đầy đủ được liên kết với cả hai quy trình bất thường này.

```
(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" cmdline -p 3820,3720
Volatility Foundation Volatility Framework 2.6.1
*****
Rick And Morty pid: 3820
Command line : "C:\Torrents\Rick And Morty season 1 download.exe"
*****
vmware-tray.exe pid: 3720
Command line : "C:\Users\Rick\AppData\Local\Temp\RarSFX0\vmware-tray.exe"
```

→ Mã độc dưới dạng file .exe, CTF{vmware-tray.exe}

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

- Sử dụng plugin filescan và lọc bằng grep sẽ cung cấp cho chúng ta một vài nơi để xem xét.

```
(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" filescan | grep -i "rick and morty"
Volatility Foundation Volatility Framework 2.6.1
0x000000007d63dbc0 10 0 R-r-d \Device\HarddiskVolume1\Torrents\Rick And Morty season 1 download.exe
0x000000007d6b3a10 11 1 R-rw- \Device\HarddiskVolume1\Torrents\Rick and Morty - Season 3 (2017) [1080p]\Rick.and.Morty.S03E07.The.Ricklantis.Mixup.1080p.Ama
on.WEB-DL.x264-Rapta.mkv
0x000000007d7adb50 17 1 R-rw- \Device\HarddiskVolume1\Torrents\Rick and Morty - Season 3 (2017) [1080p]\Rick.and.Morty.S03E06.Rest.and.Ricklaxation.1080p.Ama
zon.WEB-DL.x264-Rapta.mkv
0x000000007d8813c0 2 0 RW-rwd \Device\HarddiskVolume1\Users\Rick\Downloads\Rick And Morty season 1 download.exe.torrent
0x000000007da56240 2 0 RW-rwd \Device\HarddiskVolume1\Torrents\Rick And Morty season 1 download.exe
0x000000007dae9350 2 0 RWD— \Device\HarddiskVolume1\Users\Rick\AppData\Roaming\BitTorrent\Rick And Morty season 1 download.exe.1.torrent
0x000000007dcbf6f0 2 0 RW-rwd \Device\HarddiskVolume1\Users\Rick\AppData\Roaming\BitTorrent\Rick And Morty season 1 download.exe.1.torrent
0x000000007e5f5d10 3 1 R-rw- \Device\HarddiskVolume1\Torrents\Rick and Morty Season 2 [WEBRIP] [1080p] [HEVC]\[pseudo] Rick and Morty S02E03 Auto Erotic Ass
imilation [1080p] [h.265].mkv
0x000000007e710070 8 0 R-rwd \Device\HarddiskVolume1\Torrents\Rick And Morty season 1 download.exe
0x000000007e7ae700 3 1 R-rw- \Device\HarddiskVolume1\Torrents\Rick and Morty Season 2 [WEBRIP] [1080p] [HEVC]\Sample\Screenshot 08.png
```

- Chúng ta có thể trích xuất các tệp từ hình ảnh bộ nhớ bằng cách chuyển o vào plugin dumpfiles.

- Sử dụng cat để hiển thị nội dung của file, chúng ta thấy rằng đó là Mã định danh vùng chứ không phải chính torrent. Dòng ZoneId=3 chỉ ra rằng torrent đã được tải xuống từ internet

```
(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" dumpfiles -Q 0x00000007d8813c0 -D .
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0x7d8813c0 None \Device\HarddiskVolume1\Users\Rick\Downloads\Rick And Morty season 1 download.exe.torrent

(root@kali)~[~/volatility]
# cat file.None.0xfffffa801af10010.dat
[ZoneTransfer]
ZoneId=3

(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" dumpfiles -Q 0x00000007dae9350 -D .
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0x7dae9350 None \Device\HarddiskVolume1\Users\Rick\AppData\Roaming\BitTorrent\Rick And Morty season 1 download.exe.1.torrent
```

- Chạy chuỗi lần này, chúng ta có thể xem chi tiết về torrent, bao gồm cả nhận xét ở dòng cuối cùng là FLAG

```
(root@kali)~[~/volatility]
# strings file.None.0xfffffa801b42c9e0.dat
d8:announce44:udp://tracker.openbittorrent.com:80/announce13:announce-list144:udp://tracker.openbittorrent.com:80/announceel42:udp://tracker.opentracker.org:1337/annou
nceee10:created by17:BitTorrent/7.10.313:creation date1533150595e8:encoding5:UTF-84:infod6:length1456670e4:name36:Rick And Morty season 1 download.exe12:piece length1
16384e6:pieces560:I
!PC<^X
B.k_Rk
0<;087o
!4^"
3hq,
6iW1l
K68:o
w-Q~YT
$$o9p
bwF:u
e7:website19:M3an_T0rren7_4_R!cke
```

- 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?

```
(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" filescan | grep -ie "history$"
Volatility Foundation Volatility Framework 2.6.1
0x00000007d45dcc0 18 1 RW-rw- \Device\HarddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History

(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" dumpfiles -Q 0x00000007d45dcc0 -D .
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0x7d45dcc0 None \Device\HarddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History
SharedCacheMap 0x7d45dcc0 None \Device\HarddiskVolume1\Users\Rick\AppData\Local\Google\Chrome\User Data\Default\History

(root@kali)~[~/volatility]
# strings /home/hahien/Downloads/Kb05-dp-E81.vmem > /home/hahien/Downloads/Kb05-dp-E81.vmem.strings

(root@kali)~[~/volatility]
# grep "@mail.com" /home/hahien/Downloads/Kb05-dp-E81.vmem.strings
J{"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"}}0
n"rickopicko@mail.com" <rickopicko@mail.com>
usernamerickypinky@mail.comrickypinky@mail.com[
usernamerickopicko@mail.comrickopicko@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.comrickypinky@mail.com[
usernamerickopicko@mail.comrickopicko@mail.com[
usernamerickypinky@mail.com
usernamerickopicko@mail.com
RickoPicko@mail.com
usernamerickopicko@mail.comrickopicko@mail.com[
usernamerickypinky@mail.com
usernamerickopicko@mail.com
```



- Dòng thứ hai của đầu ra grep giống với trường địa chỉ của tiêu đề email; có lẽ nội dung tin nhắn nào đó vẫn còn trong bộ nhớ khi hình ảnh được tạo. Sử dụng grep với cờ -A 20 để hiển thị 20 dòng theo địa chỉ email của Rick sẽ cho chúng ta những thông tin sau:

```
(root@kali)-[~/volatility]
# grep -A 20 "<rickopicko@mail.com>" /home/hahien/Downloads/Kb05-dp-E81.vmem.strings
n"rickopicko@mail.com" "<rickopicko@mail.com>"
button transparent normal closeconfirmboxsm
jSpecial Offer: 20% off your first order!jss
jhttps://sb.scorecardresearch.com/beacon.js'
digitalmars-d-announce-request@puremagic.com
font-family: Verdana;font-size: 12.0px;.png
JLAST CHANCE: 20% off your first order.com
navigation-collapse toggle-resolution.comsQ=
M8.81 5h2.4l-.18 7H8.98l-.17-7zM9 14h2v2H9z=
simple-icon_mail-classification-feedbackKw=
form-composite-switchable-content_condition
form-composite-addresschooser_textfieldc.com
SPnvideo-label video-title trc_ellipsis ]"sAE=
display:inline;width:56px;height:200px;m>
Hum@n_I5_Th3_Weak3s7_Link_In_Th3_Ch@inYear
//sec-s.uicdn.com/nav-cdn/home/preloader.gif
simple-icon_toolbar-change-view-horizontal
nnx-track-sec-click-communication-inboxic.com
nx-track-sec-click-dashboard-hide_smileyable
Nftd-box stem-north big fullsize js-focusable
js-box-flex need-overlay js-componenttone
```

→ CTF{Hum@n\_I5\_Th3\_Weak3s7\_Link\_In\_Th3\_Ch@in}

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

- Câu hỏi cho chúng ta biết rằng phần mềm độc hại là một loại ransomware nào đó và yêu cầu địa chỉ Bitcoin được liên kết. Ransomware có xu hướng để lại thông báo đòi tiền chuộc trên Desktop, vì vậy hãy tìm kiếm thông báo đó trước.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" filescan | grep "Desktop"
Volatility Foundation Volatility Framework 2.6.1
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
```

- READ\_IT.txt và flag.txt có thể hữu ích.

```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" dumpfiles -Q 0x000000007d660500 -D .
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0x7d660500 None \Device\HarddiskVolume1\Users\Rick\Desktop\READ_IT.txt

(root@kali)-[~/volatility]
# cat file.None.0xfffffa801b2def10.dat
Your files have been encrypted.
Read the Program for more information
read program for more information.
```

- Rất tiếc, ghi chú chỉ yêu cầu chúng ta đọc chương trình để biết thêm thông tin. Chúng ta đã xác định được ransomware PID ở câu hỏi trước đó, vì vậy ta xuất bộ nhớ tiến trình và chạy chuỗi và grep để tìm kiếm bất kỳ đề cập nào đến "ransom". Cờ -e l được sử dụng để tìm kiếm chuỗi Unicode.



```
(root@kali)-[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" memdump -p 3720 -D .
Volatility Foundation Volatility Framework 2.6.1
*****
Writing vmware-tray.ex [ 3720] to 3720.dmp

(root@kali)-[~/volatility]
# strings -e l 3720.dmp | grep -i -A 5 "ransom"
This is Ransomware. It locks your files until you pay for them. Before you ask, Yes we will
give you your files back once you pay and our server confrim that you pay.
System.Drawing, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a
@PPP
0000
0000
--
This is Ransomware. It locks your files until you pay for them. Before you ask, Yes we will
give you your files back once you pay and our server confrim that you pay.
Send 0.16 to the address below.
e al
I paid, Now give me back my files.
1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M
--
This is Ransomware. It locks your files until you pay for them. Before you ask, Yes we will
give you your files back once you pay and our server confrim that you pay.
Your Files are locked. They are locked because you downloaded something with this file in it.
This is Ransomware. It locks your files until you pay for them. Before you ask, Yes we will
give you your files back once you pay and our server confrim that you pay.
\\.\DISPLAY1
\\.\DISPLAY1
\\.\DISPLAY1
\\.\DISPLAY1
--
This is Ransomware. It locks your files until you pay for them. Before you ask, Yes we will
give you your files back once you pay and our server confrim that you pay.
Send 0.16 to the address below.
e al
I paid, Now give me back my files.
1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M
```

→ CTF{1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M}

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

```
(root@kali)-[~/volatility]
# strings -e l 3720.dmp > 3720.dmp.strings

(root@kali)-[~/volatility]
# wc -l 3720.dmp.strings
399214 3720.dmp.strings

(root@kali)-[~/volatility]
# wc -l 3720.dmp.strings
399214 3720.dmp.strings

Network
(root@kali)-[~/volatility]
# grep "WIN-LO6FAF3DTFE" 3720.dmp.strings | wc -l
658
```

```
(root@kali)~[~/volatility]
# grep "WIN-L06FAF3DTFE" 3720.dmp.strings | sort | uniq
80000171WIN-L06FAF3DTFE
-AdministratorWIN-L06FAF3DTFE
\BaseNamedObjects\Global\WIN-L06FAF3DTFE
computername=WIN-L06FAF3DTFE
COMPUTERNAME=WIN-L06FAF3DTFE
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe (WIN-L06FAF3DTFE)
\Device\NetbiosSmbWIN-L06FAF3DTFE\WORKGROUP
\Device\NetBT_Tcpip_{7F5B9219-B869-4AEA-84AF-CC6E4C2486FA}\WIN-L06FAF3DTFE\WORKGROUP
-GuestWIN-L06FAF3DTFE
Logoff PolicyWIN-L06FAF3DTFE
logonserver=\\WIN-L06FAF3DTFE
LOGONSERVER=\\WIN-L06FAF3DTFE
NoneWIN-L06FAF3DTFE
Password PolicyWIN-L06FAF3DTFE
-RickWIN-L06FAF3DTFE
RickWIN-L06FAF3DTFE
User32 NegotiateWIN-L06FAF3DTFE
userdomain=WIN-L06FAF3DTFE
USERDOMAIN=WIN-L06FAF3DTFE
USERNAME=WIN-L06FAF3DTFE$
\\WIN-L06FAF3DTFE
WIN-L06FAF3DTFE
WIN-L06FAF3DTFE
WIN-L06FAF3DTFE$
WIN-L06FAF3DTFE$WORKGROUP
WIN-L06FAF3DTFE
WIN-L06FAF3DTFE\Rick
WIN-L06FAF3DTFE-Rick aDOBofVYUNVnmp7
WORKGROUP\WIN-L06FAF3DTFE$
```

- Dòng cuối cùng thứ hai có vẻ thú vị; tên máy chủ và tên người dùng được nối với nhau bằng một chuỗi chữ và số dường như ngẫu nhiên.
- Sử dụng grep chúng ta thấy rằng chuỗi dường như ngẫu nhiên này xuất hiện nhiều lần nên đây có thể là password.

```
(root@kali)~[~/volatility]
# grep -C 5 "aDOBofVYUNVnmp7" 3720.dmp.strings
2bf8098_r15_ad1
2bf8098_r15_ad1
WindowsForms10.STATIC.app.0.2bf8098_r15_ad1
\Desktop\
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890*!@#%&/
aDOBofVYUNVnmp7
aDOBofVYUNVnmp7
C:\Users\Rick\Desktop\
WIN-L06FAF3DTFE-Rick aDOBofVYUNVnmp7
.txt
.doc
.docx
.xls
.xlsx
mode
access
rights
share
C:\Users\Rick\Desktop\Flag.txt
aDOBofVYUNVnmp7
count
charIndex
System.Security.Cryptography.SHA256
cryptoNameMapping
oidMap
```

→ CTF{aDOBofVYUNVnmp7}

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

- Bây giờ chúng ta đã xác định được ransomware và tìm được password, chúng ta hãy giải nén file chứa cờ cuối cùng. Trong câu trước, chúng ta thấy một tệp có tên Flag.txt.

```
(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" filescan | grep "Flag.txt$"
Volatility Foundation Volatility Framework 2.6.1
0x000000007e410890 16 0 R--r-- \Device\HarddiskVolume1\Users\Rick\Desktop\Flag.txt

(root@kali)~[~/volatility]
# python2 vol.py -f /home/hahien/Downloads/Kb05-dp-E81.vmem --profile="Win7SP1x64" dumpfiles -Q 0x000000007e410890 -D .
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0x7e410890 None \Device\HarddiskVolume1\Users\Rick\Desktop\Flag.txt

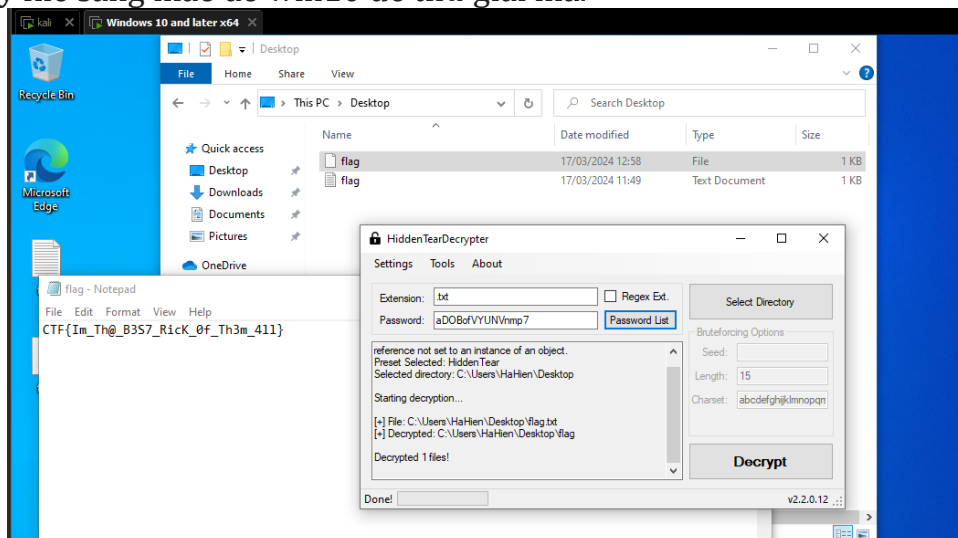
(root@kali)~[~/volatility]
# xxd file.None.0xfffffa801b0532e0.dat
00000000: 7be6 2456 9e5c 0fef 8e43 28f7 e4c5 83ff {.$V.\ ... C(....
00000010: 6c31 d7e6 1cda ea54 cf72 ddd6 ec7e b07b l1....T.r...~.{
00000020: c68d d0a8 ccc2 ce6e 3eee 0347 c10b b3e8 .....n>..G....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
```

- Sau khi trích xuất tệp từ bộ nhớ, chúng ta có thể kiểm tra nó bằng xxd, hiển thị một khối gồm 48 byte dường như ngẫu nhiên, theo sau là phần đệm byte rỗng.
- Phần đệm có thể gây ra sự cố khi giải mã, vì vậy chúng ta phải trích xuất byte mà chúng ta muốn vào một file mới có tên flag.txt bằng cách sử dụng dd.

```
(root@kali)~[~/volatility]
# dd bs=1 count=48 if=file.None.0xfffffa801b0532e0.dat of=flag.txt
48+0 records in
48+0 records out
48 bytes copied, 0.000189549 s, 253 kB/s

(root@kali)~[~/volatility]
# xxd flag.txt
00000000: 7be6 2456 9e5c 0fef 8e43 28f7 e4c5 83ff {.$V.\ ... C(....
00000010: 6c31 d7e6 1cda ea54 cf72 ddd6 ec7e b07b l1....T.r...~.{
00000020: c68d d0a8 ccc2 ce6e 3eee 0347 c10b b3e8 .....n>..G....
```

- Copy file sang máy ảo win10 để thử giải mã.



→ CTF{Im\_Th@\_B3S7\_RicK\_Of\_Th3m\_4ll}

---

*Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này*

## YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

### Báo cáo:

- File **.DOCX và .PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach) – cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-ExeX\_GroupY. (trong đó X là Thứ tự Bài tập, Y là mã số thứ tự nhóm trong danh sách mà GV phụ trách công bố).  
*Ví dụ: [NT101.K11.ANTT]-Exe01\_Group03.*
- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài nộp.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại [courses.uit.edu.vn](https://courses.uit.edu.vn).

### Đánh giá:

- Hoàn thành tốt yêu cầu được giao.
- Có nội dung mở rộng, ứng dụng.

*Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.*

**HẾT**