BỘ THÔNG TIN VÀ TRUYỀN THÔNG

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

------o0o------

BÁO CÁO BÀI THỰC HÀNH

PHÂN TÍCH TĨNH WINDOWS

Giảng viên hướng dẫn: TS. Nguyễn Ngọc Điệp

Sinh viên thực hiện: Phạm Vũ Minh Hiếu – B20DCAT061

Lớp: D20CQAT01-B

Mã sinh viên: B20DCAT061

Hà Nội - 2024

MỤC LỤC

DANH MỤC CÁC HÌNH VẼ

DANH MỤC CÁC BẢNG BIỂU

DANH MỤC VIẾT TẮT

Từ viết tắt	Thuật ngữ tiếng Anh / Giải thích	Thuật ngữ tiếng Việt / Giải thích
Peid	PE Identifier	Công cụ nhận diện loại packer hoặc compiler sử dụng trong tệp PE (Portable Executable).
CFF	CFF Explorer	Công cụ phân tích và chỉnh sửa file PE, hỗ trợ kiểm tra thông tin header, import/export.
UPX	Ultimate Packer for eXecutables	Công cụ nén tệp thực thi để giảm kích thước, thường được sử dụng bởi mã độc để làm khó quá trình phân tích.
SCP	Secure Copy Protocol	Giao thức sao chép file an toàn qua SSH.
IOC	Indicator of Compromise	Dấu hiệu nhận diện mã độc, bao gồm hash, IP, domain, hoặc các hành vi đáng ngờ.
MD5	Message-Digest Algorithm 5	Thuật toán hash 128-bit, thường dùng để kiểm tra tính toàn vẹn của file.
SHA1	Secure Hash Algorithm 1	Thuật toán hash 160-bit, cung cấp độ an toàn cao hơn MD5.
	Payload	Tải trọng, đoạn mã thực hiện nhiệm vụ chính của mã độc, ví dụ: đánh cắp dữ liệu hoặc tấn công hệ thống.
	PowerShell	Shell dòng lệnh mạnh mẽ trong Windows, thường bị mã độc lợi dụng để thực thi mã độc hoặc tải payload.

	Remote	Điều khiển từ xa hoặc kết nối từ xa giữa hai hệ thống.
	Checkwork	Quá trình kiểm tra, xác nhận công việc hoặc trạng thái của hệ thống.
	File	Tệp tin, thường là đối tượng chính của các công cụ như Peid hoặc CFF Explorer để phân tích mã độc.
	Version	Phiên bản của file, ứng dụng, hoặc phần mềm được kiểm tra thông qua header hoặc metadata.
	Submit	Quá trình gửi file hoặc IOC lên các nền tảng như VirusTotal để kiểm tra.
Mal	Malware	Một file có hành vi độc hại
Stic	Static	Phân tích tĩnh

1.1. Giới thiệu bài thực hành

Bài thực hành " Tìm hiểu về phân tích tĩnh cơ bản mã độc Windows " được thiết kế nhằm giúp sinh viên hiểu rõ hơn về kỹ thuật kỹ thuật phân tích tĩnh mức cơ bản, một phương thức phân tích phổ biến phổ biến trong lĩnh vực phân tích mã độc, đặc biệt là trong việc nhận diện và xử lý các mối đe dọa từ phần mềm độc hại. Đây là bước khởi đầu quan trọng để sinh viên làm quyen với quy trình phân tích mã độc và nhận thức được cách thức hoạt động của các chương trình độc hại

Trong bài thực hành này, sinh viên sẽ tìm hiểu cách thực hiện phân tích mã độc để đánh giá các đặc điểm của file thực thi mà không cần chạy nó. Sinh viên sẽ học các sử dụng các công cụ như PeiD, CFF, Strings, UPX, … Và cả VirusTotal để kiểm tra, thu thập thông tin về cấu trúc file, các chuỗi ký tự liên quan và các thư viện được sử dụng bởi mã độc. Đây là một kỹ năng cơ bản nhưng rất quan trọng, vì nó giúp sinh viên xác định được các đặc điểm đáng ngờ trong mã độc mà không gây rủi ro cho môi trường thực thi.

Thông qua bài thực hành này, sinh viên không chỉ hiểu cách phân tích tĩnh mã độc mà còn nhận thức được những rủi ro bảo mật tiềm ẩn mà mã độc có thể gây ra. Điều này giúp sinh viên nâng cao kiến thức và kỹ năng về bảo mật thông tin, đồng thời làm quen với các công cụ và quy trình phân tích cần thiết trong công việc điều tra và đánh giá an toàn hệ thống. Đây cũng là một kỹ thuật cơ bản nhưng hữu ích, giúp sinh viên chuẩn bị tốt hơn trong các công việc liên quan đến xử lý mã độc và bảo mật thông tin trong tương lai.

1.2. Nội dung và hướng dẫn bài thực hành

1.2.1. Mục đích

Giúp sinh viên tìm hiểu khái niệm về phân tích mã độc, sử dụng các công cụ khác nhau như CFF, Peid, Dei,…., từ đó làm quen với các công cụ và tìm hiểu được các hành vi đáng ngờ (như tải payload, thực thi PowerShell), và phát hiện liên kết độc hại.

1.2.2. Yêu cầu đối với sinh viên

Có kiến thức cơ bản về hệ điều hành Window – Linux và các công cụ như Strings, CFF, Peid, Die, UPX.

1.2.3. Nội dung thực hành

Khởi động bài lab:

Vào terminal, gõ:

labtainer -r mal-stic

(chú ý: sinh viên sử dụng mã sinh viên của mình để nhập thông tin email người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)

Sau khi khởi động xong sẽ có 1 terminal ảo sẽ xuất hiện, có tên là mal-stic. Trên terminal ***mal-stic*** đã được cài đặt sẵn công cụ để remote tới máy Windows chứa mã độc là Xfreerdp.

xfreerdp /u:user /p:password /v:ip

(user: tài khoản máy Window, password: mật khẩu của tài khoản, ip: IP của máy Window)

Ví dụ: xfreerdp /v:192.168.18.128 /u:minhh /p:1

Sau khi remote thành công sang máy Window, sẽ thấy 1 folder **mal-stic** trên màn hình, bên trong đó có sẵn 1 file mã độc và 1 file ATTT.txt (điền output mà bài yêu cầu để checkwork)

- Sinh viên sử dụng công cụ *certutil, CFF* hoặc một công cụ nào khác mà sinh viên biết để lấy hash của file BasicMalware trên màn hình Desktop

Certutil -hashfile BasicMalware

- Sinh viên submit hash lên VirusTotal để kiểm tra đánh giá của các AV để xác minh xem có phải file độc hại hay không, có bao nhiêu AV đánh giá là file độc hại. Và copy hash hoàn chỉnh của VirusTotal, số các AV đánh giá là file độc hại và viết lại vào file ATTT.txt trong thư mục DATN. Mỗi một giá trị một dòng như ví dụ bên dưới. Nếu không thấy thì check unpack bước bên dưới, xong quay lại check lại hash.

23/50

0fa1498340fca6c562cfa389ad3e93395f44c72fd128d7ba08579a69aaf3b126

- Sinh viên dùng PeiD, Die hoặc một công cụ nào khác mà sinh viên biết để kiểm tra xem file đã cho có bị nén hay không. Và viết vào file ATTT.txt tên viết tắt (không viết version) của công cụ/phương thức nén. Và tìm cách giải nén nó

Upx -d path/to/file

- Sinh viên dùng CFF/ResourceHacker/… hoặc các công cụ khác mà sinh viên biết để tìm xem có những thư viện lớn nào được import vào trong file đã cho, và viết vào trong file ATTT.txt (viết hoa tất cả). Ví dụ về cách trình bày ở bên dưới.

ABC.dll

DEF.dll

- Sinh viên dùng Strings hoặc các công cụ khác mà sinh viên biết để lấy các chuỗi ký tự trong file và thu thập các chỉ báo dựa trên máy chủ và mạng và viết vào trong file ATTT.txt

Strings BasicMalware.exe

Cách trình bày:

/winup.exe

https://abc.com

- Sau khi xong hết các nhiêm vụ, sinh viên lưu lại file.
- Về lại terminal mal-stic dùng scp để tiến hành kéo thư mục DATN từ máy Windows về (có thể dùng cách khác để kéo file về mà sinh viên biết).

scp -r minhh@192.168.18.128:"C:/Users/minhh/Desktop/mal-stic" ~/

- Lệnh sẽ tự tạo 1 folder share trên máy Linux, tiến hành cd sang folder **mal-stic** và cat file ATTT.txt

cd mal-stic

cat ATTT.txt

- Cuối cùng ta sẽ checkwork
- Kết thúc bài lab:

Trên terminal đầu tiên sử dụng câu lệnh sau để kết thúc bài lab:

stoplab mal-stic

Khi bài lab kết thúc, một tệp zip lưu kết quả được tạo và lưu vào một vị trí được hiển thị bên dưới stoplab.

Khởi động lại bài lab:

Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, dùng câu lệnh:

startlab –r mal-stic

1.3. Phân tích yêu cầu bài thực hành

Bài thực hành yêu cầu sinh viên làm quen với các công cụ thường dùng để bước đầu thu thập thông tin qua phân tích tĩnh như CFF, Peid, Die, Strings, UPX. Những công cụ chuyên dùng để phân tích file độc hại nhằm phát hiện các đặc điểm đáng ngờ và trích xuất các chỉ số nguy cơ (IOC). Để hoàn thành bài thực hành, sinh viên cần phân tích tĩnh một file độc hại đã được cài sẵn trong môi trường làm việc.

Sinh viên sẽ bắt đầu bài thực hành bằng lệnh khởi tạo startlab <tên bài lab> và nhập MSV của mình. Sau đó, sử dụng công cụ CFF cho sẵn để xác định các đặc điểm của file. Tiếp theo, sinh viên sẽ sử dụng Peid để xác định xem file có bị nén hay không, nếu có thì xác định file bị nén bởi công cụ nào và giải nén file để thu dược file unpack. Sau đó dùng Strings để trích xuất các IOC như URL, IP, hoặc domain, và ghi nhận các phát hiện này. Để kiểm tra tính nguy hiểm của các IOC, file đã cho, sinh viên sẽ sử dụng các công cụ trực tuyến như VirusTotal bằng cách submit lên trang web hash file độc hại để thu thập kết quả đánh giá của các AV, submit URL để tiến hành thu thập đánh giá của trang web về URL đã submit và ghi lại kết quả phân tích vào file kết quả ATTT.txt. Cuối cùng, sinh viên sẽ tải file kết quả về máy Linux bằng lệnh scp và dừng bài lab bằng lệnh stoplab <tên bài lab>.

Hệ thống yêu cầu file kết quả của sinh viên phải bao gồm mã hash duy nhất của file mã độc thông qua VirusTotal, các IOC đã trích xuất được, đặc

điểm ban đầu của file như bị nén bởi công cụ nào, các thư viện lớn được file độc hại sử dụng và các phát hiện liên quan đến hành vi mã độc.

1.4. Thiết kế bài thực hành

Trên môi trường máy ảo Ubuntu được cung cấp, sử dụng docker tạo ra container mang tên "mal-stic".

Cấu hình docker gồm có:

- mal-stic: Lưu cấu hình cho máy thực hành, trong đó gồm có:
    - Tên máy: mal-stic
- Config: Lưu cấu hình hoạt động của hệ thống.
- Dockerfile: Mô tả cấu hình của container mal-stic, trong đó:
    - mal-stic: Sử dụng các thư viện mặc định của hệ thống và tích hợp sẵn công cụ xfreerdp và dcp
    - xfreerdp: Hỗ trợ remote sang máy tính Windows
    - scp: Dùng để truyền file từ máy Windows về Linux

Trên môi trường máy Windows được cài:

- Mở tính năng remote cho phép máy ubuntu remote tới
- SSH server
- Thư mục gồm file độc hại để phân tích và một file để điền kết quả.
- Các công cụ dùng để phân tích như CFF Explorer, Peid, UPX, IDA Pro, Strings.

Các nhiệm vụ trong bài thực hành phải thực hiện để thành công

- Lấy mã hash duy nhất của file mã độc qua VirusTotal
- Kiểm tra đặc điểm của file độc hại xem có bị pack hay không.

- Thu thập thông tin về các thư viện mà file đọc hại sử dụng.
- Thu thập thông tin về IOC.
- Chuyển file kết quả về máy Linux

Kết thúc bài lab sau khi hoàn thành tất cả các nhiệm vụ và kiểm tra file kết quả ATTT.txt.

Để đánh giá được sinh viên đã hoàn thành bài thực hành hay chưa, cần chia bài thực hành thành các nhiệm vụ nhỏ, mỗi nhiệm vụ cần phải chỉ rõ kết quả để có thể dựa vào đó đánh giá, chấm điểm. Do vậy, trong bài thực hành này hệ thống cần ghi nhận các thao tác, sự kiện được mô tả và cấu hình như bảng 1,2,3:

Bảng 1. Bảng Results

Result Tag	Container	File	Field Type	Field ID	Timestamp Type
hash	mal-stic	cat.stdout	CONTAINS	0fa1498340fca6c562cfa389ad3e93395f44c72fd128d7ba08579a69aaf3b126	File
vt-check	mal-stic	cat.stdout	CONTAINS	63/72	File
pack	mal-stic	cat.stdout	CONTAINS	UPX	File
_import-dll1	mal-stic	cat.stdout	CONTAINS	MSVCRT.dll	File
_import-dll2	mal-stic	cat.stdout	CONTAINS	ADVAPI32.dll	File

_import-dll3	mal-stic	cat.stdout	CONTAINS	KERNEL32.dll	File
_c2c	mal-stic	cat.stdout	CONTAINS	http://www.practicalmalwareanalysis.com/updater.exe	File
_file-exe1	mal-stic	cat.stdout	CONTAINS	\winup.exe	File
_file-exe2	mal-stic	cat.stdout	CONTAINS	\system32\wupdmgr.exe	File
_file-exe3	mal-stic	cat.stdout	CONTAINS	\system32\wupdmgrd.exe	File

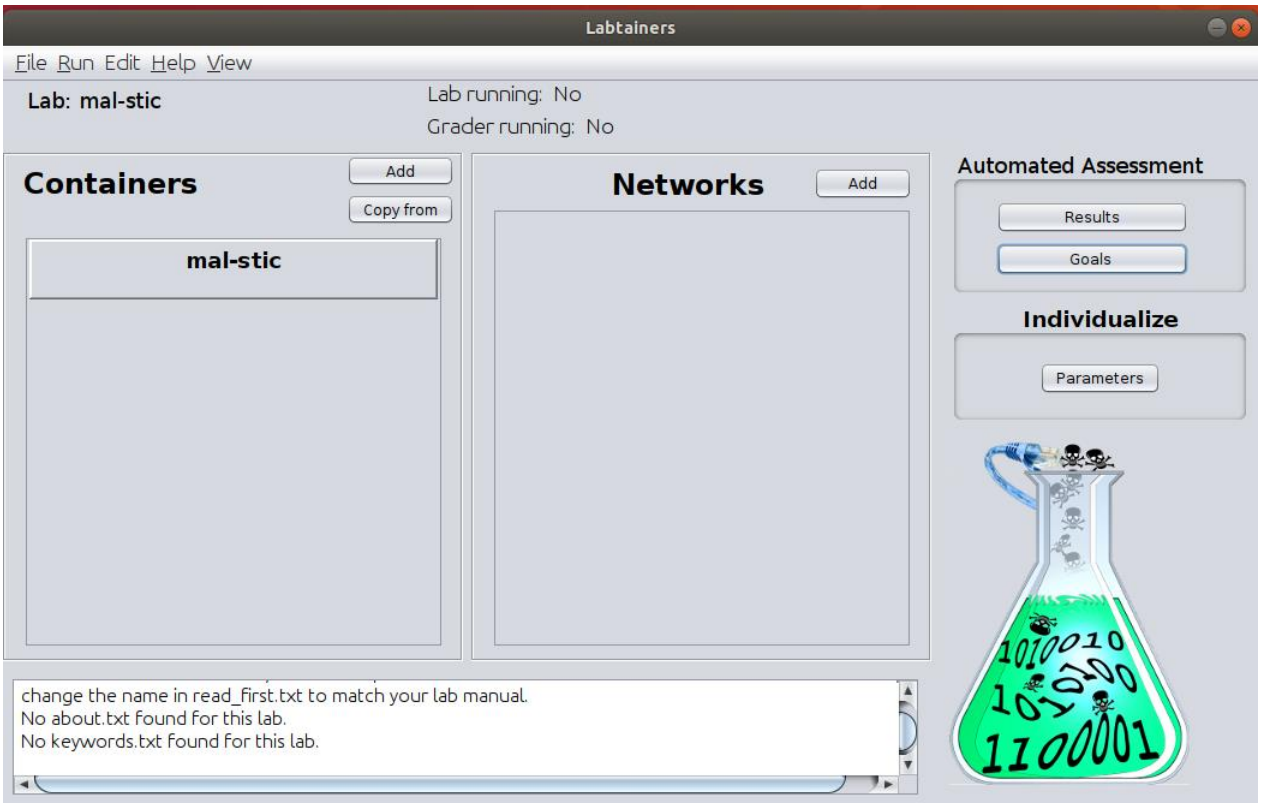
Bảng 2. Bảng Goals

Goal ID	Goal Type	Boolean	Boolean Result Tags
import-dll	boolean	_import-dll1 and _import-dll2 and _import-dll3	hash
ioc	boolean	_c2c and _file-exe1 and _file-exe3 and _file-exe3	hash

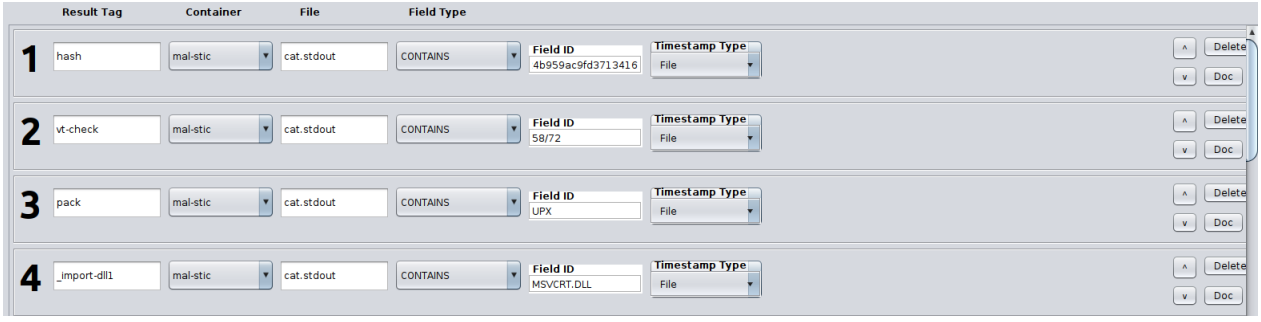
Bảng 3. Bảng Paramater

Param ID	Operator	File name	Symbol	Step	Hashstring

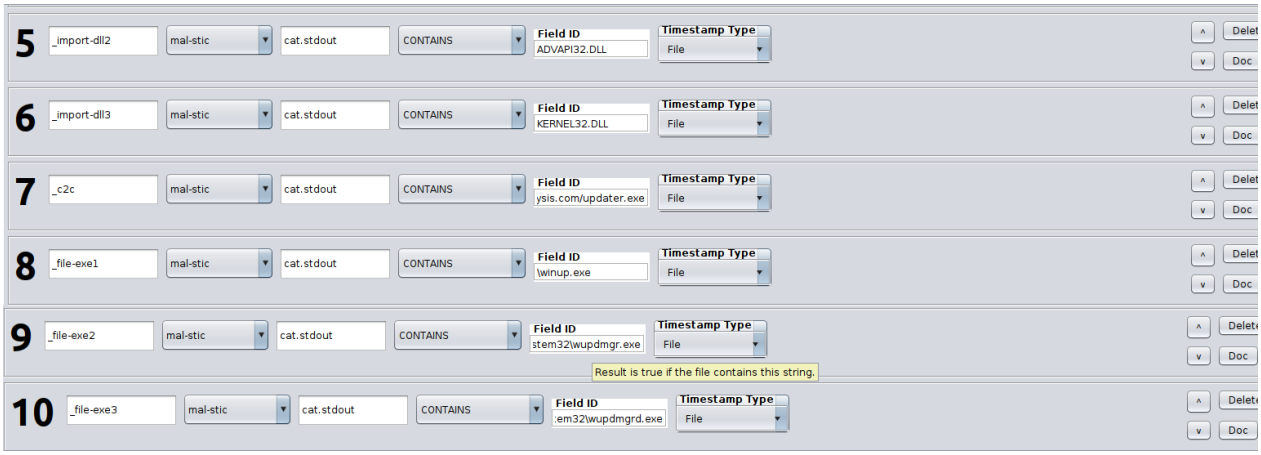
1.5. Cài đặt và cấu hình các máy ảo



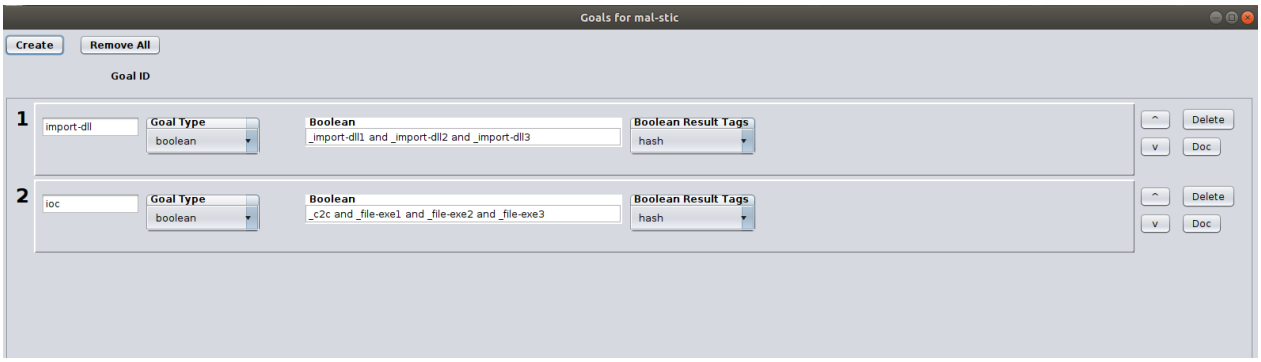
Hình 1: Giao diện Labedit



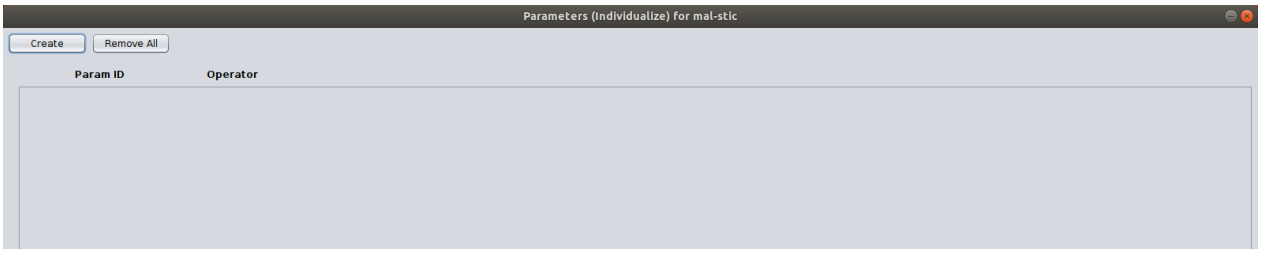
Hình 2: Cài đặt Result 1



Hình 3: Cài đặt phần Result 2



Hình 4: Cài đặt phần Goals



Hình 5: Cài đặt phần Parameter

```
ARG lab
ARG labdir
ARG imagedir
ARG user_name
ARG password
ARG apt_source
ARG version
LABEL version=$version
ENV APT_SOURCE $apt_source
RUN /usr/bin/apt-source.sh

RUN apt-get update
RUN apt install freerdp2-x11 -y
#
#  put package installation here, e.g.,
#     RUN apt-get update && apt-get install -y --no-install-recommends somepackage
#
#
#
# Install the system files found in the _system directory
#
ADD $labdir/$imagedir/sys_tar/sys.tar /
ADD $labdir/sys_$lab.tar.gz /
#
RUN useradd -ms /bin/bash $user_name
RUN echo "$user_name:$password" | chpasswd
RUN adduser $user_name sudo
# replace above with below for centos/fedora
#RUN usermod $user_name -a -G wheel


#
#  **** Perform all root operations, e.g.,           ****
#  **** "apt-get install" prior to the USER command. ****
#
```

Hình 6: Dockerfiles

1.6. Tích hợp và triển khai

1.6.1. Docker Hub

```
student@ubuntu:~/labtainer/trunk/labs$ git init
Initialized empty Git repository in /home/student/labtainer/trunk/labs/.git/
student@ubuntu:~/labtainer/trunk/labs$ git config --global user.name minhh310
student@ubuntu:~/labtainer/trunk/labs$ git config --global user.email hieupvm310@gmail
student@ubuntu:~/labtainer/trunk/labs$ git commit mal-stic-ida -m "Adding an IModule"
[master (root-commit) f65a73d] Adding an IModule
 19 files changed, 516 insertions(+)
```

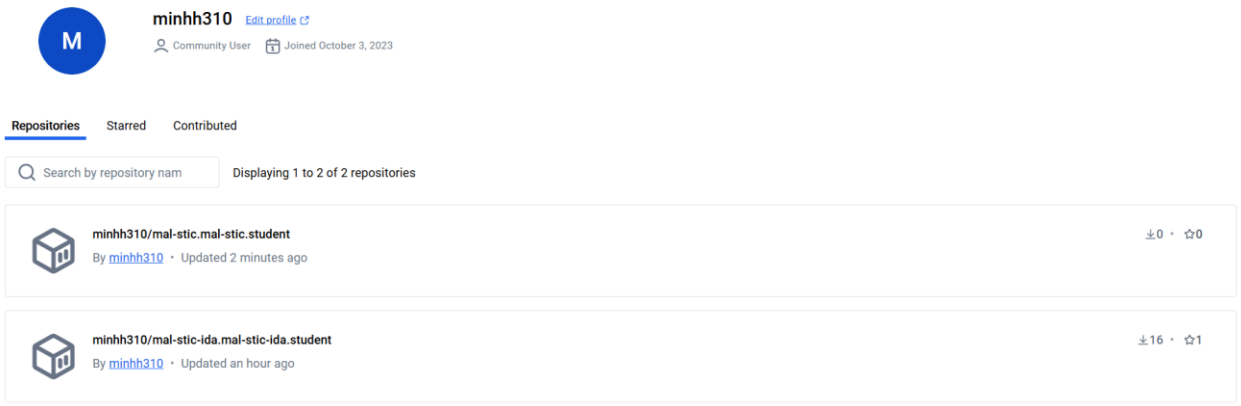
Hình 7: Add và comit bài lab

```
student@ubuntu:~/labtainer/trunk/labs$ git commit mal-stic -m "Adding an IModule"
[master 0234fdc] Adding an IModule
 20 files changed, 508 insertions(+)
 create mode 100644 mal-stic/config/basic-static-home_tar.list
 create mode 100644 mal-stic/config/mal-stic-home_tar.list
 create mode 100644 mal-stic/config/parameter.config
 create mode 100644 mal-stic/config/start.config
 create mode 100644 mal-stic/dockerfiles/Dockerfile.mal-stic.mal-stic.student
 create mode 100644 mal-stic/docs/read_first.txt
 create mode 100644 mal-stic/instr_config/goals.config
 create mode 100755 mal-stic/instr_config/pregrade.sh
 create mode 100644 mal-stic/instr_config/results.config
 create mode 100644 mal-stic/mal-stic/_bin/.treataslocal.swo
 create mode 100755 mal-stic/mal-stic/_bin/fixlocal.sh
 create mode 100644 mal-stic/mal-stic/_bin/treataslocal
 create mode 100644 mal-stic/mal-stic/_system/etc/login.defs
 create mode 100644 mal-stic/mal-stic/_system/etc/securetty
 create mode 100644 mal-stic/mal-stic/basic-static.basic-static.student.tar.gz
 create mode 100644 mal-stic/mal-stic/home_tar/home.tar
 create mode 100644 mal-stic/mal-stic/mal-stic.mal-stic.student.tar.gz
 create mode 100644 mal-stic/mal-stic/sys_basic-static.basic-static.student.tar.gz
 create mode 100644 mal-stic/mal-stic/sys_mal-stic.mal-stic.student.tar.gz
 create mode 100644 mal-stic/mal-stic/sys_tar/sys.tar
```

Hình 8: Thêm bài lab muốn lưu trữ

```
student@ubuntu:~/labtainer/trunk$ cd distrib/
student@ubuntu:~/labtainer/trunk/distrib$ ./publish.py -d -l mal-stic
adding [nmaplab]
adding [httplab]
adding [liveforensics]
adding [bind-shell]
adding [tlab]
adding [metasploitable-test]
adding [kali-test]
adding [my-remote-dns]
adding [remote-dns2]
adding [remote-dns]
adding [backups]
adding [centos-log]
adding [dhcp-test]
adding [xlab]
adding [softplc]
adding [iptables]
adding [grfics]
adding [usbtest]
adding [ida]
adding [centossix]
```

Hình 9: Quá trình tải bài lab lên dockerhub

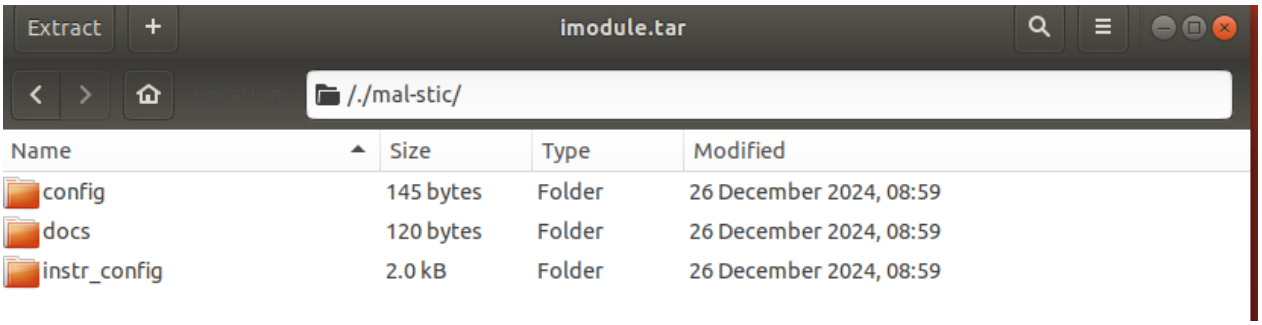


Hình 10: Đẩy thành công

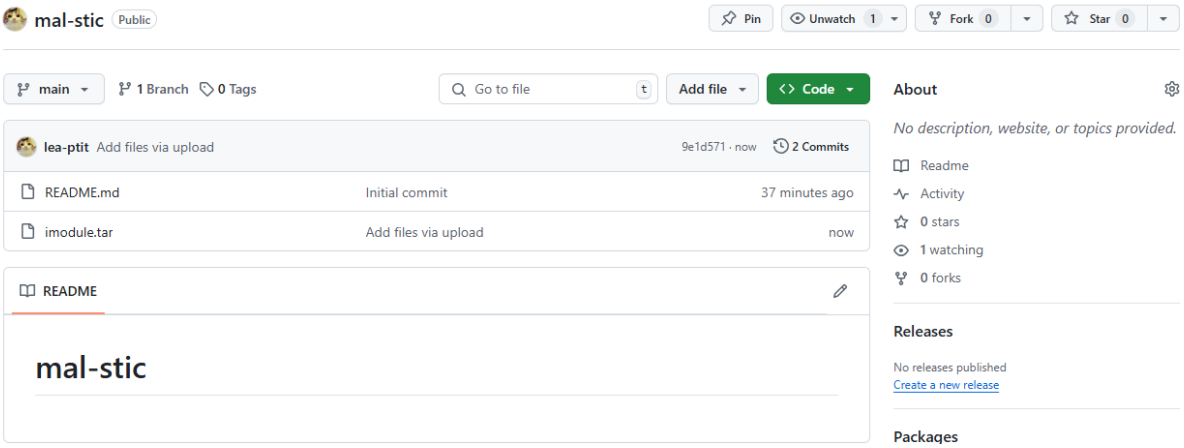
1.6.2. Github

```
student@ubuntu:~/labtainer/trunk/distrib$ create-imodules.sh
lab is mal-stic-ida
Do docs
lab is mal-stic
Do docs
************************************************************
**  Post /home/student/labtainer/trunk/imodule.tar to your web server **
************************************************************
student@ubuntu:~/labtainer/trunk/distrib$
```

Hình 11: Tạo file Imodule.tar



Hình 12: File imodule.tar chứa bài thực hành

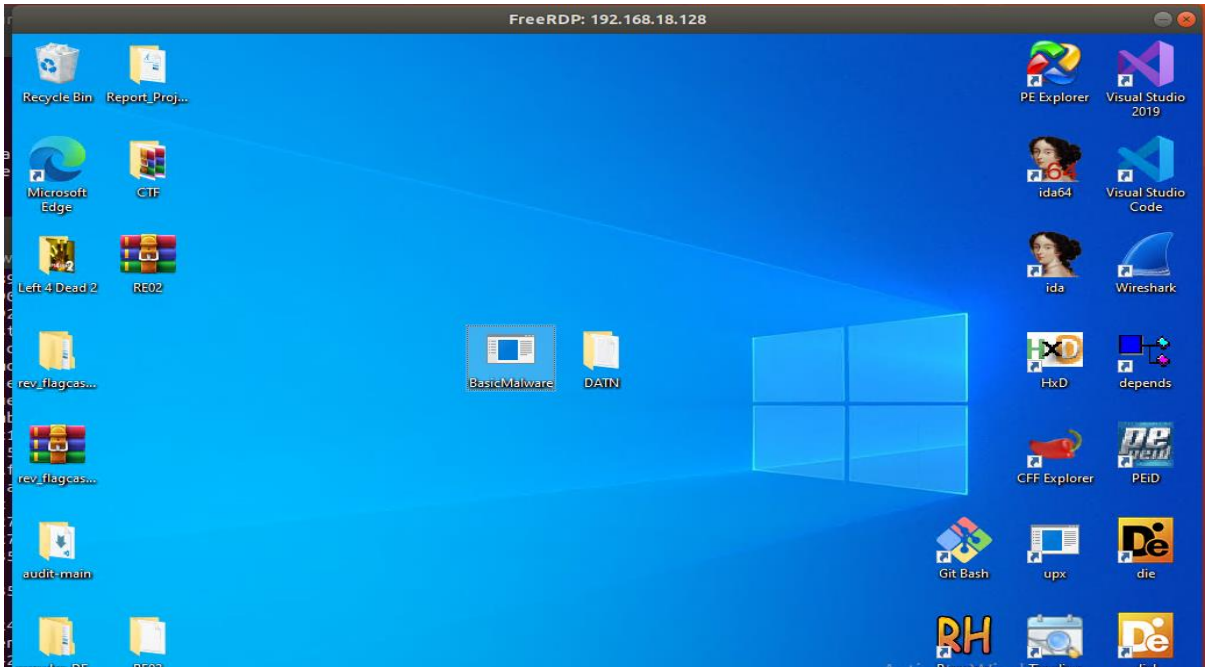


Hình 13: Đẩy lên github thành công

1.7. Thử nghiệm và đánh giá

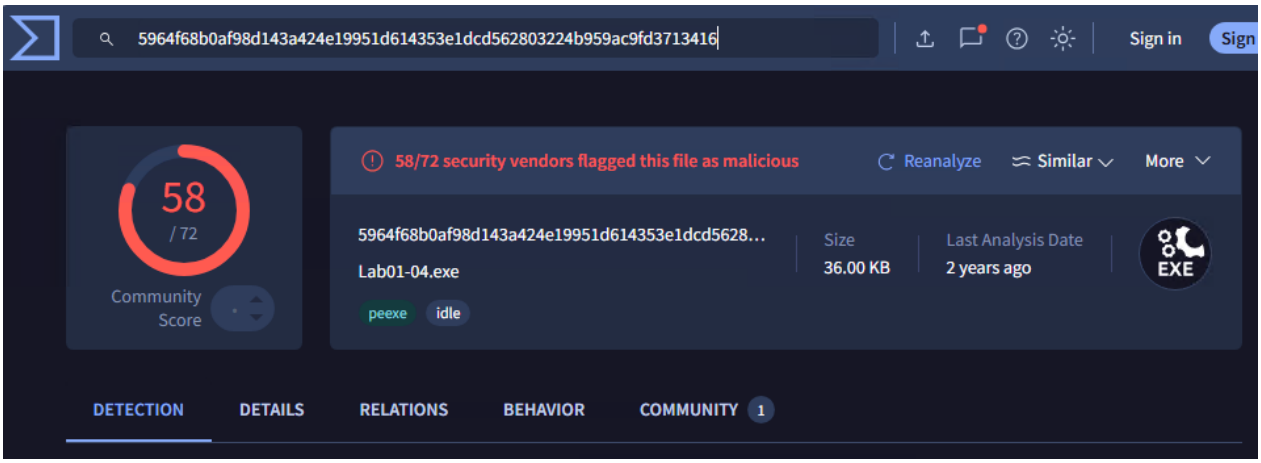
Bài thực hành đã được xây dựng thành công, dưới đây là hình ảnh minh họa về bài thực hành:

Sau khi remote vào trong Windows 10 từ terminal của bài lab ta thu được giao diện như sau.



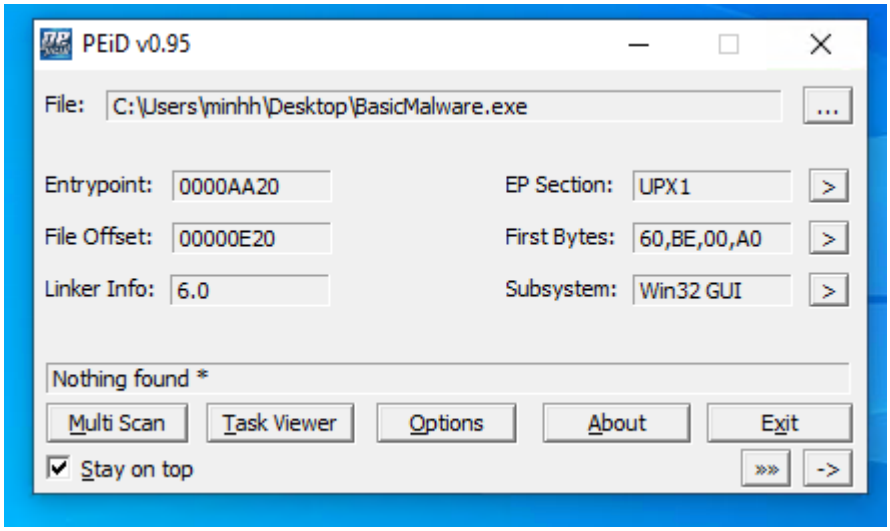
Hình 14: Remote sang máy win 10

Tiến hành lấy hash duy nhất của file độc hại bằng công cụ CFF, kéo thả file vào công cụ CFF, ta thu được hash MD5 hoặc SHA1. Xong mang hash để submit lên VirusTotal và thu đánh giá của AV về file cũng như là lấy về hash duy nhất



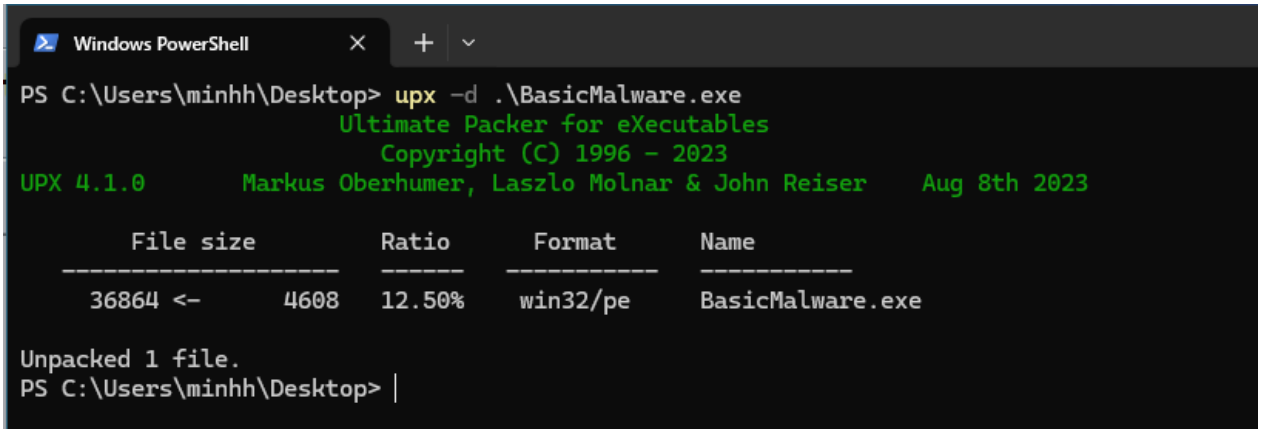
Hình 15: Lấy hash duy nhất và đánh giá của AV

Trong trường hợp khi submit lên VirusTotal ta thấy không có kết qyar trả về thì ta tiến hành kiểm tra xem file độc hại có bị nén hay không. Nếu có thì giải nén bằng công cụ phù hợp với bản nén, như trong bài lab thì công cụ nén và giải nén cần thiết là UPX.



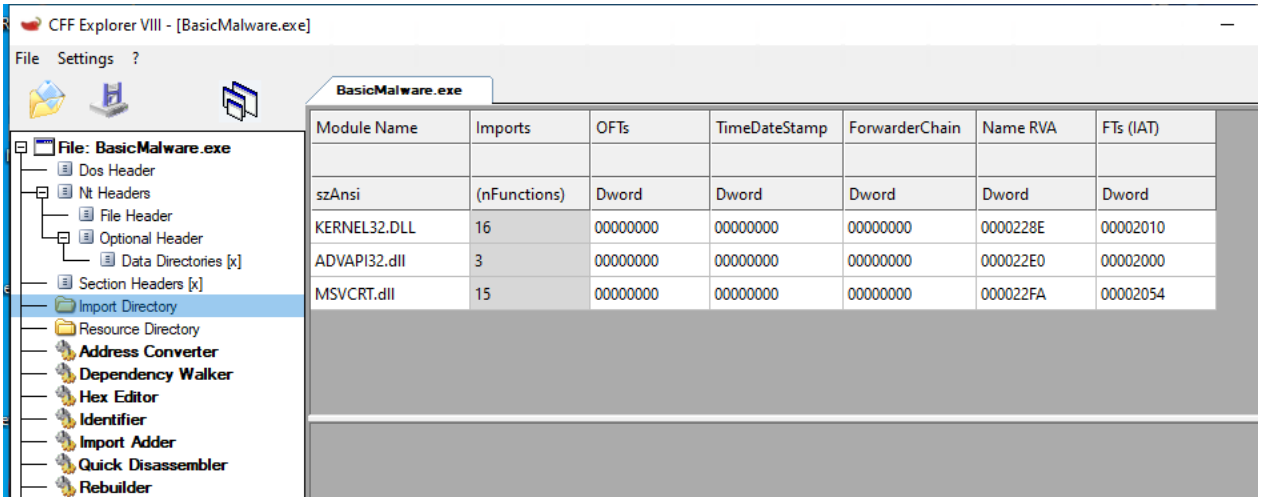
Hình 16: Kiểm tra pack của file

Ta mở PowerShell lên để giải nén file độc hại bằng câu lệnh như bên dưới



Hình 17: Umpack file độc hại

Sau đó ta kéo thả thư mục vào công cụ CFF Explorer để xem các thư viện được Import vào trong file thực thi.



Hình 18: CFF để lấy thư viện được import

Xong thì lấy các chuỗi có trong file, những chuỗi không bị mã hóa hoặc làm nhiễu bằng công cụ Strings. Ta chạy công cụ theo câu lệnh bên dưới để thu kết quả vào file output.txt.

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\minhh\Desktop> Strings .\BasicMalware.exe >>output.txt
PS C:\Users\minhh\Desktop>
```

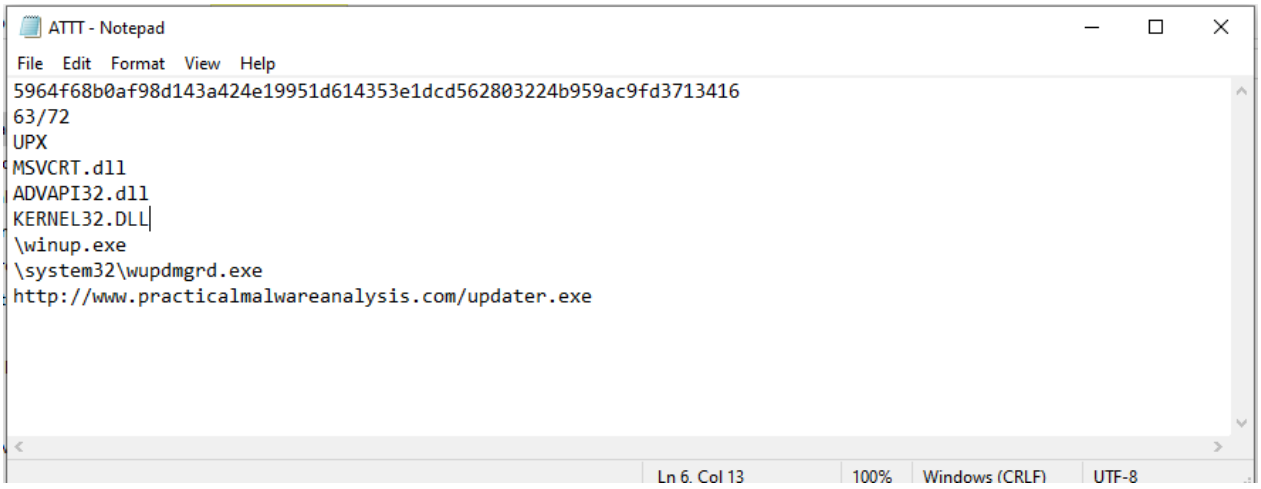
Hình 19: Dùng Strigns để lấy chuỗi của file

Sau đó mở file ra đọc thì thấy được những cảnh báo đáng ngờ như các file exe được gọi tới, đường dẫn lạ như bên dưới.

```
__set_app_type
_except_handler3
_controlfp
\winup.exe
%s%s
\system32\wupdmgrd.exe
%s%s
http://www.practicalmalwareanalysis.com/updater.exe
```

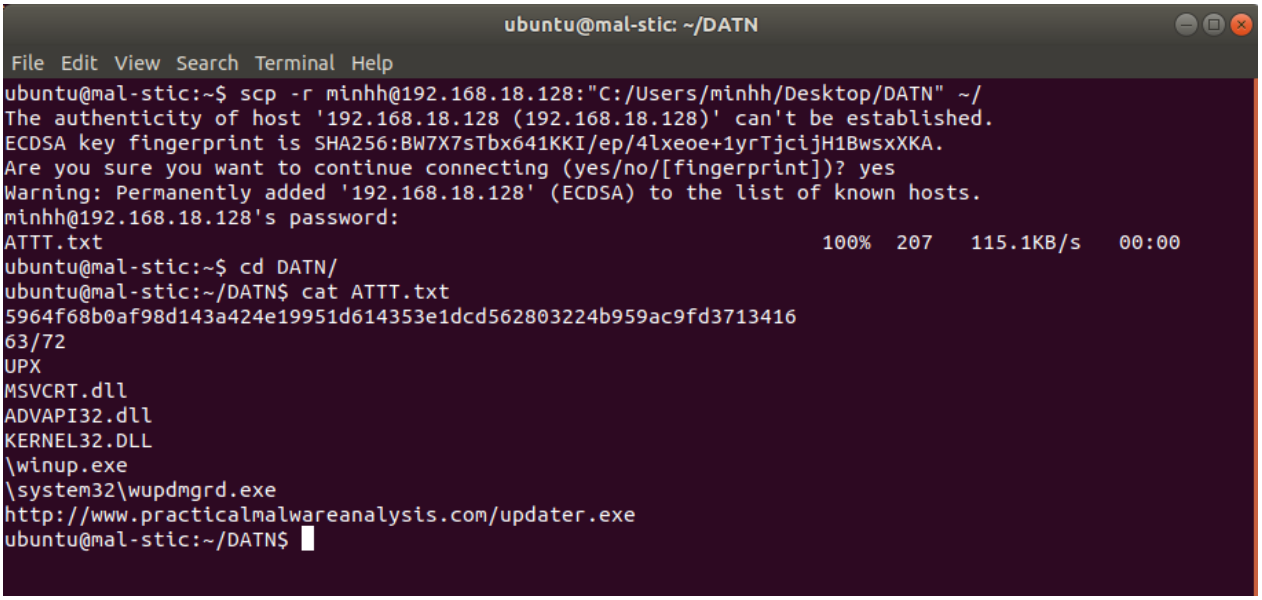
Hình 20: Các chuỗi cảnh báo IOC

Sau khi làm hết các nhiệm vụ thì ta lưu thông tin vào file ATTT.txt để kéo về bên máy Linux. Sau đó tiền hành cat file để check thông tin lần cuối.



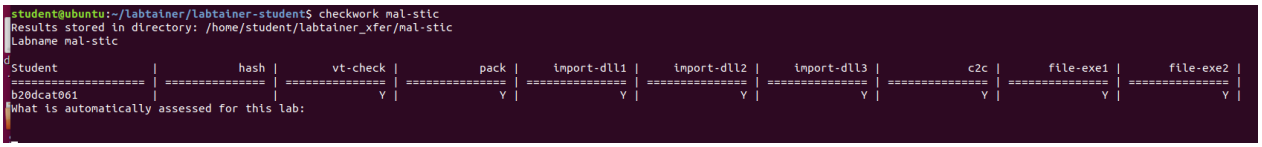
Hình 21: Điền các output theo yêu cầu

Ta dùng lệnh SCP để kéo về với IP, Username và Password đã được cho khi remote vào để thu thập thông tin về.



Hình 22: Copy file về máy ubuntu để checkwork

Sau khi kiểm tra lại và thấy thông tin không bị mất mát gì thì ta tiến hành checkwork để thu về kết quả.



Hình 23: Đánh giá kết quả bài thực hành

TÀI LIỆU THAM KHẢO

[1] Sikorski, M., & Honig, A. (2012). *Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software*.