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.O21.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 |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-1)**

|  |  |  |
| --- | --- | --- |
| **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% |

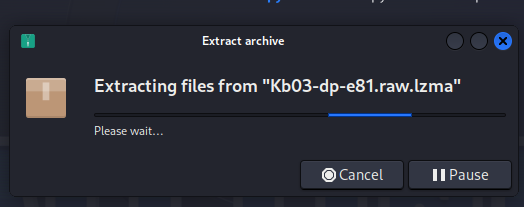
**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

BÁO CÁO CHI TIẾT

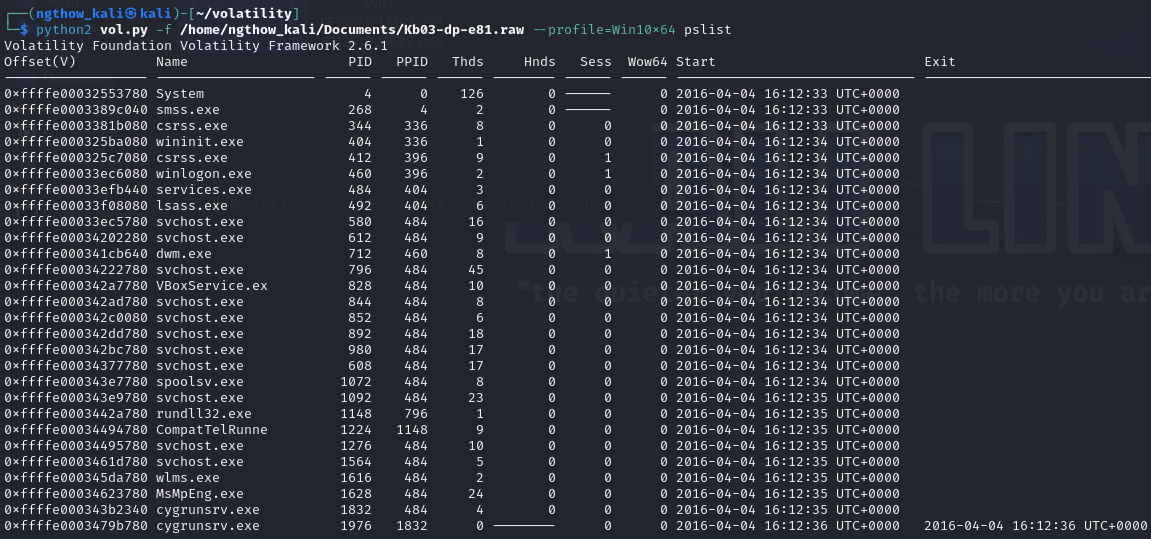
**Tài nguyên: Kb03-dp-e81.raw.lzma**

**Yêu cầu 3. Thực hiện phân tích:**

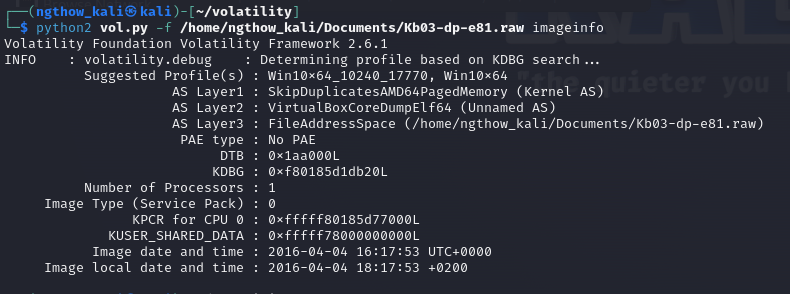
* Cung cấp bằng chứng xác định file được cho là file dump từ bộ nhớ máy ảo. Xác định hệ điều hành của máy này.
* Một tệp có phần mở rộng tệp LZMA là tệp nén, do đó cần phải giải nén file trước khi phân tích:



* 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:



* 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**.



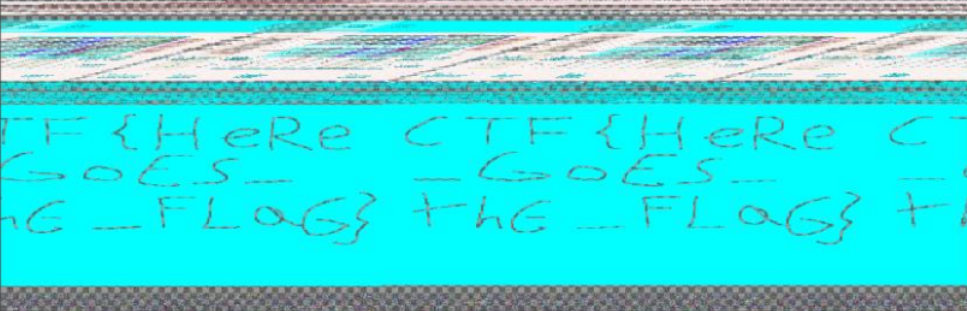
* Em dùng ***strings*** để tìm chuỗi, tuy nhiên không có kết quả:



* Có thể flag ở dạng khác ví dụ như hình ảnh. Do đó em sẽ dùng ***pslist*** để xem lịch sử tiến trình, xem thử có chạy ứng dụng nào liên quan đến hình ảnh không. Thì xuất hiện 1 tiến trình với PID 4092 *mspaint.exe*:



* Tìm hiểu sâu hơn về tiến trình này bằng ***memdump*** được file 4092.dmp.
* Đổi đuôi file .dmp thành .data để có thể đọc được nội dung, flag cần tìm là:



=> CTF{HeRe\_GoEs\_thE\_FLaG}

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

- Thực hiện các bước điều tra, mô tả rõ ràng.

- Có ảnh chụp, giải thích lí do.

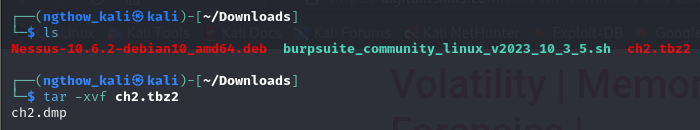
**Level 2:**

* Statement của yêu cầu này như sau:

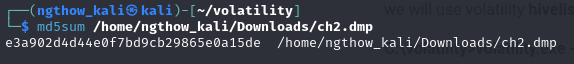
Congratulations Berthier, thanks to your help the computer has been identified. You have requested a memory dump but before starting your analysis you wanted to take a look at the antivirus’ logs. Unfortunately, you forgot to write down the workstation’s hostname. But since you have its memory dump you should be able to get it back! The validation flag is the workstation’s hostname. The uncompressed memory dump md5 hash is e3a902d4d44e0f7bd9cb29865e0a15de

Vậy flag cần tìm là tên của máy trạm.

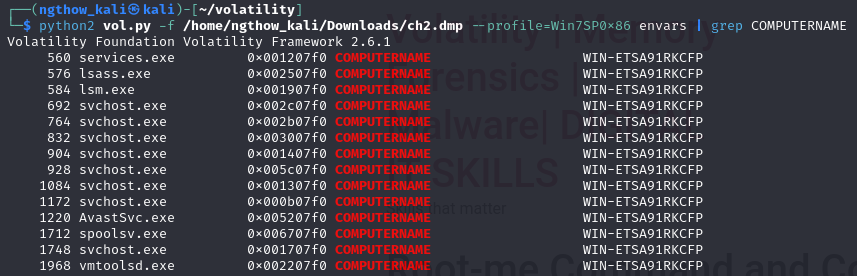
* Tải tệp về và giải nén:



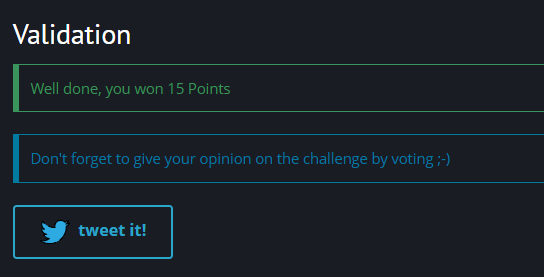
* Challenge có cung cấp mã hash của file để chúng ta kiểm tra liệu file tải về có đúng chưa, nên em sẽ băm file em tải, đối chiếu với giá trị băm web cung cấp. Kết quả trùng nhau, nên OK sẽ tiến hành phân tích file tìm flag!



* Sử dụng plugin ***imageinfo*** để xem thông tin hệ điều hành dump file:
* 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 ***envars***, biết tên máy đang sử dụng sẽ được lưu trọng biến có tên COMPUTERNAME, gõ câu lệnh và kết quả sẽ được hiển thị như hình phía dưới:



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

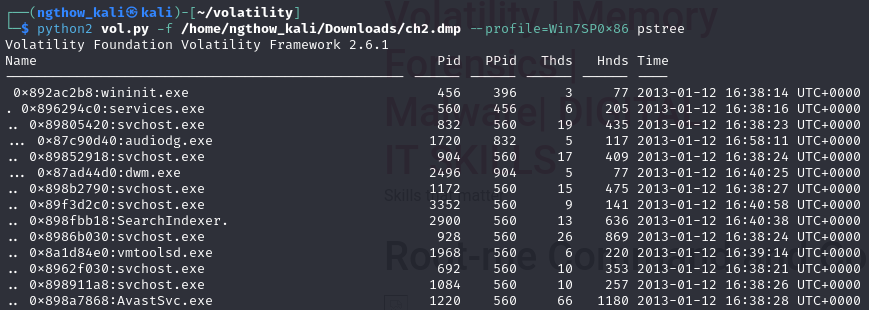


**Level 3:**

* 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.

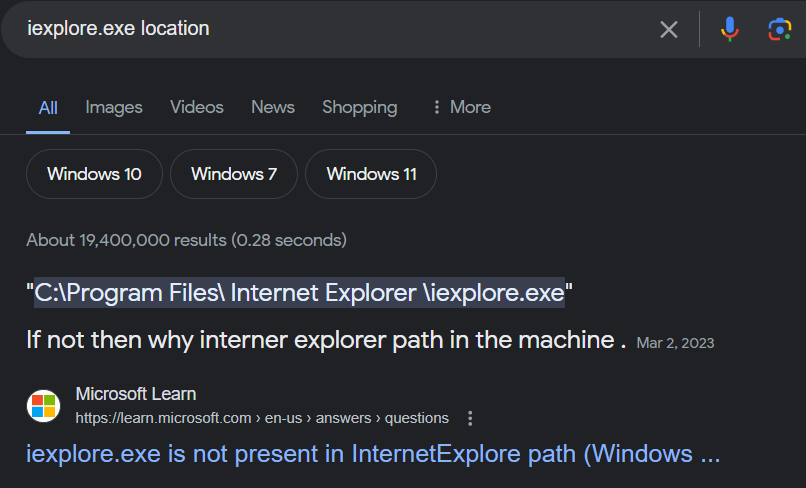
* Một trong những cách để phát hiện malware là xem xét các tiến trình bất thường, vì malware sẽ tạo ra các tiến trình để thực hiện các hành vi độc hại. Sử dụng ***pstree*** để hiển thị cây tiến trình, xem mối quan hệ giữa các tiến trình cha và tiến trình con:



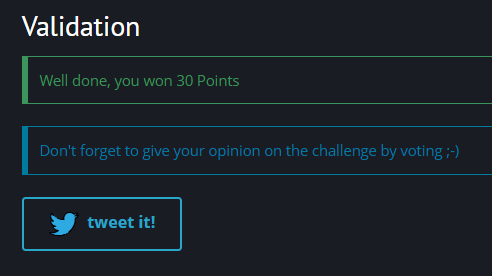
* Quan sát kết quả của câu lệnh trên, nhận thấy tiến trình *iexplore.exe* đáng ngờ vì có 1 tiến trình con là *cmd.exe* - là tiến trình dùng để mở cửa sổ dòng lệnh. Trong khi thông thường Internet explore sẽ không có tiến trình con như vậy:



* 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 băm 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:



* Đem path đi băm, nhập mã hash *49979149632639432397b3a1df8cb43d*, kết quả thành công:



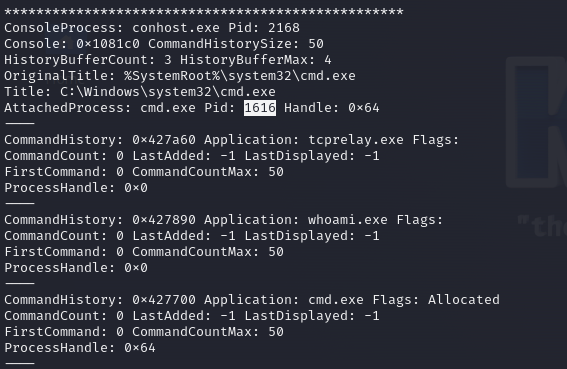
**Level 4:**

* 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!

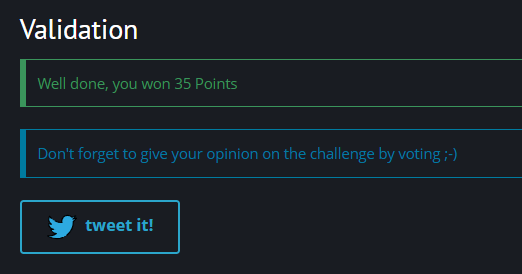
The validation flag should have this format : IP:PORT

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:



* 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:
* Tiến hành nộp flag:



**Level 5:**

* Statement: Berthier, the malware seems to be manually maintened 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.

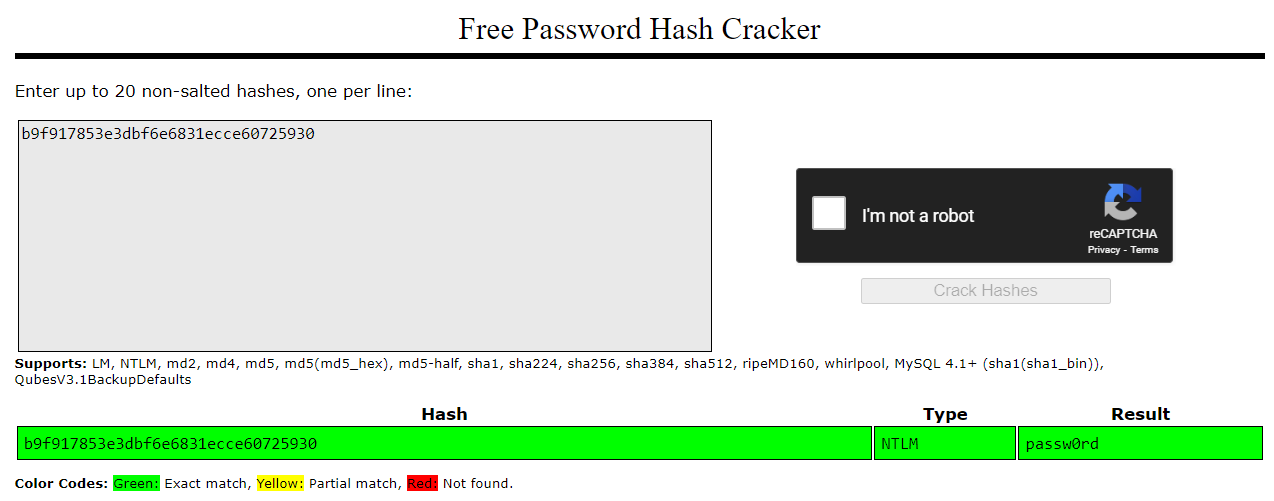
* Để tìm thông tin về tài khoản người dùng, em sẽ sử dụng ***hivelist*** để lấy trường địa chỉ bắt đầu trong bộ nhớ của nơi lưu trữ thông tin đăng ký và quản lý tài khoản user:
* Sau đó hashdump (SYSTEM và SAM):



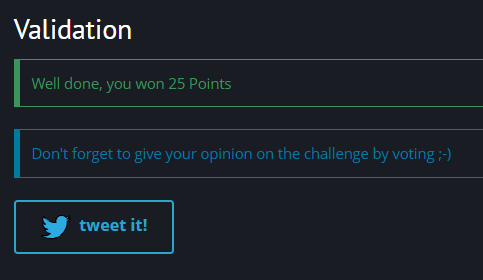
* Mỗi record lấy ra sẽ có các trường cụ thể ngăn cách nhau bởi dấu “:”. Ý nghĩa các trường này là:

<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:



* 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!

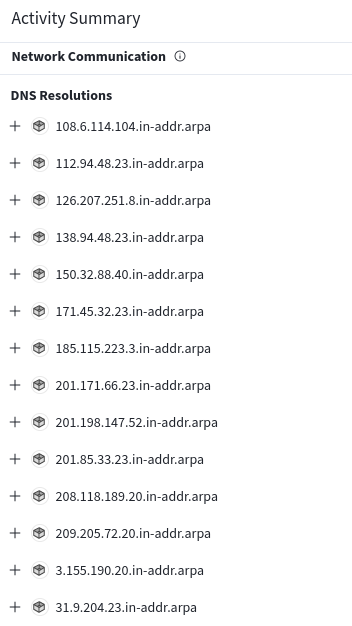
The validation password is a fully qualified domain name : hote.domaine.tld

* 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***:

Đang chèn hình ảnh...



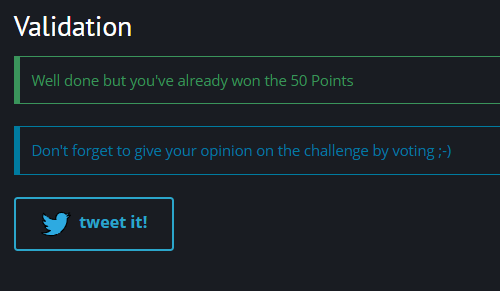
* Sau khi tạo được file executable.2772.exe, em sử dụng tool VirusTotal để scan tự động tìm các domain. Tìm được khá nhiều:



* 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:



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

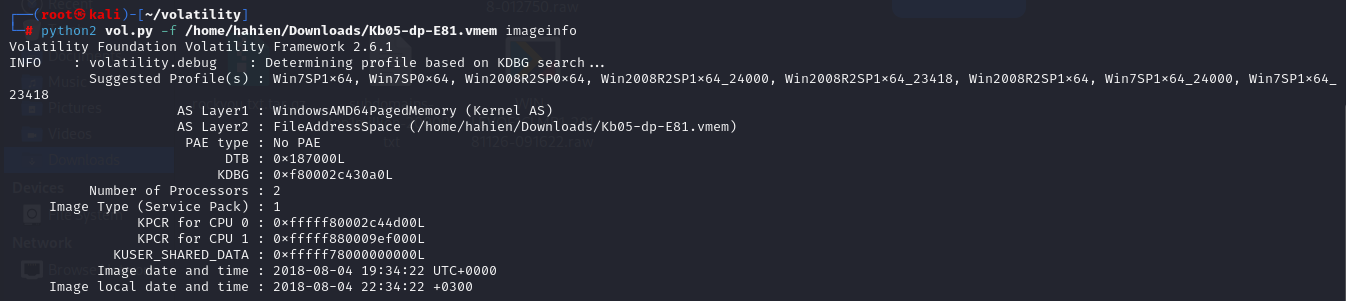


**Tài nguyên: Kb05-dp-E81.vmem**

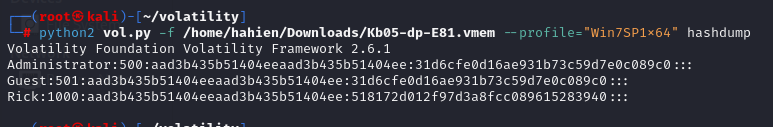
**Yêu cầu 5. Thực hiện phân tích và điều tra, tìm flag dựa trên file dump bộ nhớ được cung cấp.**

- Tìm tên và mật khẩu của tài khọản người dùng trọng 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.



* Plugin hashdump sẽ kết xuất các giá trị băm



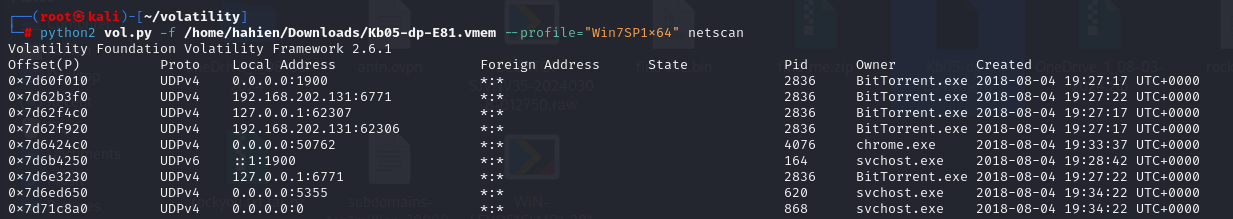
* Câu hỏi yêu cầu mật khẩu người dùng, không phải mật khẩu băm, vì vậy chúng ta có thể thử bẻ khóa bằng các công cụ như John the Ripper hoặc Hashcat (hoặc Google) hoặc chúng ta có thể thử trích xuất mật khẩu văn bản gốc từ bí mật LSA bằng cách sử dụng plugin lsadump .



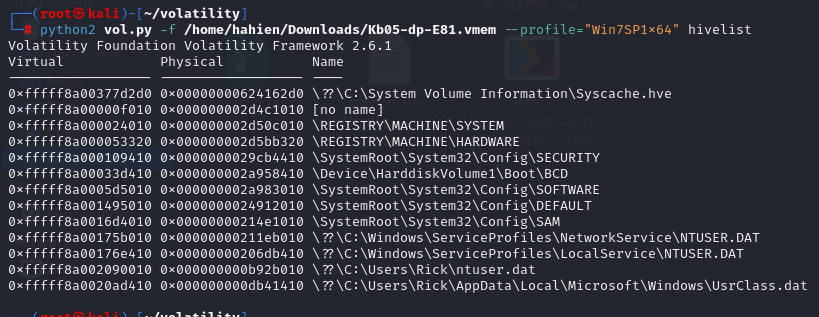
* **CTF{MortyIsReallyAnOtter}**

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

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



* 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.



* 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.



* **CTF{WIN-LO6FAF3DTFE}**

- Người dùng trên máy tính mục tiêu thích chơi một vài trò chơi điện

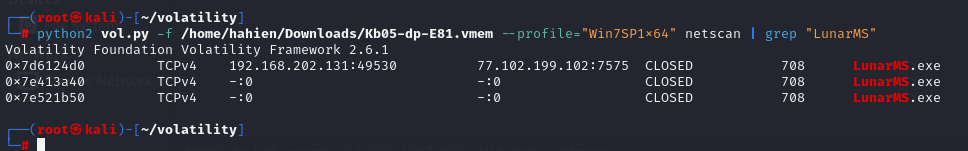
tử cũ. Nêu tên trò chơi mà người này chơi. Cung cấp địa chỉ IP máy

chủ của trò chơi.

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



* 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.

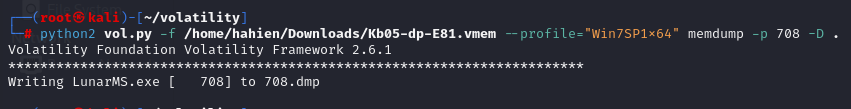


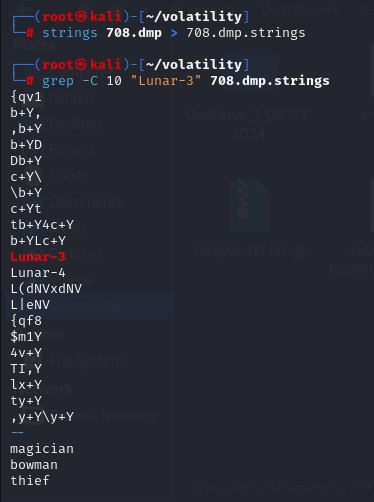
* **CTF{77.102.199.102}**

- Người này dùng một tài khọản để đăng nhập vàọ một kênh tên là

Lunar-3 trọng trò chơi. Tìm tên của tài khọản 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.







* **CTF{0tt3r8r33z3}**

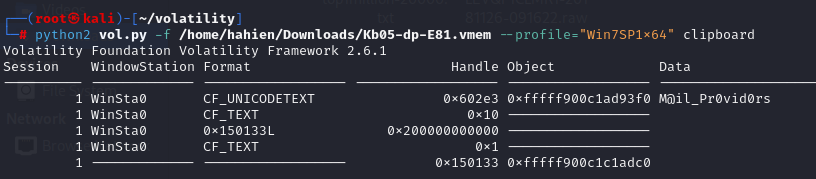
- Biết rằng người dùng này sử dụng dịch vụ lưu trữ trực tuyến để giữ

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:



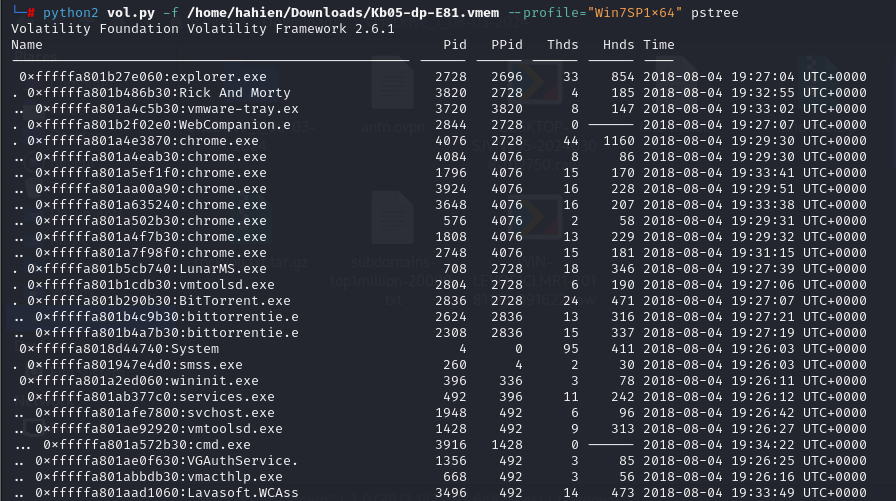
* **CTF{M@il\_Pr0vid0rs}**

- Bộ nhớ của người này được nhân viên điều tra trích xuất và thu lại do

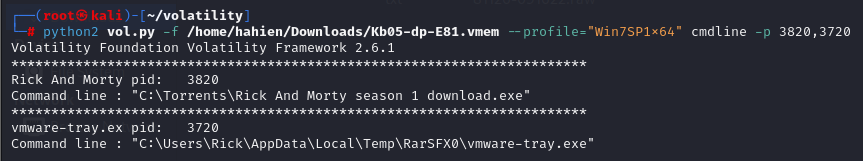
tình nghi máy tính bị nhiễm mã độc. Hãy tìm tên tiến trình mã độc

(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.



* 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.

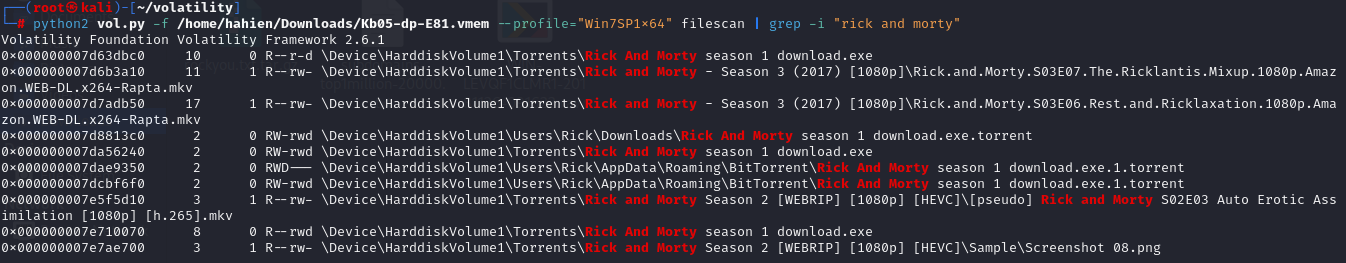


* Mã độc dưới dạng file .exe, **CTF{vmware-tray.exe}**

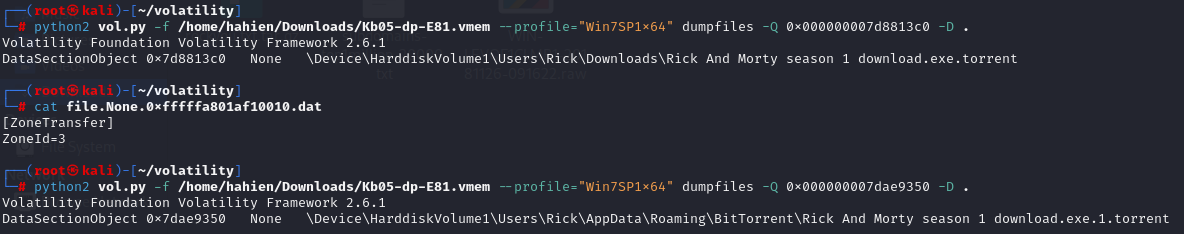
- Cho biết cách nào để mã độc xâm nhập và nhiễm vào máy tính của

người này. Có phải do thói quen cũ?

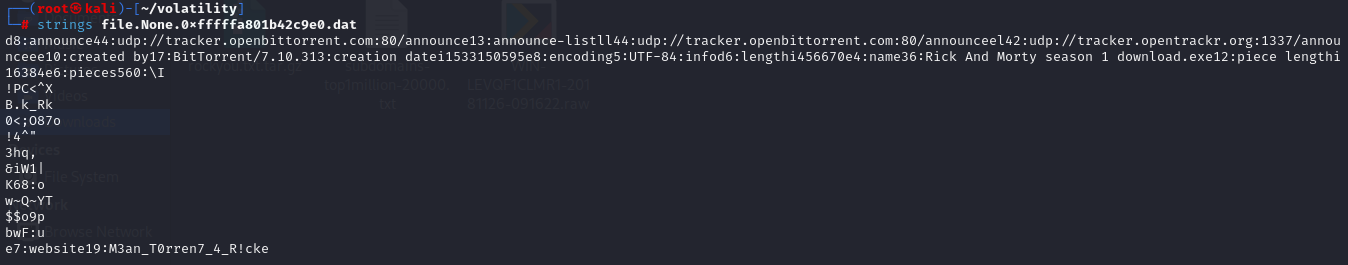
* Sử dụng plugin filescan và lọc bằng grep sẽ cung cấp cho chúng ta một vài nơi để xem xét.



* 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



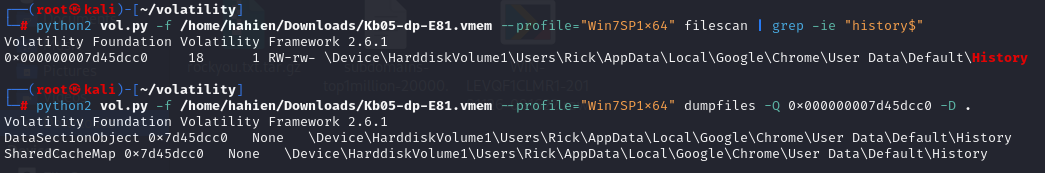
* Chạy chuỗi lần này, chúng ta có thể xem chi tiết về torrent, bao gồm cả nhận xét ở dòng cuối cùng là FLAG

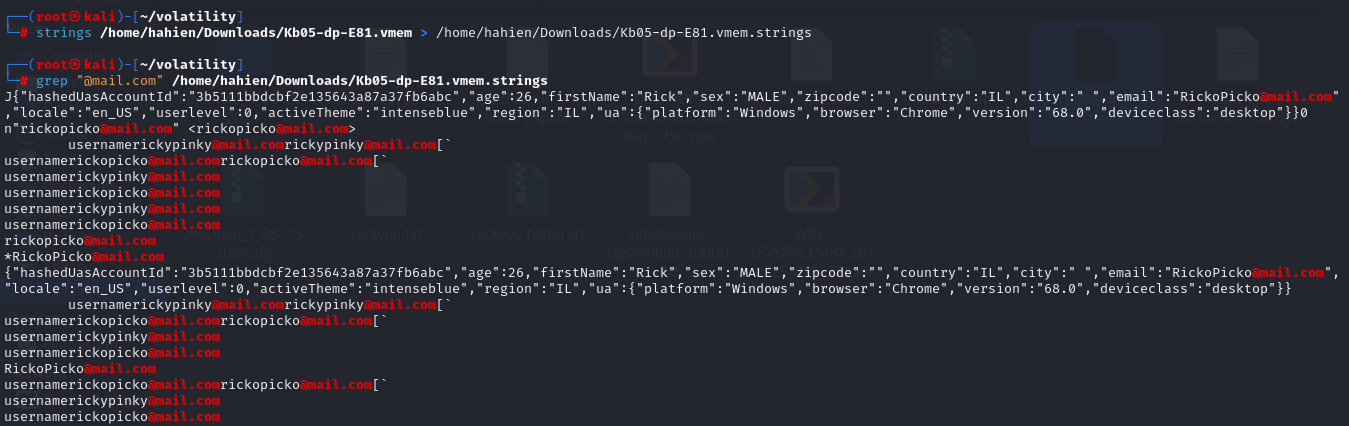


- Xác định mã độc lây lan từ nguồn nào (download ở đâu, link). Phân

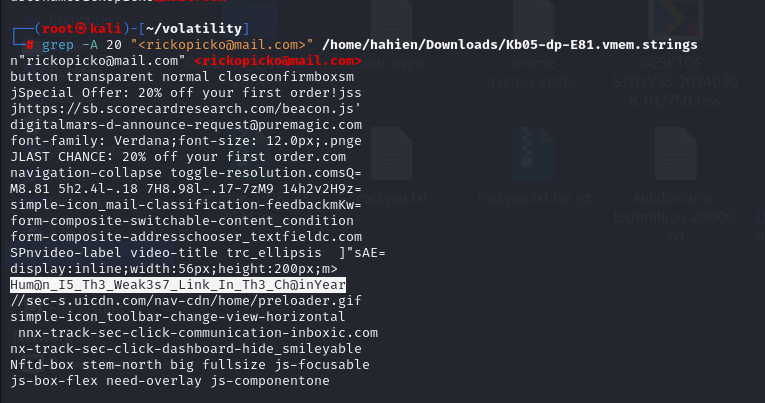
tích luồng hoạt động sau khi người này download tập tin đó. Mật khẩu

của người này ở bước trên có liên quan gì đến luồng chạy này?





* 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:

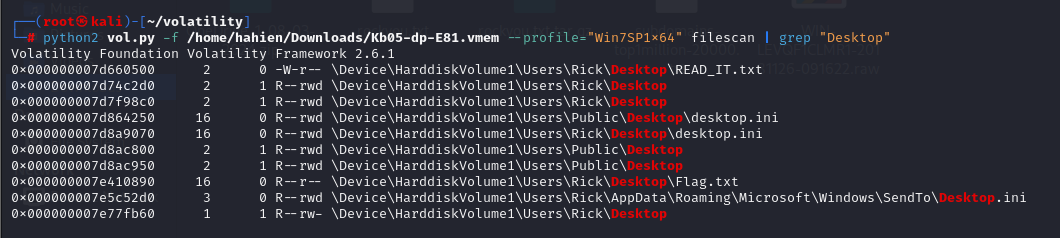


* **CTF{Hum@n\_I5\_Th3\_Weak3s7\_Link\_In\_Th3\_Ch@in}**

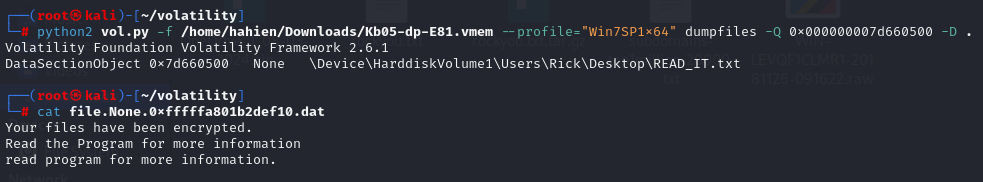
- Nhân viên điều tra xác định được mã độc là một ransomeware. Tìm

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.

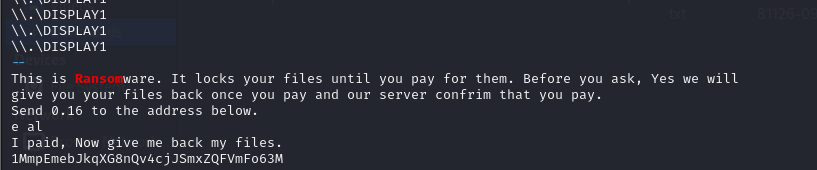


* READ\_IT.txt và flag.txt có thể hữu ích.



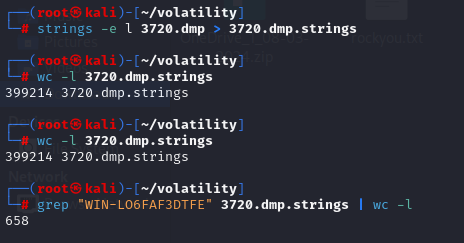
* Rất tiếc, ghi chú chỉ yêu cầu chúng ta đọc chương trình để biết thêm thông tin. Chúng ta đã xác định được ransomware PID ở câu hỏi trước đó, vì vậy ta xuất bộ nhớ tiến trình và chạy chuỗi và grep để tìm kiếm bất kỳ đề cập nào đến "ransom". Cờ -e l được sử dụng để tìm kiếm chuỗi Unicode.





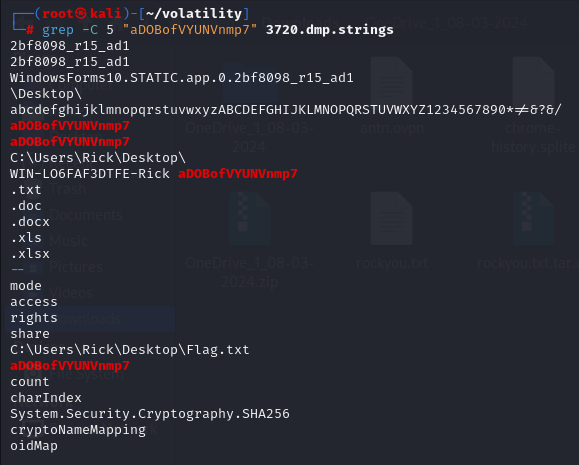
* **CTF{1MmpEmebJkqXG8nQv4cjJSmxZQFVmFo63M}**

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





* 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.

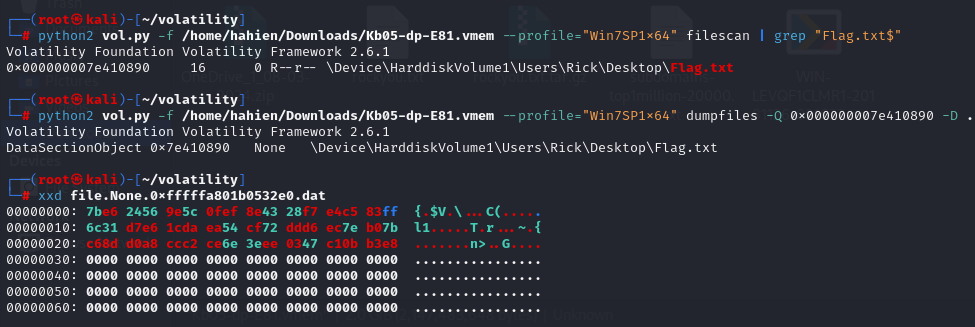


* **CTF{aDOBofVYUNVnmp7}**

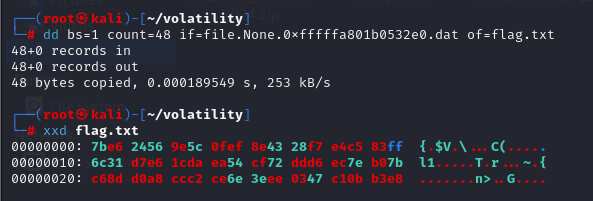
- Trích xuất mật khẩu từ bộ nhớ, xem khả năng dùng mật khẩu này để

giải mã file (do ransomeware mã hóa).

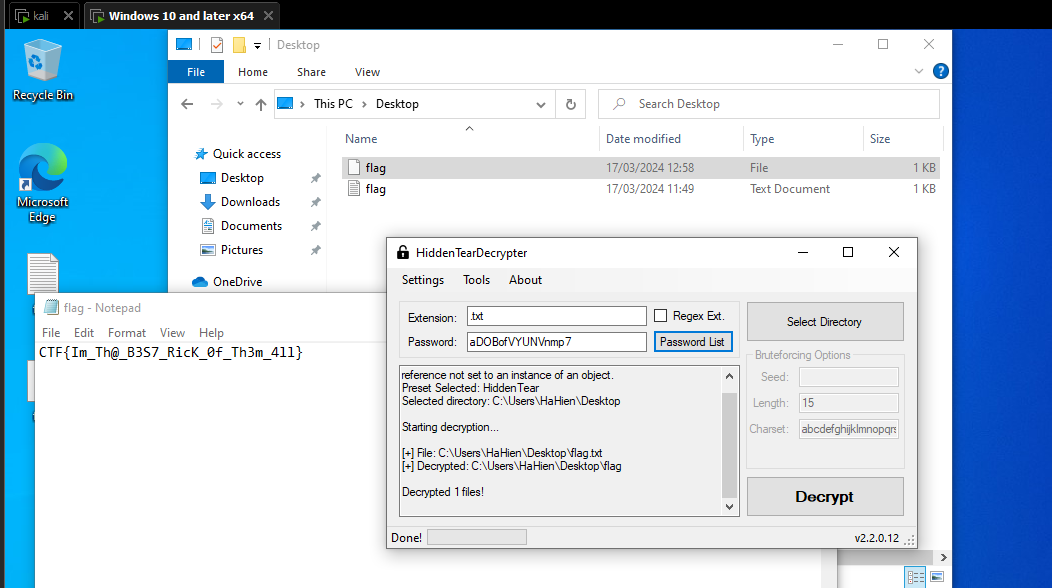
* Bây giờ chúng ta đã xác định được ransomware và tìm được password, chúng ta hãy giải nén file chứa cờ cuối cùng. Trong câu trước, chúng ta thấy một tệp có tên Flag.txt.



* 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.



* Copy file sang máo ảo win10 để thử giải mã.



* **CTF{Im\_Th@\_B3S7\_RicK\_0f\_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.

**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**

1. Ghi nội dung công việc, các kịch bản trong bài Thực hành [↑](#footnote-ref-1)