

BÁO CÁO THỰC HÀNH

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

Tên chủ đề: Triển khai Snort Inline

GVHD: Đỗ Hoàng Hiến

Nhóm: 8

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT204.O21.ANTT.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Lê Thảo Ngọc	21521191	21521191@gm.uit.edu.vn
2	Trần Lê Minh Ngọc	21521195	21521195@gm.uit.edu.vn
3	Trần Văn Thái	21522583	21522583@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Nội dung	Tình trạng	Trang
1	Yêu cầu 1	100%	2 – 3
2	Yêu cầu 2	100%	3 – 22
3	Yêu cầu 3	100%	22 – 28
Điểm tự đánh giá			10/10

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

B.1 Tìm hiểu và sử dụng Snort

Yêu cầu 1: Sinh viên trả lời các câu hỏi bên dưới.

1.1a. Tìm hiểu về Snort? Snort cho phép chạy trên những chế độ (mode) nào?

1.1b. Trình bày những tính năng chính của Snort?

Trả lời:

1.1.a

- Snort là một hệ thống phát hiện xâm nhập (IDS - Intrusion Detection System) mã nguồn mở và một hệ thống ngăn chặn xâm nhập (IPS - Intrusion Prevention System). Nó được sử dụng để giám sát và phát hiện các hoạt động xâm nhập vào mạng máy tính, bao gồm các cuộc tấn công từ bên ngoài hoặc bên trong mạng. Nếu một cuộc tấn công được phát hiện bởi Snort thì nó có thể phản ứng bằng nhiều cách khác nhau phụ thuộc vào cấu hình mà bạn thiết lập, chẳng hạn 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 các gói tin đó. Tuy nhiên snort cũng có điểm yếu. Đó là tương tự như các bộ quét virus (virus scanner), 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.
- Snort có thể chạy trong các chế độ sau:
 - Sniffer Mode: Trong chế độ này, Snort hoạt động giống như một công cụ giám sát gói tin (packet sniffer), lắng nghe và ghi lại gói tin trên một giao diện mạng cụ thể nhưng không thực hiện bất kỳ xử lý hay phân tích nào trên dữ liệu ghi lại.
 - Packet Logger Mode: Ở chế độ này, Snort ghi lại tất cả hoặc một phần nhỏ của gói tin mà nó nhận được vào một tệp log, để sau này có thể phân tích.
 - Network Intrusion Detection Mode: Đây là chế độ chính của Snort. Trong chế độ này, Snort phân tích các gói tin mạng đến và ra khỏi một máy tính hoặc một mạng, xác định các mẫu hoặc dấu hiệu của các cuộc tấn công được định nghĩa trước và cảnh báo hoặc thực hiện các biện pháp ngăn chặn tương ứng.
 - Inline Mode: Trong chế độ này, Snort hoạt động như một hệ thống ngăn chặn xâm nhập (IPS), nghĩa là nó có khả năng chặn các gói tin độc hại trước khi chúng tiếp tục vào mạng nội bộ.

1.1.b. Tính năng chính của Snort

- Sniffer: Trong chế độ này, Snort hoạt động như một công cụ Sniffer, có khả năng lắng nghe và thu thập các gói tin mạng trên một giao diện mạng cụ thể. Snort không thực hiện bất kỳ xử lý hay phân tích nào trên dữ liệu ghi lại. Thay vào đó,

nó chỉ thu thập gói tin và truyền chúng đến các ứng dụng hoặc tệp lưu trữ. Chế độ Sniffer thường được sử dụng để kiểm tra hoặc ghi lại các luồng dữ liệu mạng cho mục đích kiểm tra và phân tích.

- Ghi Log (Logging): Snort có khả năng ghi lại các gói tin mạng đáng ngờ hoặc các sự kiện quan trọng vào các tệp log. Các tệp log này sau đó có thể được sử dụng để phân tích sâu hơn, điều tra sự cố, hoặc đánh giá cấp độ rủi ro trong mạng. Thông thường, các thông tin ghi log bao gồm thời gian xảy ra sự kiện, địa chỉ IP nguồn và đích, các giao thức và port sử dụng, cũng như mô tả chi tiết về các hoạt động phát hiện.
- Phát hiện xâm nhập (Intrusion Detection): Snort có khả năng phát hiện các hoạt động xâm nhập vào mạng máy tính thông qua việc phân tích các gói tin mạng đến và ra khỏi mạng. Sử dụng các luật (rules) dựa trên chữ ký và phân tích dựa trên hành vi, Snort nhận diện các mẫu hoặc dấu hiệu của các cuộc tấn công mạng. Khi phát hiện một mẫu tấn công được xác định, Snort có thể cảnh báo người quản trị bằng cách ghi log, hiển thị thông báo trên giao diện người dùng hoặc thậm chí ngăn chặn cuộc tấn công nếu nó được cấu hình để hoạt động trong chế độ IPS.

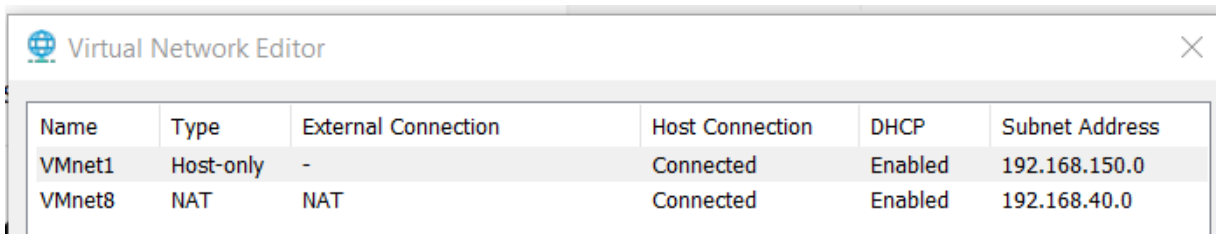
B.2 Cài đặt và cấu hình Snort để giám sát mạng

Lưu ý: Trong mô hình triển khai x là 2 số cuối MSSV của 1 thành viên trong nhóm.

Yêu cầu 2: Sinh viên cài đặt và cấu hình Snort Inline theo các bước bên dưới. Chụp lại các hình ảnh minh chứng (chụp full màn hình) cho từng bước làm.

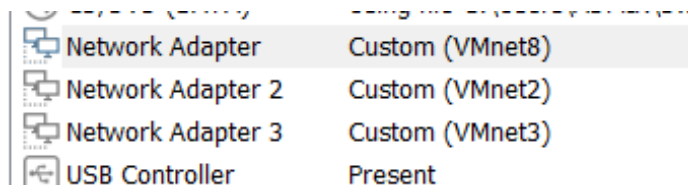
2.1a. Cấu hình mạng cho các máy theo mô hình

- Kiểm tra card VMnet8 (NAT) đã tồn tại và được bật DHCP



Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet1	Host-only	-	Connected	Enabled	192.168.150.0
VMnet8	NAT	NAT	Connected	Enabled	192.168.40.0

- Gán các card mạng cho máy **Router**



Network Adapter	Custom (VMnet8)
Network Adapter 2	Custom (VMnet2)
Network Adapter 3	Custom (VMnet3)
USB Controller	Present

- Gán các card mạng cho máy **Kali**



Network Adapter	Custom (VMnet2)
USB Controller	Present

- Gán các card mạng cho máy **Snort**

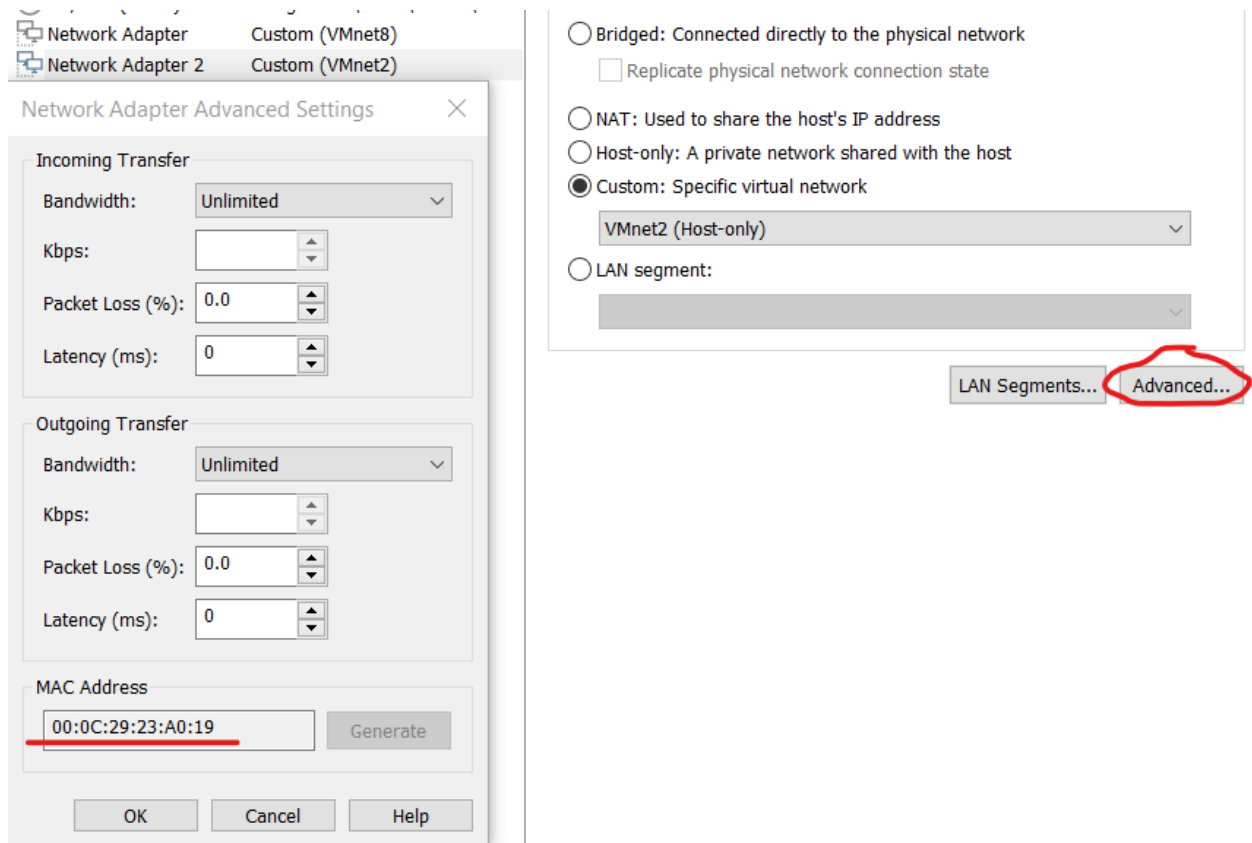
Network Adapter	Custom (VMnet8)
Network Adapter 2	Custom (VMnet3)
Network Adapter 3	Custom (VMnet4)
USB Controller	Present

- Gán các card mạng cho máy **Victim**

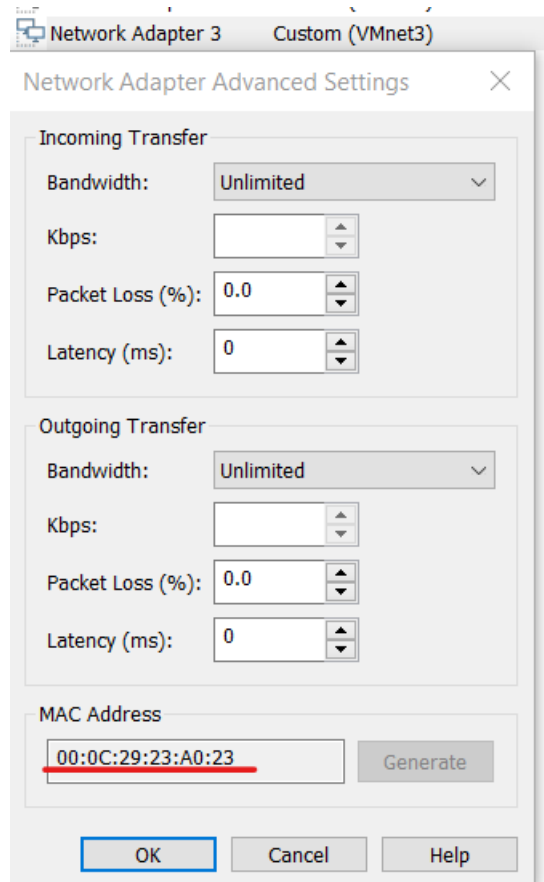
Network Adapter	Custom (VMnet4)
USB Controller	Present
Display	Auto detect

2.1b. Cấu hình địa chỉ ip cho các máy

- Máy **Router**
 - Thực hiện kiểm tra địa chỉ MAC của adapter tương ứng với interface cần cấu hình IP trước khi gán địa chỉ IP



⇒ Interface có địa chỉ MAC: 00:0C:29:23:A0:19 sẽ có IP: 10.81.95.1



⇒ Interface có địa chỉ MAC: 00:0C:29:23:A0:23 sẽ có IP: 192.168.95.1

- Vì ens34 có địa chỉ MAC của adapter VMnet2 nên gán cho nó IP thuộc lớp 10.81.95.0

```
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:23:a0:19 brd ff:ff:ff:ff:ff:ff
    altname enp2s2
```

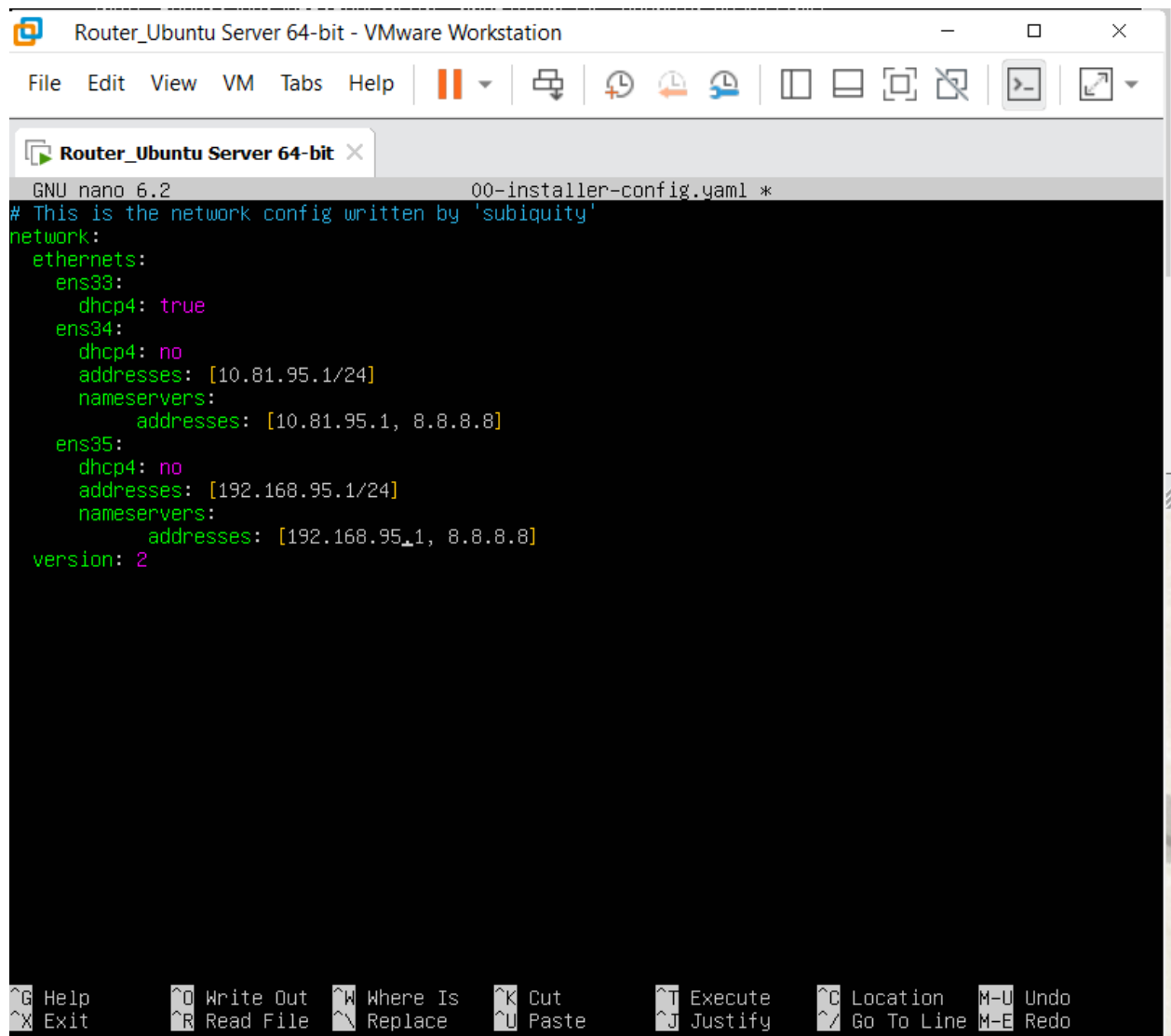
- Vì ens35 có địa chỉ MAC của adapter VMnet3 nên gán cho nó IP thuộc lớp 192.168.95.0

```
4: ens35: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:23:a0:23 brd ff:ff:ff:ff:ff:ff
    altname enp2s3
```

- Thực hiện lệnh: `cd /etc/netplan` để chuyển tới thư mục của netplan

```
root@serverrouter:/etc/netplan# ls
00-installer-config.yaml
```

- Nhập lệnh: `sudo nano {file_name}`. Để mở file cấu hình mạng



```
Router_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit x
GNU nano 6.2 00-installer-config.yaml *
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens33:
      dhcp4: true
    ens34:
      dhcp4: no
      addresses: [10.81.95.1/24]
      nameservers:
        addresses: [10.81.95.1, 8.8.8.8]
    ens35:
      dhcp4: no
      addresses: [192.168.95.1/24]
      nameservers:
        addresses: [192.168.95.1, 8.8.8.8]
  version: 2
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

- Thực hiện lệnh: `sudo netplan apply`. Để thiết lập cấu hình đã thực hiện.
- Thực hiện lệnh: `ifconfig` để kiểm tra lại IP của router

```

Router_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help

Router_Ubuntu Server 64-bit x Debian 11.x (kali linux) x

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.197.130 netmask 255.255.255.0 broadcast 192.168.197.255
    inet6 fe80::20c:29ff:fe23:a00f prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:23:a0:0f txqueuelen 1000 (Ethernet)
    RX packets 53701 bytes 60926622 (60.9 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11797 bytes 850890 (850.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens34: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.81.95.1 netmask 255.255.255.0 broadcast 10.81.95.255
    inet6 fe80::20c:29ff:fe23:a019 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:23:a0:19 txqueuelen 1000 (Ethernet)
    RX packets 23914 bytes 4065256 (4.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22526 bytes 2092747 (2.0 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens35: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.95.1 netmask 255.255.255.0 broadcast 192.168.95.255
    inet6 fe80::20c:29ff:fe23:a023 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:23:a0:23 txqueuelen 1000 (Ethernet)
    RX packets 5832 bytes 873362 (873.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4982 bytes 2475326 (2.4 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

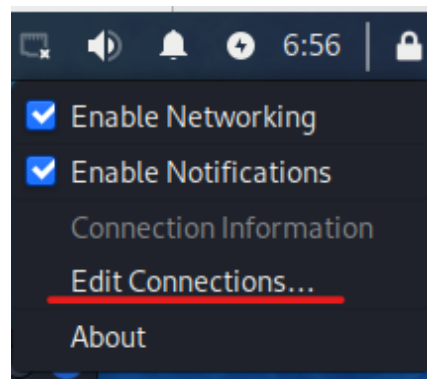
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1929 bytes 167138 (167.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1929 bytes 167138 (167.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@serverrouter:/etc/netplan# _

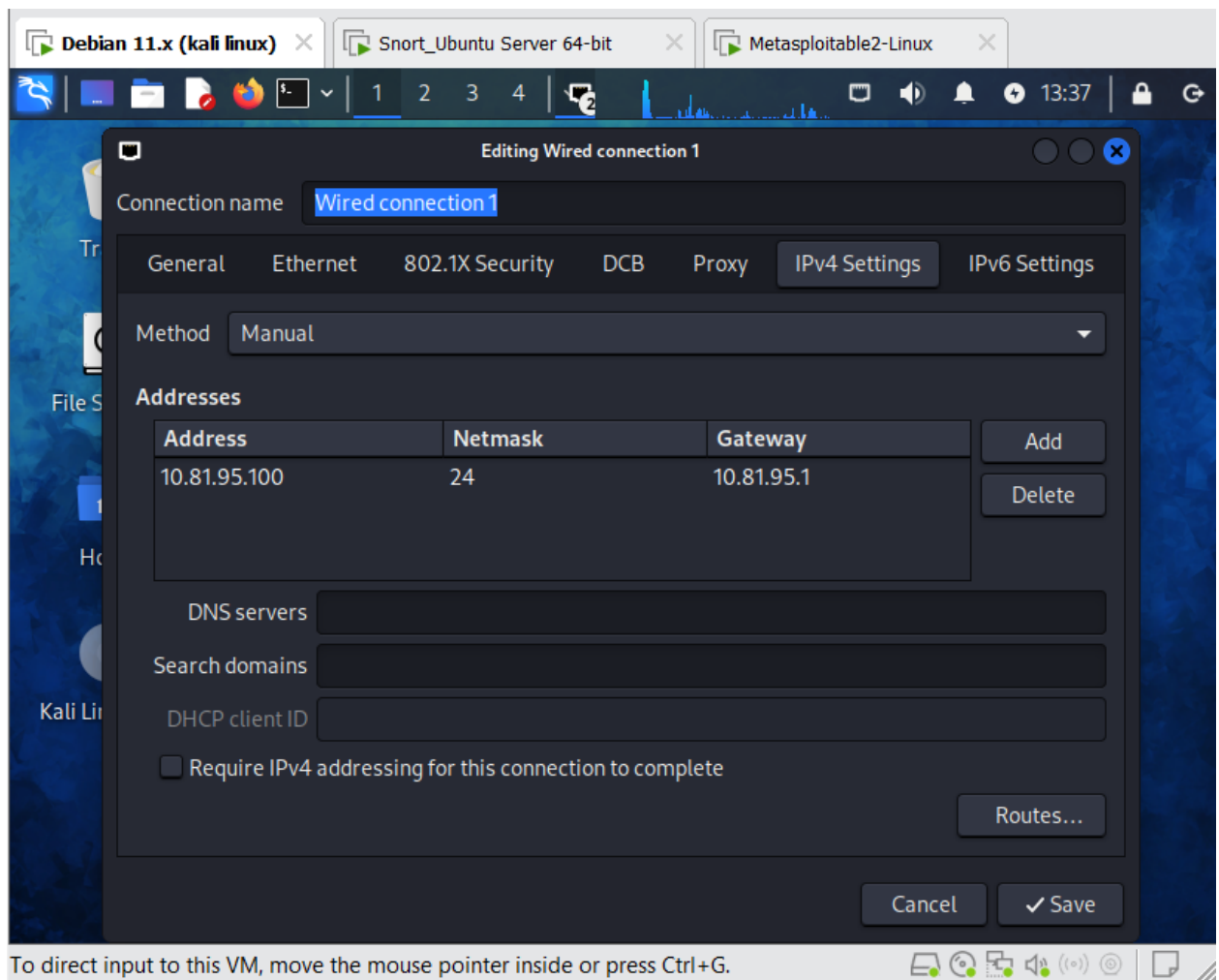
```

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

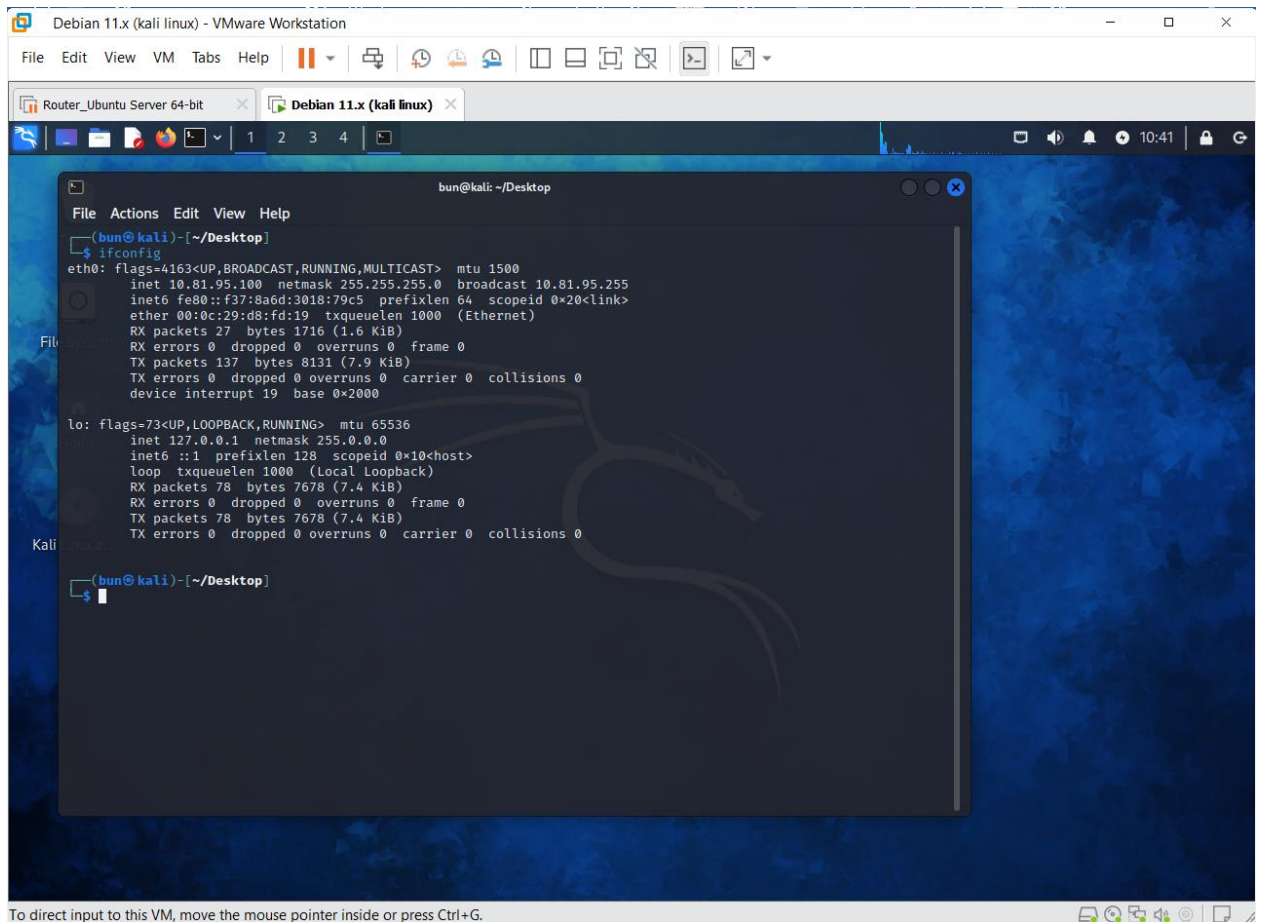
- Máy Kali
 - Vô mục **Edit Connections** để cấu hình mạng



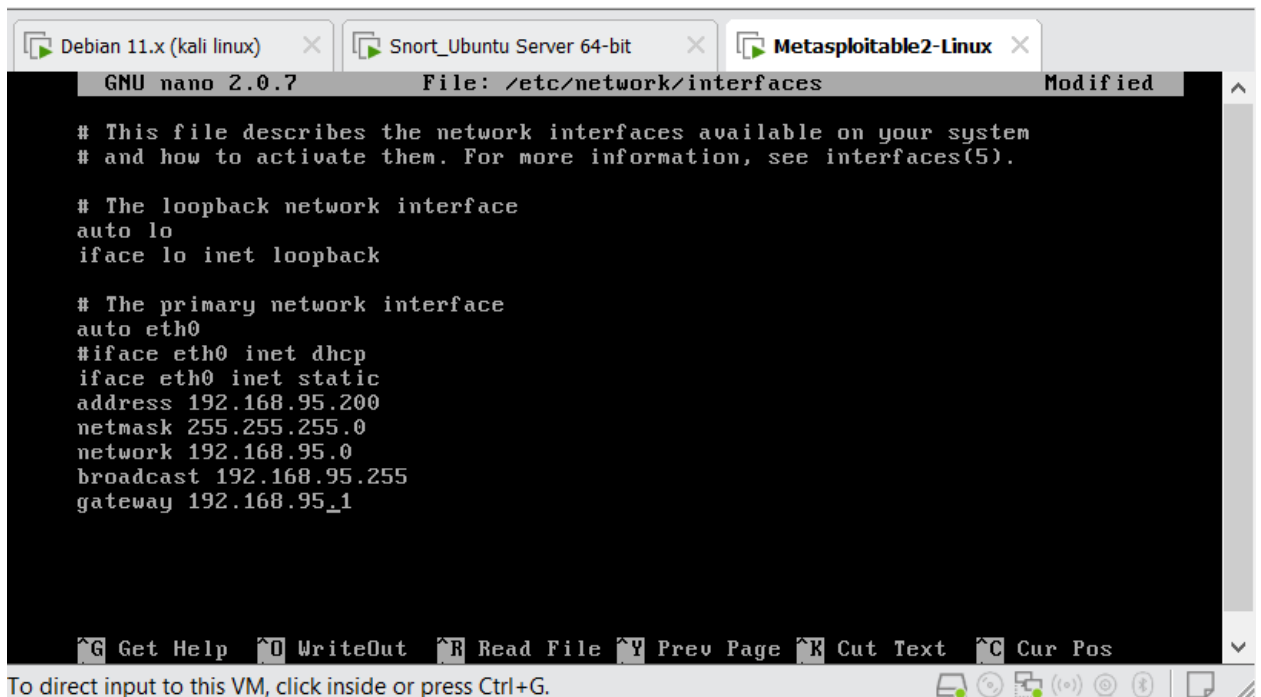
- Chọn tab Ipv4 Setting rồi tạo địa chỉ như mong muốn



Thực hiện kết nối lại vào mạng “Wired connection1”. Sau đó kiểm tra lại địa chỉ IP của máy



- Máy Victim
 - Nhập lệnh: `sudo nano /etc/network/interfaces`



- Thực hiện lệnh: `"sudo /etc/init.d/networking restart"` để thay đổi cấu hình IP

2.1c. Cấu hình NAT outbound cho máy router

- Chuyển qua chế độ root

```
server1@serverrouter:~$ sudo su
root@serverrouter:/home/server1# _
```

- Cấu hình NAT cho router theo hướng dẫn

```
Router_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit x Metasploitable2-Linux x Snort_Ubuntu Server 64-bit x
root@serverrouter:/etc/netplan# ping 10.81.95.100
PING 10.81.95.100 (10.81.95.100) 56(84) bytes of data:
64 bytes from 10.81.95.100: icmp_seq=1 ttl=64 time=0.376 ms
64 bytes from 10.81.95.100: icmp_seq=2 ttl=64 time=0.408 ms
64 bytes from 10.81.95.100: icmp_seq=3 ttl=64 time=0.602 ms
64 bytes from 10.81.95.100: icmp_seq=4 ttl=64 time=0.417 ms
64 bytes from 10.81.95.100: icmp_seq=5 ttl=64 time=0.423 ms
^C
--- 10.81.95.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.376/0.445/0.602/0.080 ms
root@serverrouter:/etc/netplan# cd /etc/netplan
root@serverrouter:/etc/netplan# ls
ls: command not found
root@serverrouter:/etc/netplan# ls
00-installer-config.yaml
root@serverrouter:/etc/netplan# ping 192.168.95.200
PING 192.168.95.200 (192.168.95.200) 56(84) bytes of data:
64 bytes from 192.168.95.200: icmp_seq=1 ttl=64 time=1.54 ms
64 bytes from 192.168.95.200: icmp_seq=2 ttl=64 time=0.821 ms
64 bytes from 192.168.95.200: icmp_seq=3 ttl=64 time=0.786 ms
64 bytes from 192.168.95.200: icmp_seq=4 ttl=64 time=0.664 ms
64 bytes from 192.168.95.200: icmp_seq=5 ttl=64 time=0.775 ms
^C
--- 192.168.95.200 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4077ms
rtt min/avg/max/mdev = 0.664/0.916/1.538/0.315 ms
root@serverrouter:/etc/netplan# iptables --flush
root@serverrouter:/etc/netplan# iptables --table nat --flush
root@serverrouter:/etc/netplan# iptables --delete-chain
root@serverrouter:/etc/netplan# iptables --table nat --delete-chain
root@serverrouter:/etc/netplan# iptables --append POSTROUTING --out-interface ens33 -j MASQUERADE
root@serverrouter:/etc/netplan# iptables --append FORWARD --in-interface ens37 -j ACCEPT
root@serverrouter:/etc/netplan# iptables --append FORWARD --in-interface ens38 -j ACCEPT
root@serverrouter:/etc/netplan# echo 1 > /proc/sys/net/ipv4/ip_forward
root@serverrouter:/etc/netplan# service iptables restart_
```

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

- Chi tiết mỗi câu lệnh

Delete and flush. Default table is "filter". Others like "nat" must be explicitly stated.

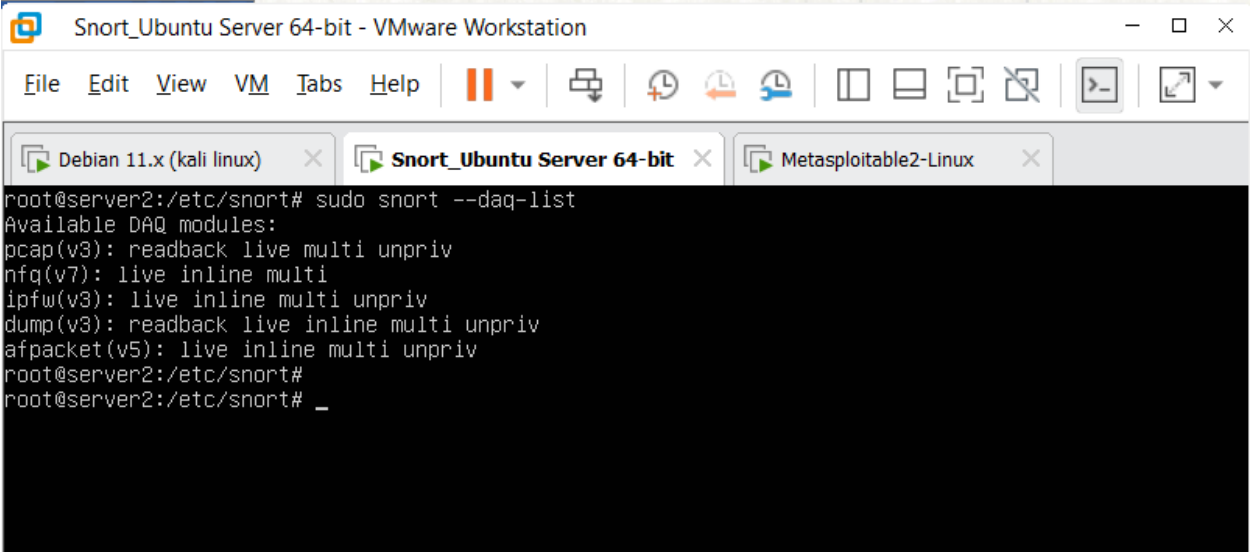
"iptables --flush": # Flush all the rules in filter and nat tables

```
root@serverrouter:/etc/netplan# iptables --flush
root@serverrouter:/etc/netplan# iptables --table nat --flush
root@serverrouter:/etc/netplan# iptables --delete-chain
```

Delete all chains that are not in default filter and nat table

```
root@serverrouter:/etc/netplan# iptables --table nat --delete-chain
```


- Kiểm tra afpacket DAQ đã được cài đặt chưa (để sử dụng được mode inline).



```
Snort_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help
Debian 11.x (kali linux) Snort_Ubuntu Server 64-bit Metasploitable2-Linux
root@server2:/etc/snort# sudo snort --daq-list
Available DAQ modules:
pcap(v3): readback live multi unpriv
nfq(v7): live inline multi
ipfw(v3): live inline multi unpriv
dump(v3): readback live inline multi unpriv
afpacket(v5): live inline multi unpriv
root@server2:/etc/snort#
root@server2:/etc/snort# _
```

- Xóa tất cả các file rule mặc định của Snort .

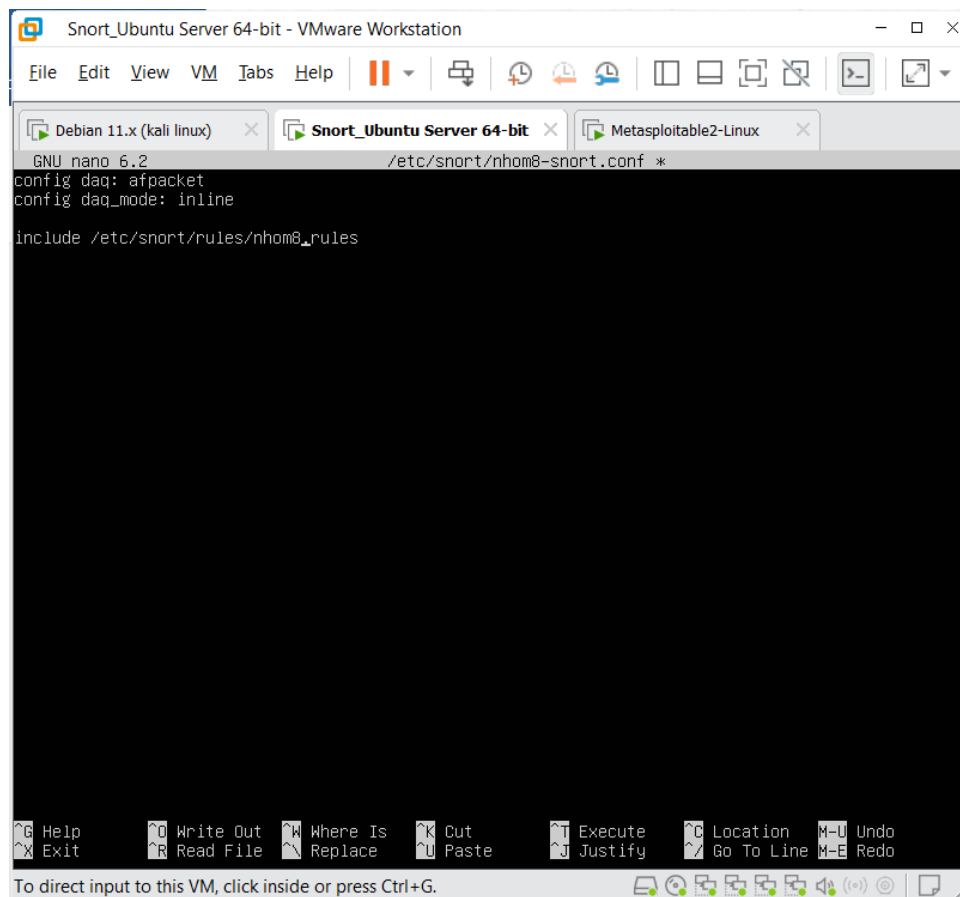
```
root@server2:/etc/snort#
root@server2:/etc/snort# sudo rm -rf /etc/snort/rules/*
```

- Tạo file rule của nhóm định nghĩa. Ví dụ ở đây là nhóm 0.

```
root@server2:/etc/snort# sudo touch /etc/snort/rules/nhom0.rules
```

- Tạo file cấu hình snort của nhóm tại `/etc/snort/nhomX-snort.conf` (với X là số thứ tự của nhóm) với nội dung như bên dưới để bật mode inline.
- Tạo file với nano:

```
root@server2:/etc/snort# sudo nano /etc/snort/nhom0-snort.conf
```



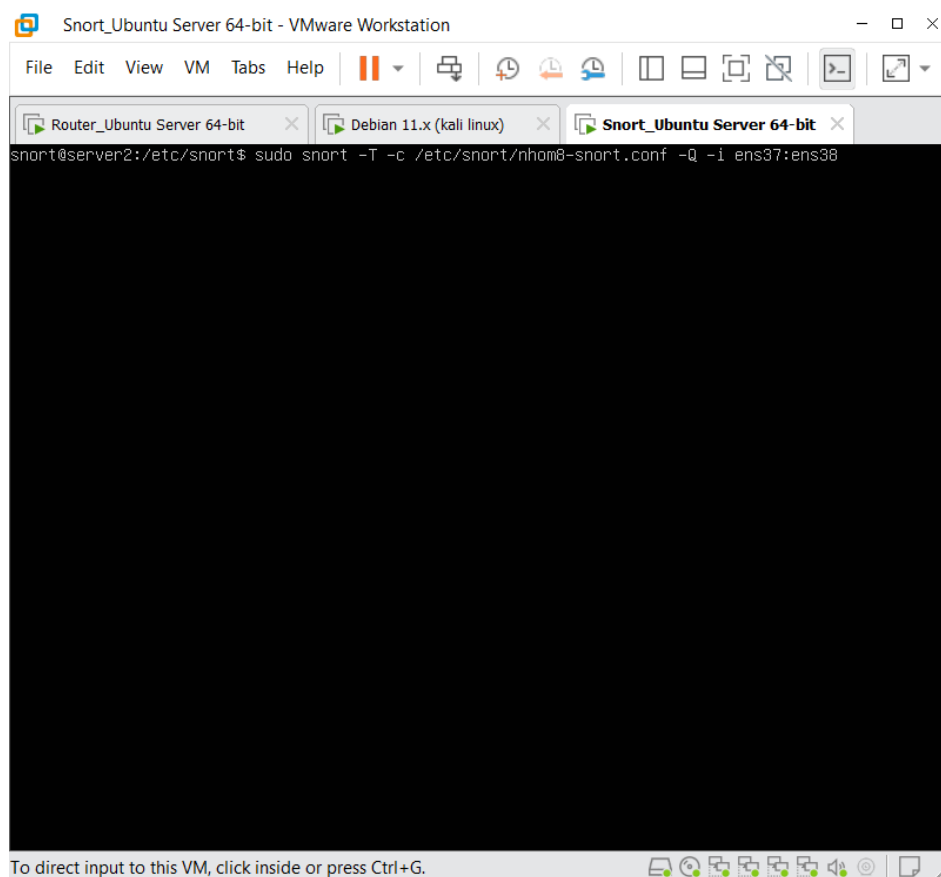
The screenshot shows a VMware Workstation window titled "Snort_Ubuntu Server 64-bit - VMware Workstation". The main window displays the nano text editor editing the file "/etc/snort/nhom8-snort.conf". The content of the file is as follows:

```
GNU nano 6.2 /etc/snort/nhom8-snort.conf *
config daq: afpacket
config daq_mode: inline

include /etc/snort/rules/nhom8_rules
```

The bottom status bar of the nano editor shows various shortcuts: Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, M-U, Undo, M-E, Redo.

- Kiểm tra file cấu hình snort bằng lệnh sau:

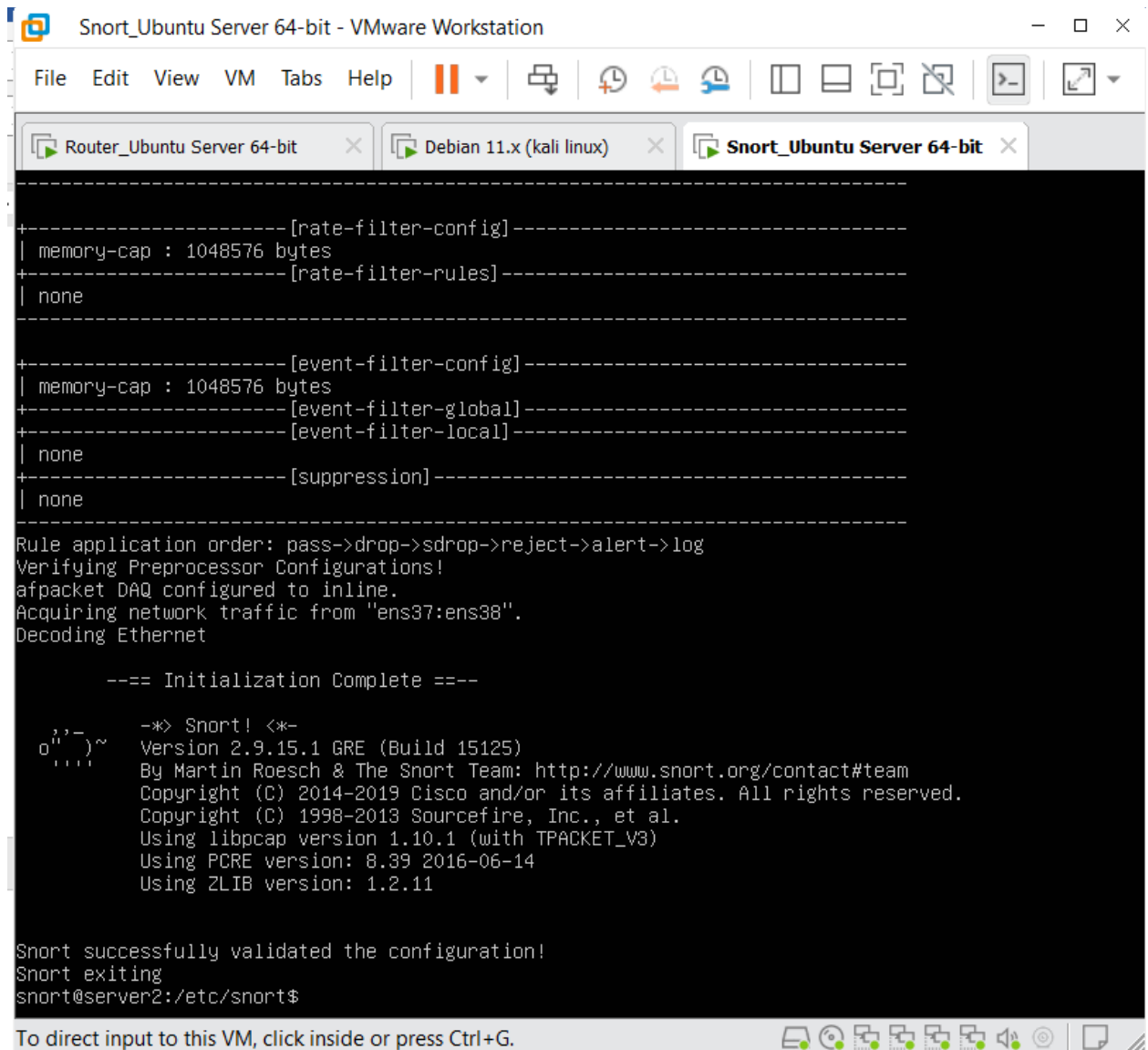


The screenshot shows a VMware Workstation window titled "Snort_Ubuntu Server 64-bit - VMware Workstation". The main window displays a terminal window with the following command and output:

```
snort@server2:/etc/snort$ sudo snort -T -c /etc/snort/nhom8-snort.conf -Q -i ens37:ens38
```

The terminal output is currently blank, indicating that the command has been executed but the results have not yet appeared.

- Kết quả cài đặt thành công



```
-----[rate-filter-config]-----
| memory-cap : 1048576 bytes
-----[rate-filter-rules]-----
| none
-----

+-----[event-filter-config]-----
| memory-cap : 1048576 bytes
+-----[event-filter-global]-----
+-----[event-filter-local]-----
| none
+-----[suppression]-----
| none
-----

Rule application order: pass->drop->sdrop->reject->alert->log
Verifying Preprocessor Configurations!
afpacket DAQ configured to inline.
Acquiring network traffic from "ens37:ens38".
Decoding Ethernet

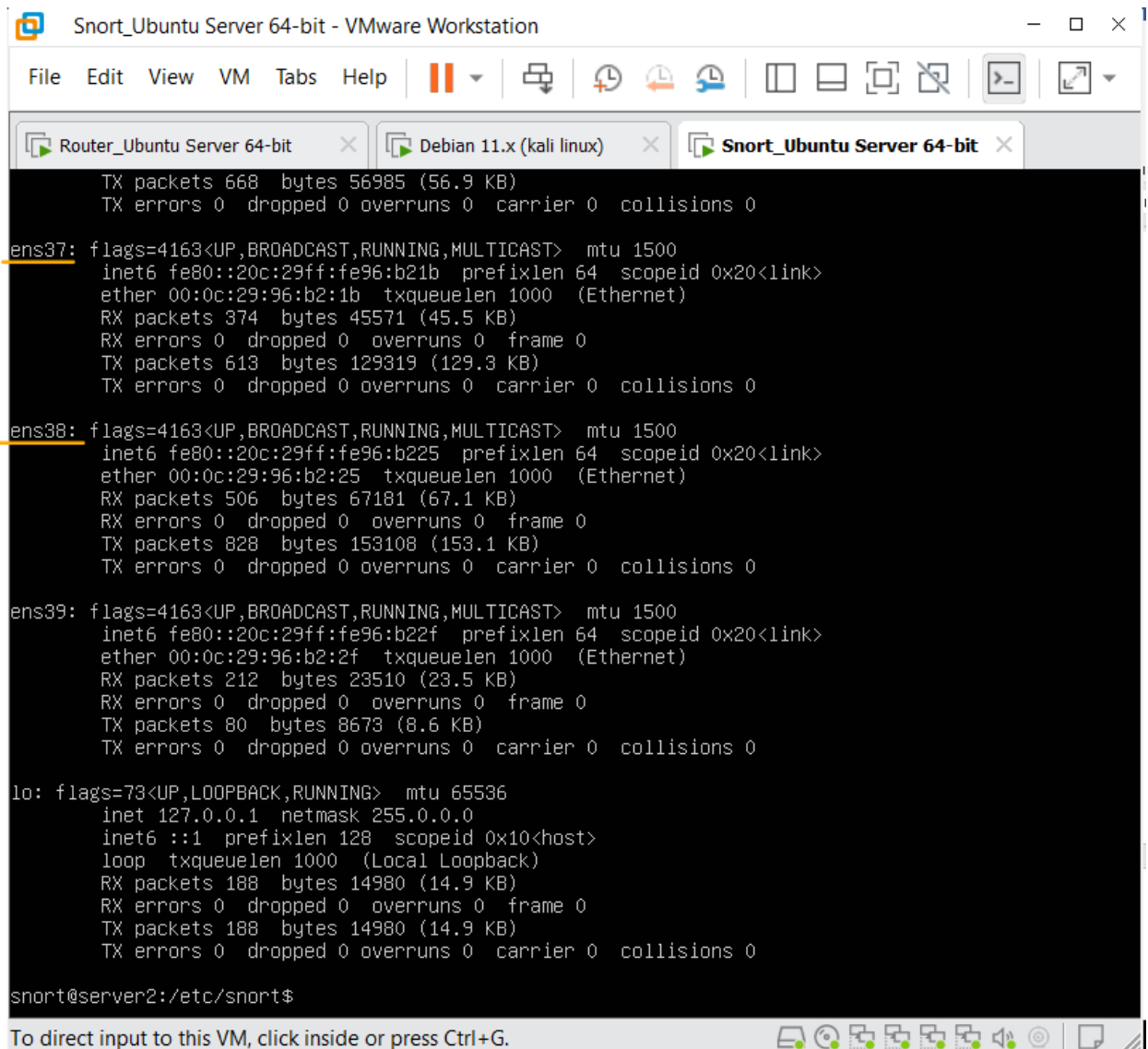
--== Initialization Complete ==--

o''~
'''~
-*) Snort! <*-
Version 2.9.15.1 GRE (Build 15125)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.10.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

Snort successfully validated the configuration!
Snort exiting
snort@server2:/etc/snort$

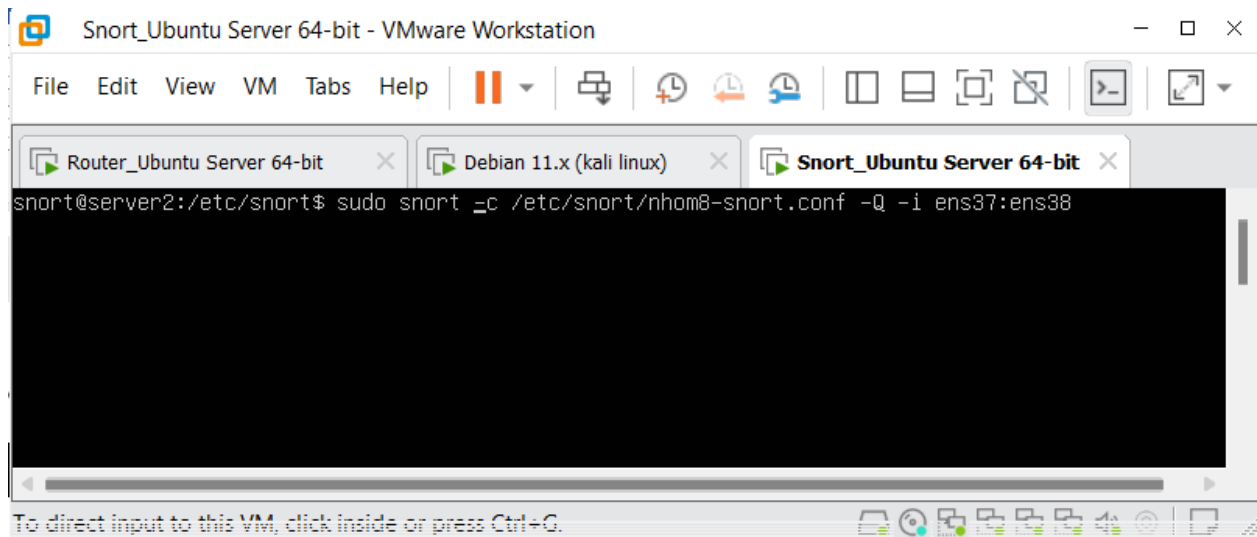
To direct input to this VM, click inside or press Ctrl+G.
```

*ens37 và ens38 là các interface của máy snort kết nối với VMnet2 và VMnet3



```
Snort_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit Debian 11.x (kali linux) Snort_Ubuntu Server 64-bit
TX packets 668 bytes 56985 (56.9 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet6 fe80::20c:29ff:fe96:b21b prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:96:b2:1b txqueuelen 1000 (Ethernet)
RX packets 374 bytes 45571 (45.5 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 613 bytes 129319 (129.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
ens38: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet6 fe80::20c:29ff:fe96:b225 prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:96:b2:25 txqueuelen 1000 (Ethernet)
RX packets 506 bytes 67181 (67.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 828 bytes 153108 (153.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
ens39: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet6 fe80::20c:29ff:fe96:b22f prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:96:b2:2f txqueuelen 1000 (Ethernet)
RX packets 212 bytes 23510 (23.5 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 80 bytes 8673 (8.6 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 188 bytes 14980 (14.9 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 188 bytes 14980 (14.9 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
snort@server2:/etc/snort$
To direct input to this VM, click inside or press Ctrl+G.
```

- Chạy snort trong mode inline với dòng lệnh sau:



- Kết quả:

```

-----[rate-filter-config]-----
| memory-cap : 1048576 bytes
-----[rate-filter-rules]-----
| none
-----[event-filter-config]-----
| memory-cap : 1048576 bytes
-----[event-filter-global]-----
-----[event-filter-local]-----
| none
-----[suppression]-----
| none
-----
Rule application order: pass->drop->sdrop->reject->alert->log
Verifying Preprocessor Configurations!
afpacket DAQ configured to inline.
Acquiring network traffic from "ens37:ens38".
Reload thread starting...
Reload thread started, thread 0x7f85ba5e8640 (3167)

--== Initialization Complete ==--

o''~)~
  ''')~
  ''')~

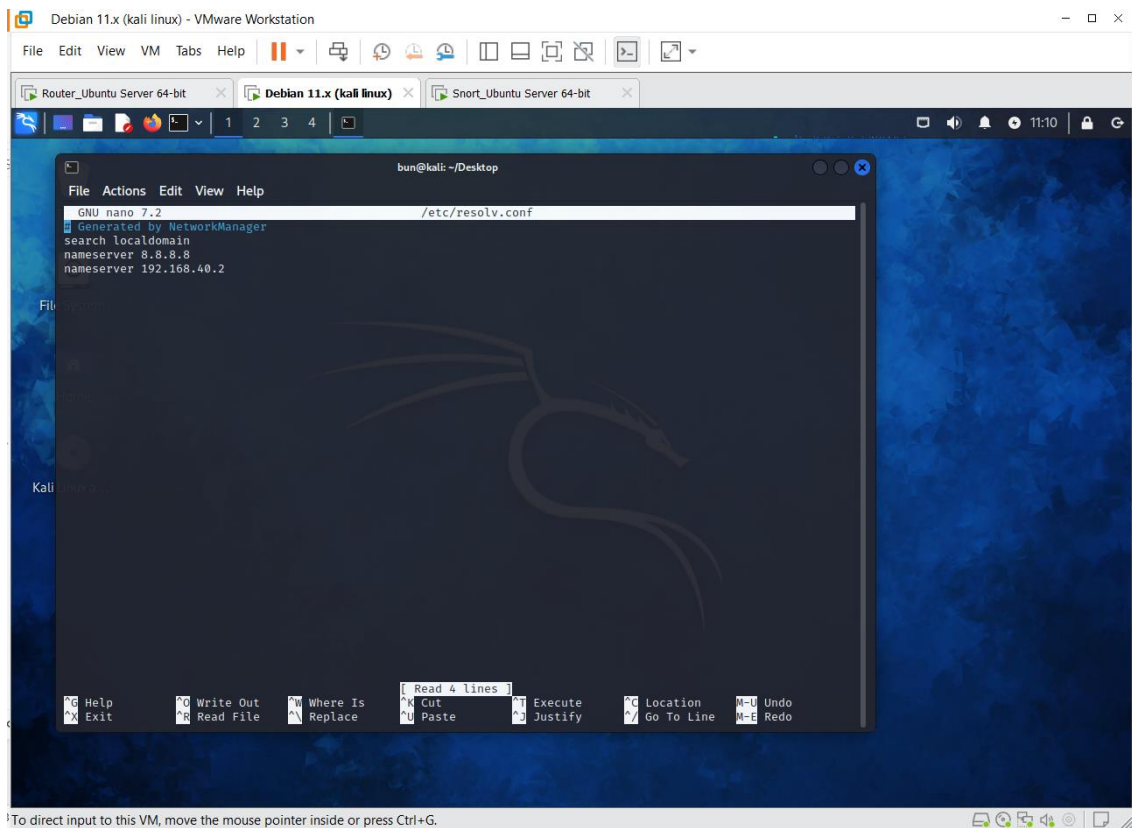
-*> Snort! <*-
Version 2.9.15.1 GRE (Build 15125)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.10.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

Commencing packet processing (pid=3158)
Decoding Ethernet

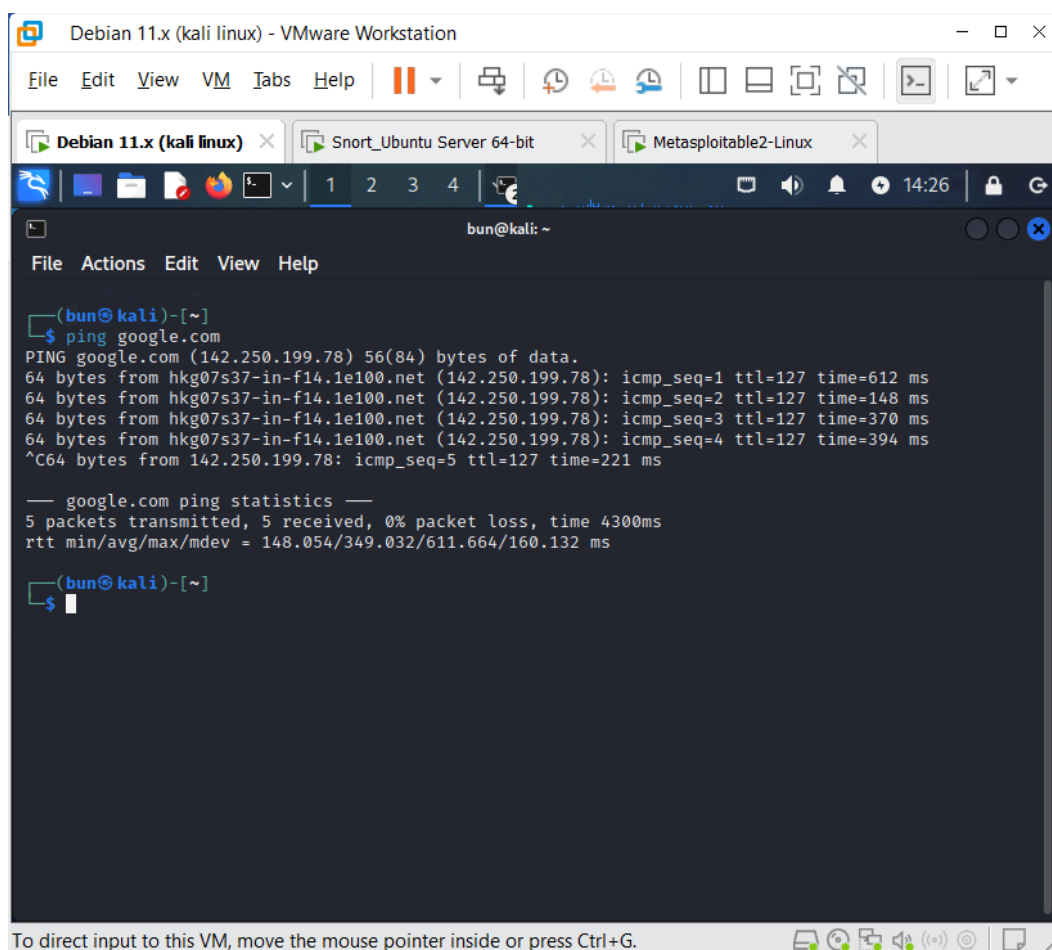
```

- Sau khi chạy thành công, kiểm tra kết nối của các máy.
 - Máy Kali ping google.com

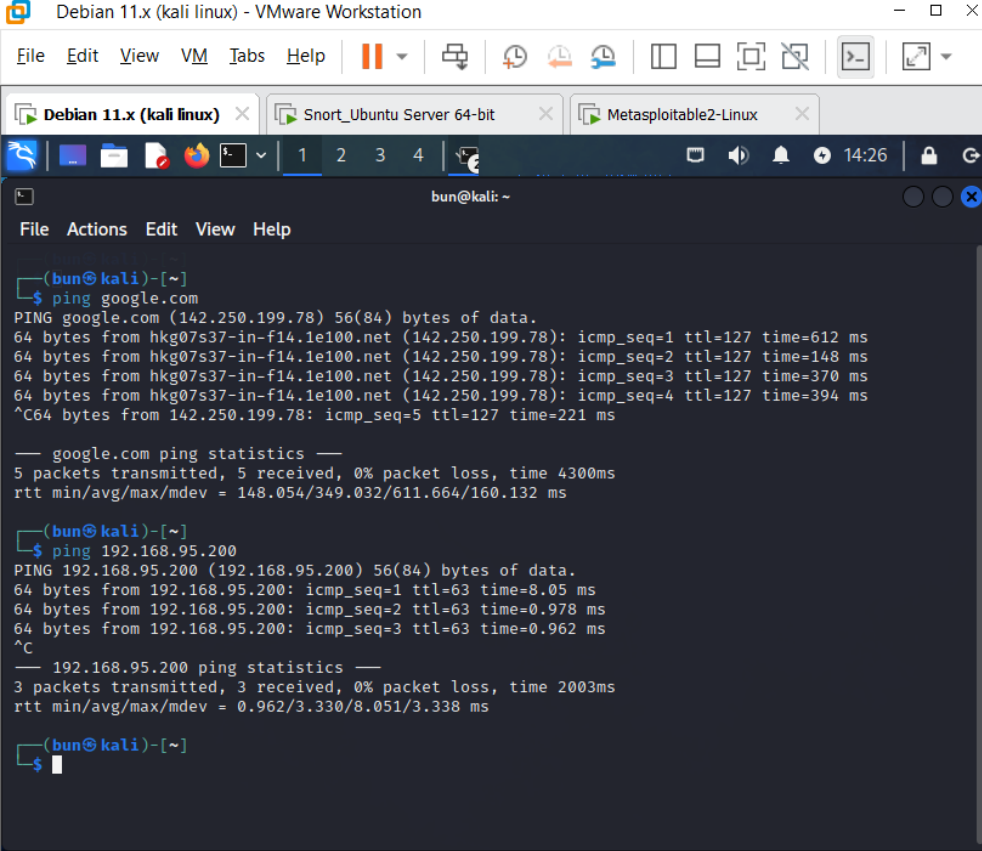
Để ping được tới domain thì chúng ta cần thêm nameserver của google vào file resolv.conf



Thực hiện ping tới domain



- Máy Kali ping máy Victim



```
Debian 11.x (kali linux) - VMware Workstation
File Edit View VM Tabs Help
Debian 11.x (kali linux) x Snort_Ubuntu Server 64-bit x Metasploitable2-Linux x
bun@kali: ~
File Actions Edit View Help
(bun@kali)~[~]
$ ping google.com
PING google.com (142.250.199.78) 56(84) bytes of data.
64 bytes from hkg07s37-in-f14.1e100.net (142.250.199.78): icmp_seq=1 ttl=127 time=612 ms
64 bytes from hkg07s37-in-f14.1e100.net (142.250.199.78): icmp_seq=2 ttl=127 time=148 ms
64 bytes from hkg07s37-in-f14.1e100.net (142.250.199.78): icmp_seq=3 ttl=127 time=370 ms
64 bytes from hkg07s37-in-f14.1e100.net (142.250.199.78): icmp_seq=4 ttl=127 time=394 ms
^C64 bytes from 142.250.199.78: icmp_seq=5 ttl=127 time=221 ms

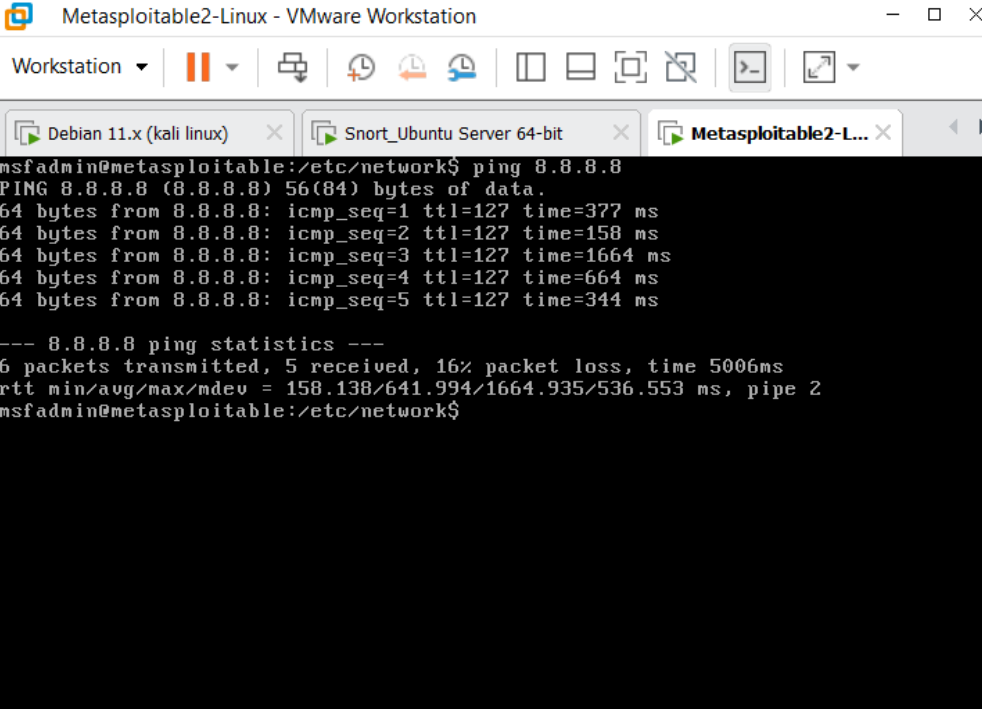
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4300ms
rtt min/avg/max/mdev = 148.054/349.032/611.664/160.132 ms

(bun@kali)~[~]
$ ping 192.168.95.200
PING 192.168.95.200 (192.168.95.200) 56(84) bytes of data.
64 bytes from 192.168.95.200: icmp_seq=1 ttl=63 time=8.05 ms
64 bytes from 192.168.95.200: icmp_seq=2 ttl=63 time=0.978 ms
64 bytes from 192.168.95.200: icmp_seq=3 ttl=63 time=0.962 ms
^C
--- 192.168.95.200 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.962/3.330/8.051/3.338 ms

(bun@kali)~[~]
$
```

- Máy Victim ping google.com

Ping thành công tới nameserver của google



```
Metasploitable2-Linux - VMware Workstation
Workstation
Debian 11.x (kali linux) x Snort_Ubuntu Server 64-bit x Metasploitable2-L... x
msfadmin@metasploitable:/etc/network$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=127 time=377 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=127 time=158 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=127 time=1664 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=127 time=664 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=127 time=344 ms

--- 8.8.8.8 ping statistics ---
6 packets transmitted, 5 received, 16% packet loss, time 5006ms
rtt min/avg/max/mdev = 158.138/641.994/1664.935/536.553 ms, pipe 2
msfadmin@metasploitable:/etc/network$
```

Tuy nhiên để ping tới domain google.com thì cần thêm nameserver 8.8.8.8 vào file resolv.conf.

Dùng nano để chỉnh sửa file

```
msfadmin@metasploitable:~$ sudo nano /etc/resolv.conf
```

Thêm nameserver của google vào file resolv.conf

```
GNU nano 2.0.7 File: /etc/resolv.conf
search localdomain
nameserver 8.8.8.8
nameserver 192.168.40.2
```

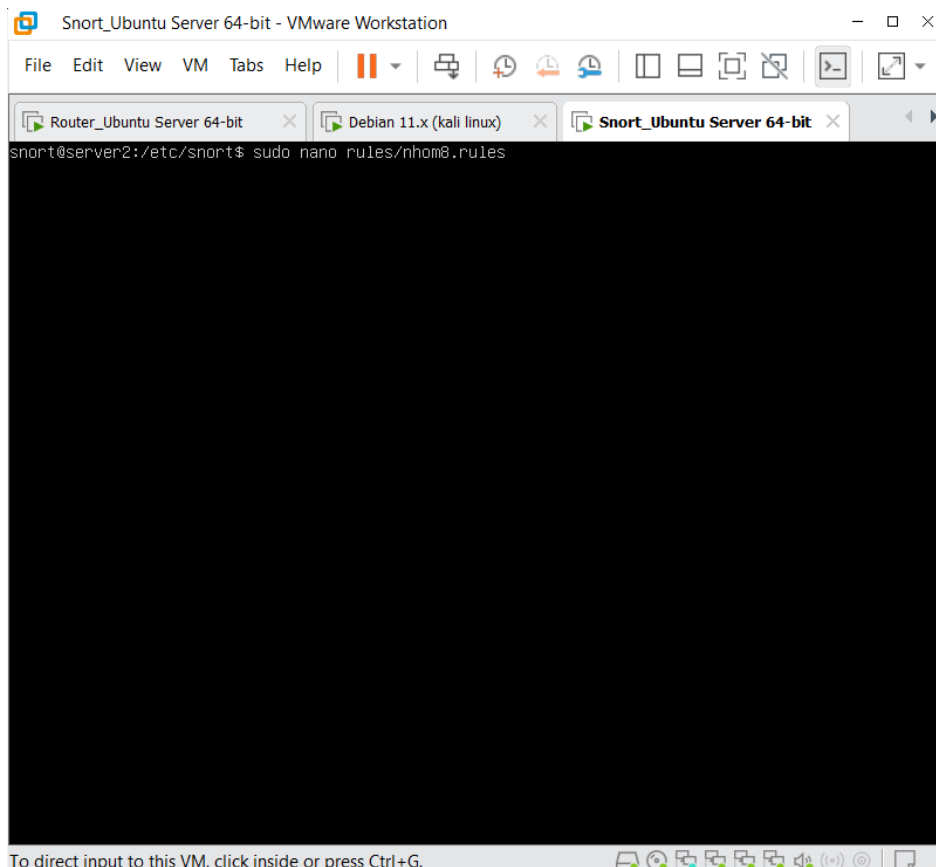
Thực hiện ping lại tới domain google.com

```
msfadmin@metasploitable:~$ ping google.com
PING google.com (172.217.27.46) 56(84) bytes of data:
64 bytes from hkg12s37-in-f14.1e100.net (172.217.27.46): icmp_seq=1 ttl=127 time=45.4 ms
64 bytes from sin11s03-in-f46.1e100.net (172.217.27.46): icmp_seq=2 ttl=127 time=58.9 ms
64 bytes from hkg12s37-in-f14.1e100.net (172.217.27.46): icmp_seq=3 ttl=127 time=60.7 ms

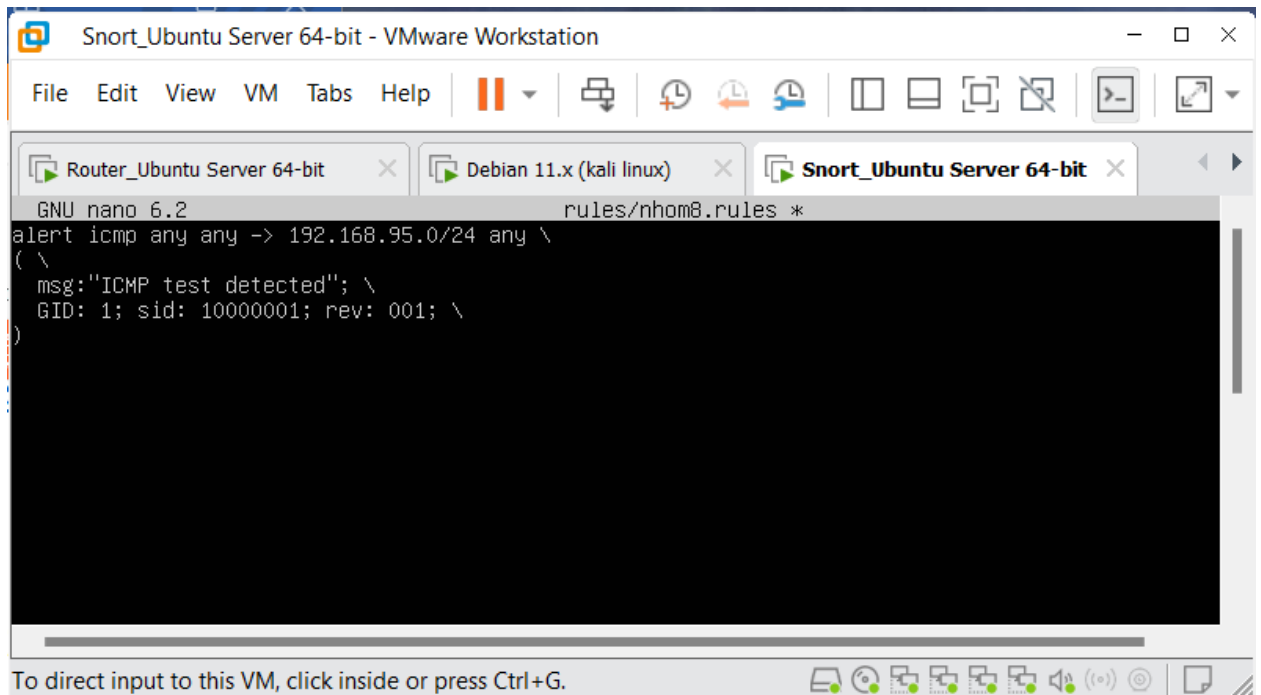
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 45.497/55.060/60.748/6.802 ms
msfadmin@metasploitable:~$
```

2.1e. Viết rule cho Snort

Dùng nano để viết rule trong /etc/snort/rules/nhom8.rules

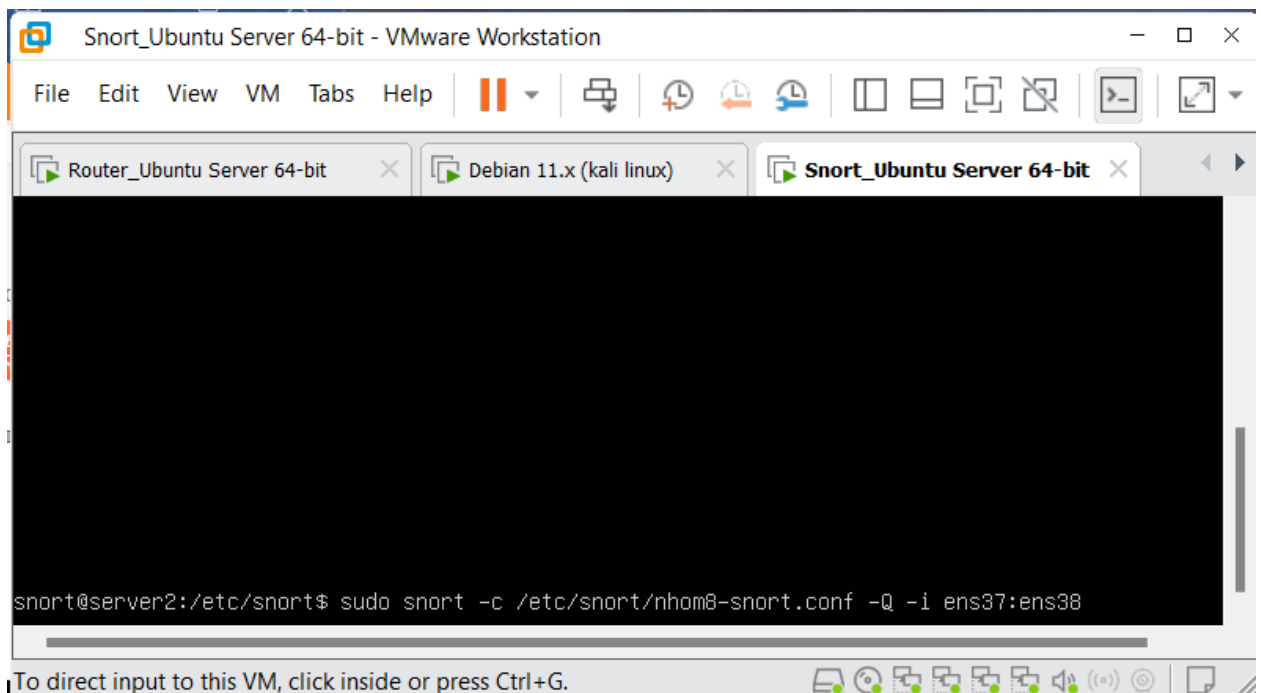


Viết rule phát hiện gói ICMP gửi đến lớp mạng 192.168.x.0/24 trong file `/etc/snort/rules/nhomX.rules` như sau:

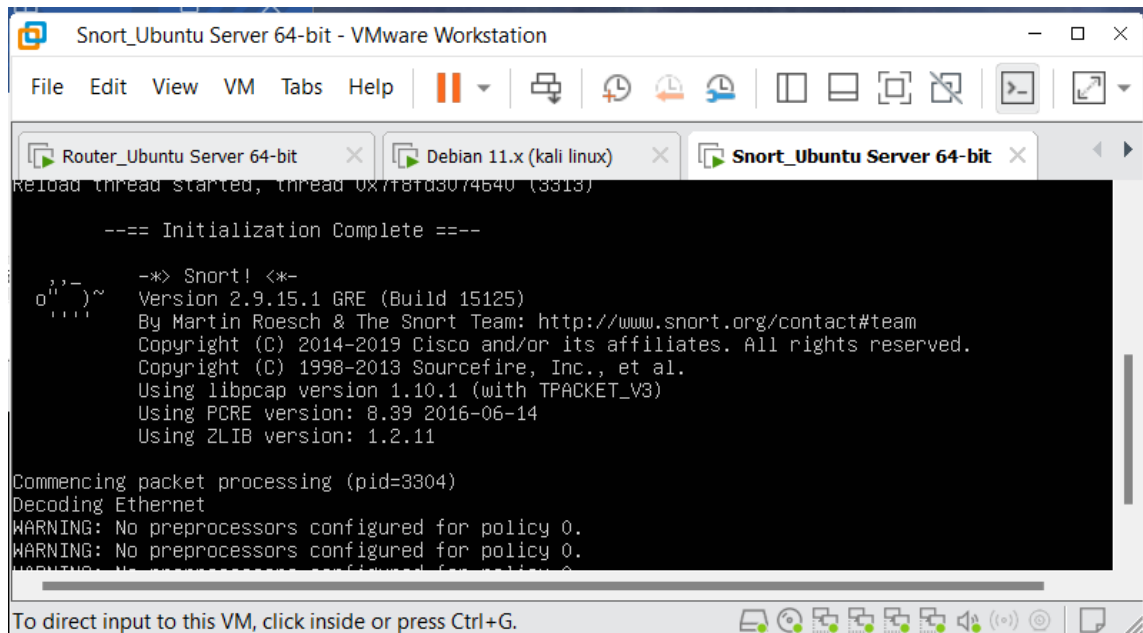


```
GNU nano 6.2 rules/nhom8.rules *
alert icmp any any -> 192.168.95.0/24 any \
( \
  msg:'ICMP test detected'; \
  GID: 1; sid: 10000001; rev: 001; \
)
```

- Chạy snort trong mode inline



```
snort@server2:/etc/snort$ sudo snort -c /etc/snort/nhom8-snort.conf -Q -i ens37:ens38
```



```
Snort_Ubuntu Server 64-bit - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit Debian 11.x (kali linux) Snort_Ubuntu Server 64-bit
Reload thread started, thread 0x7f8fd3074b40 (3313)

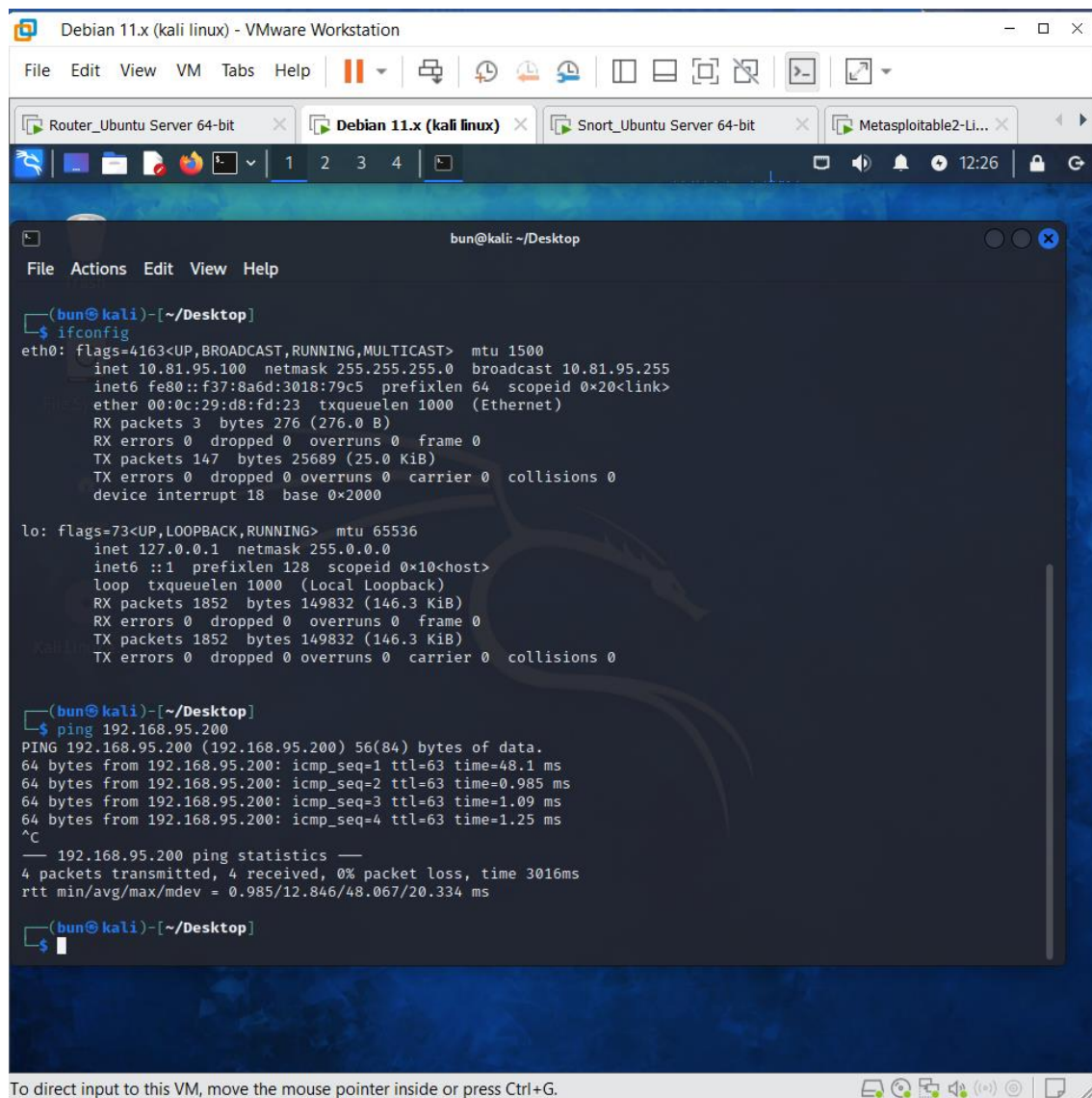
---- Initialization Complete ----

--> Snort! <*-
Version 2.9.15.1 GRE (Build 15125)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.10.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

Commencing packet processing (pid=3304)
Decoding Ethernet
WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.

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

Thực hiện ping từ máy kali tới máy victim



```
Debian 11.x (kali linux) - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit Debian 11.x (kali linux) Snort_Ubuntu Server 64-bit Metasploitable2-Li...
bun@kali: ~/Desktop

(bun@kali)~[/Desktop]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.81.95.100 netmask 255.255.255.0 broadcast 10.81.95.255
    inet6 fe80::f37:8a6d:3018:79c5 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:d8:fd:23 txqueuelen 1000 (Ethernet)
    RX packets 3 bytes 276 (276.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 147 bytes 25689 (25.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 18 base 0x2000

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1852 bytes 149832 (146.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1852 bytes 149832 (146.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(bun@kali)~[/Desktop]
$ ping 192.168.95.200
PING 192.168.95.200 (192.168.95.200) 56(84) bytes of data:
64 bytes from 192.168.95.200: icmp_seq=1 ttl=63 time=48.1 ms
64 bytes from 192.168.95.200: icmp_seq=2 ttl=63 time=0.985 ms
64 bytes from 192.168.95.200: icmp_seq=3 ttl=63 time=1.09 ms
64 bytes from 192.168.95.200: icmp_seq=4 ttl=63 time=1.25 ms
^C
--- 192.168.95.200 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3016ms
rtt min/avg/max/mdev = 0.985/12.846/48.067/20.334 ms

(bun@kali)~[/Desktop]
$
```

Kiểm tra log của snort trên `/var/log/snort/alert`.

```

GNU nano 6.2 /var/log/snort/alert
[**] [4:2:1] SYN Flood to port 443 [**]
[Priority: 0]
03/03-15:40:47.368087 10.81.20.100:55698 -> 192.168.20.200:443
TCP TTL:56 TOS:0x0 ID:28492 IpLen:20 DgmLen:44
*****S* Seq: 0xD4A6588F Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [4:2:1] SYN Flood to port 443 [**]
[Priority: 0]
03/03-15:40:53.963469 10.81.20.100:55954 -> 192.168.20.200:443
TCP TTL:50 TOS:0x0 ID:41059 IpLen:20 DgmLen:44
*****S* Seq: 0x29EE0D60 Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [4:2:1] SYN Flood to port 443 [**]
[Priority: 0]
03/03-15:40:55.064299 10.81.20.100:55956 -> 192.168.20.200:443
TCP TTL:38 TOS:0x0 ID:6987 IpLen:20 DgmLen:44
*****S* Seq: 0x29EC0D62 Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [4:2:1] SYN Flood to port 443 [**]
[Priority: 0]
03/03-15:52:54.694435 10.81.20.100:48247 -> 192.168.20.200:443
TCP TTL:56 TOS:0x0 ID:31778 IpLen:20 DgmLen:44
*****S* Seq: 0x4CE2E644 Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [4:2:1] SYN Flood to port 443 [**]
[Priority: 0]
03/03-15:52:56.697178 10.81.20.100:48249 -> 192.168.20.200:443
TCP TTL:46 TOS:0x0 ID:49823 IpLen:20 DgmLen:44
*****S* Seq: 0x4CE0E646 Ack: 0x0 Win: 0x400 TcpLen: 24
[ File '/var/log/snort/alert' is unwritable ]

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
  
```

Yêu cầu 3: Sinh viên viết rule drop các gói ICMP đi đến máy **Victim** (rule #1). Sử dụng **tcpdump** trên máy **Victim** kiểm tra các trường hợp sau:

- Trước khi viết áp dụng rule #1.
- Sau khi áp dụng rule #1.

Kiểm tra alert log của Snort để xem kết quả.

a) Trước khi viết áp dụng rule #1.

- Trên máy Victim thực thi lệnh **tcpdump -i eth0**

```

msfadmin@metasploitable:~$ sudo tcpdump -i eth0
[sudol password for msfadmin:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
  
```


- Sau đó thực hiện ping từ máy kali tới máy victim (10.81.7.100 → 192.168.7.200)
 - Máy kali

```
Debian 11.x (kali linux) - VMware Workstation
File Edit View VM Tabs Help
Router_Ubuntu Server 64-bit Debian 11.x (kali linux) Snort_Ubuntu Server 64-bit
bun@kali: ~/Desktop
File Actions Edit View Help
└─$ ping 192.168.95.200
PING 192.168.95.200 (192.168.95.200) 56(84) bytes of data.
64 bytes from 192.168.95.200: icmp_seq=1 ttl=63 time=48.1 ms
64 bytes from 192.168.95.200: icmp_seq=2 ttl=63 time=0.985 ms
64 bytes from 192.168.95.200: icmp_seq=3 ttl=63 time=1.09 ms
64 bytes from 192.168.95.200: icmp_seq=4 ttl=63 time=1.25 ms
^C
  ─ 192.168.95.200 ping statistics ─
  4 packets transmitted, 4 received, 0% packet loss, time 3016ms
 rtt min/avg/max/mdev = 0.985/12.846/48.067/20.334 ms

(bun@kali)-[~/Desktop]
└─$ ping 192.168.95.200
PING 192.168.95.200 (192.168.95.200) 56(84) bytes of data.
64 bytes from 192.168.95.200: icmp_seq=1 ttl=63 time=1.03 ms
64 bytes from 192.168.95.200: icmp_seq=2 ttl=63 time=1.08 ms
64 bytes from 192.168.95.200: icmp_seq=3 ttl=63 time=1.02 ms
64 bytes from 192.168.95.200: icmp_seq=4 ttl=63 time=0.889 ms
^C
  ─ 192.168.95.200 ping statistics ─
  4 packets transmitted, 4 received, 0% packet loss, time 3025ms
 rtt min/avg/max/mdev = 0.889/1.003/1.082/0.070 ms

(bun@kali)-[~/Desktop]
└─$

(bun@kali)-[~/Desktop]
└─$

(bun@kali)-[~/Desktop]
└─$

(bun@kali)-[~/Desktop]
└─$

(bun@kali)-[~/Desktop]
└─$
```

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

- Máy victim

```

23:33:01.999832 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request
from 00:0c:29:96:b2:1b (oui Unknown), length 291
23:33:04.295077 IP 10.81.95.100 > 192.168.95.200: ICMP echo request, id 20437, s
eq 1, length 64
23:33:04.295104 IP 192.168.95.200 > 10.81.95.100: ICMP echo reply, id 20437, seq
1, length 64
23:33:04.295206 IP 192.168.95.200.56857 > dns.google.domain: 6119+ PTR? 100.95.8
1.10.in-addr.arpa. (43)
23:33:04.342081 IP dns.google.domain > 192.168.95.200.56857: 6119 NXDomain 0/0/0
(43)
23:33:05.304791 IP 10.81.95.100 > 192.168.95.200: ICMP echo request, id 20437, s
eq 2, length 64
23:33:05.304804 IP 192.168.95.200 > 10.81.95.100: ICMP echo reply, id 20437, seq
2, length 64
23:33:06.307121 IP 10.81.95.100 > 192.168.95.200: ICMP echo request, id 20437, s
eq 3, length 64
23:33:06.307137 IP 192.168.95.200 > 10.81.95.100: ICMP echo reply, id 20437, seq
3, length 64
23:33:07.321576 IP 10.81.95.100 > 192.168.95.200: ICMP echo request, id 20437, s
eq 4, length 64
23:33:07.321588 IP 192.168.95.200 > 10.81.95.100: ICMP echo reply, id 20437, seq
4, length 64
23:33:10.612720 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request
from 00:0c:29:96:b2:25 (oui Unknown), length 291

```

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

Ngừng sử dụng tcpdump

```

23:33:58.145002 arp who-has 192.168.95.1 tell 192.168.95.200
23:33:58.145677 arp reply 192.168.95.1 is-at 00:0c:29:23:a0:23 (oui Unknown)
23:33:58.399966 arp who-has 192.168.95.200 tell 192.168.95.1
23:33:58.399994 arp reply 192.168.95.200 is-at 00:0c:29:83:94:9e (oui Unknown)
23:34:05.912263 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request
from 00:0c:29:96:b2:1b (oui Unknown), length 291
23:34:15.246491 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request
from 00:0c:29:96:b2:25 (oui Unknown), length 291
23:34:29.013213 IP 169.254.22.23.netbios-ns > 169.254.255.255.netbios-ns: NBT UD
P PACKET(137): QUERY: REQUEST: BROADCAST
23:34:29.770710 IP 169.254.22.23.netbios-ns > 169.254.255.255.netbios-ns: NBT UD
P PACKET(137): QUERY: REQUEST: BROADCAST
23:34:30.528357 IP 169.254.22.23.netbios-ns > 169.254.255.255.netbios-ns: NBT UD
P PACKET(137): QUERY: REQUEST: BROADCAST
23:34:31.285161 IP 169.254.255.97.netbios-ns > 169.254.255.255.netbios-ns: NBT U
DP PACKET(137): QUERY: REQUEST: BROADCAST
23:34:32.043580 IP 169.254.255.97.netbios-ns > 169.254.255.255.netbios-ns: NBT U
DP PACKET(137): QUERY: REQUEST: BROADCAST
23:34:32.798675 IP 169.254.255.97.netbios-ns > 169.254.255.255.netbios-ns: NBT U
DP PACKET(137): QUERY: REQUEST: BROADCAST

102 packets captured
102 packets received by filter
0 packets dropped by kernel
msfadmin@metasploitable:~$ _

```

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

- Máy snort

```

GNU nano 6.2 /var/log/snort/alert

[**] [1:10000001:1] ICMP test detected [**]
[Priority: 0]
03/22-05:43:46.026940 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:52288 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:20437 Seq:1 ECHO

[**] [1:10000001:1] ICMP test detected [**]
[Priority: 0]
03/22-05:43:47.036087 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:52328 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:20437 Seq:2 ECHO

[**] [1:10000001:1] ICMP test detected [**]
[Priority: 0]
03/22-05:43:48.037801 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:52504 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:20437 Seq:3 ECHO

[**] [1:10000001:1] ICMP test detected [**]
[Priority: 0]
03/22-05:43:49.051600 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:52719 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:20437 Seq:4 ECHO

[ Read 25 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
To direct input to this VM, click inside or press Ctrl+G.
  
```

b) Sau khi áp dụng rule

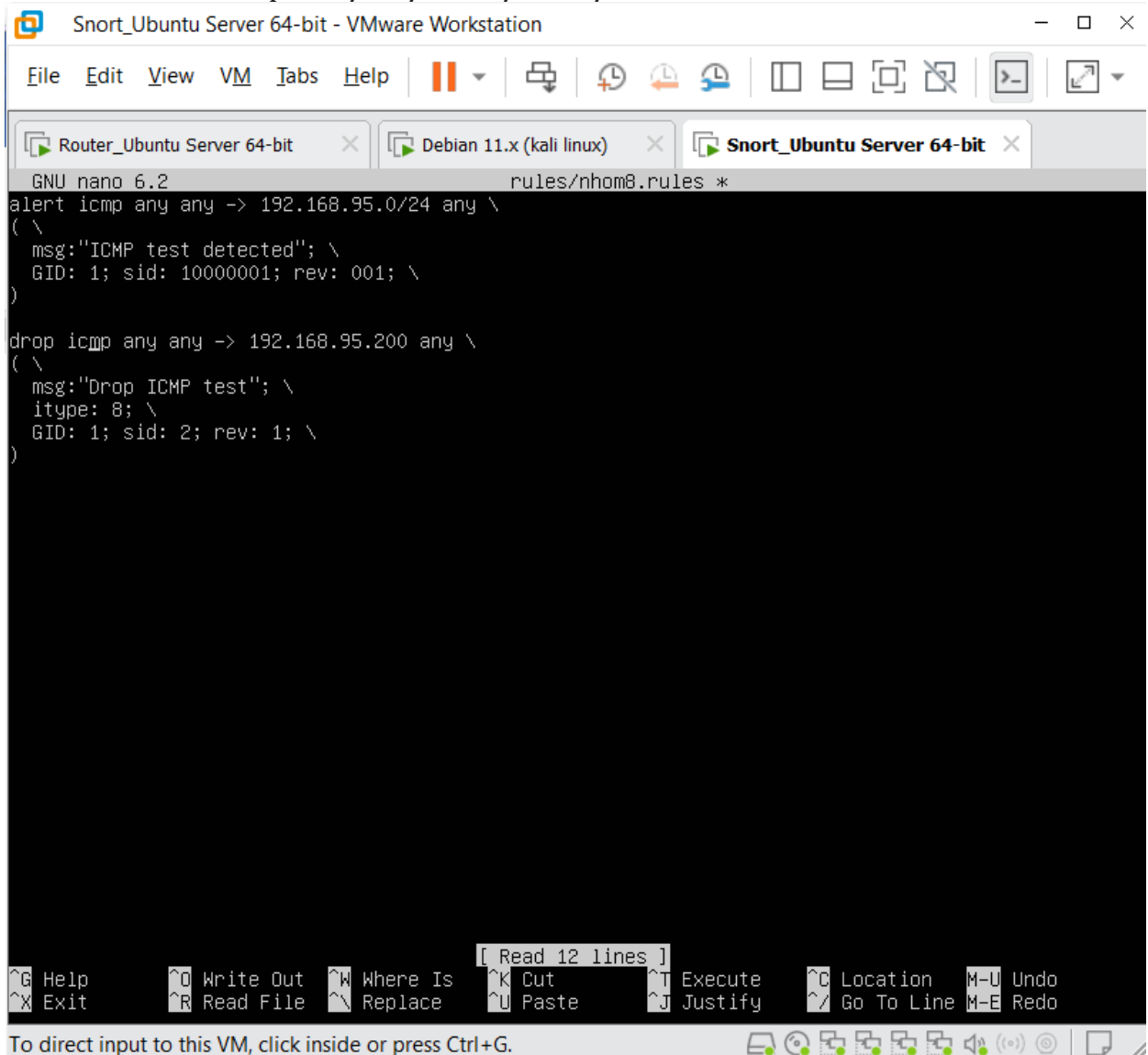
- Viết rule để chặn các gói tin ICMP (cụ thể là gói echo request) do attacker sẽ thực hiện ping tới victim

drop ICMP any any -> 192.168.95.200 any (msg: "Drop ICMP"; GID: 1; sid: 2; rev: 1;)

Ý nghĩa :

- drop ICMP any any : chặn tất cả gói tin ICMP đến từ bất cứ port nào của bất kỳ địa chỉ nào
- -> 192.168.95.200 any : gói tin có địa chỉ đích là 192.168.95.200 và port đích là bất kỳ
- msg: "Drop ICMP" : Hiển thị thông báo khi drop gói tin thỏa điều kiện của rule
- itype: 8 : dùng để chỉ định loại gói tin cần drop là Echo request
- sid : định danh cho rule
- rev : phiên bản của 1 rule
- GID: là generator ID của rule thông báo

- Thêm rule drop vào **/etc/snort/rules/nhomX.rules** như sau:



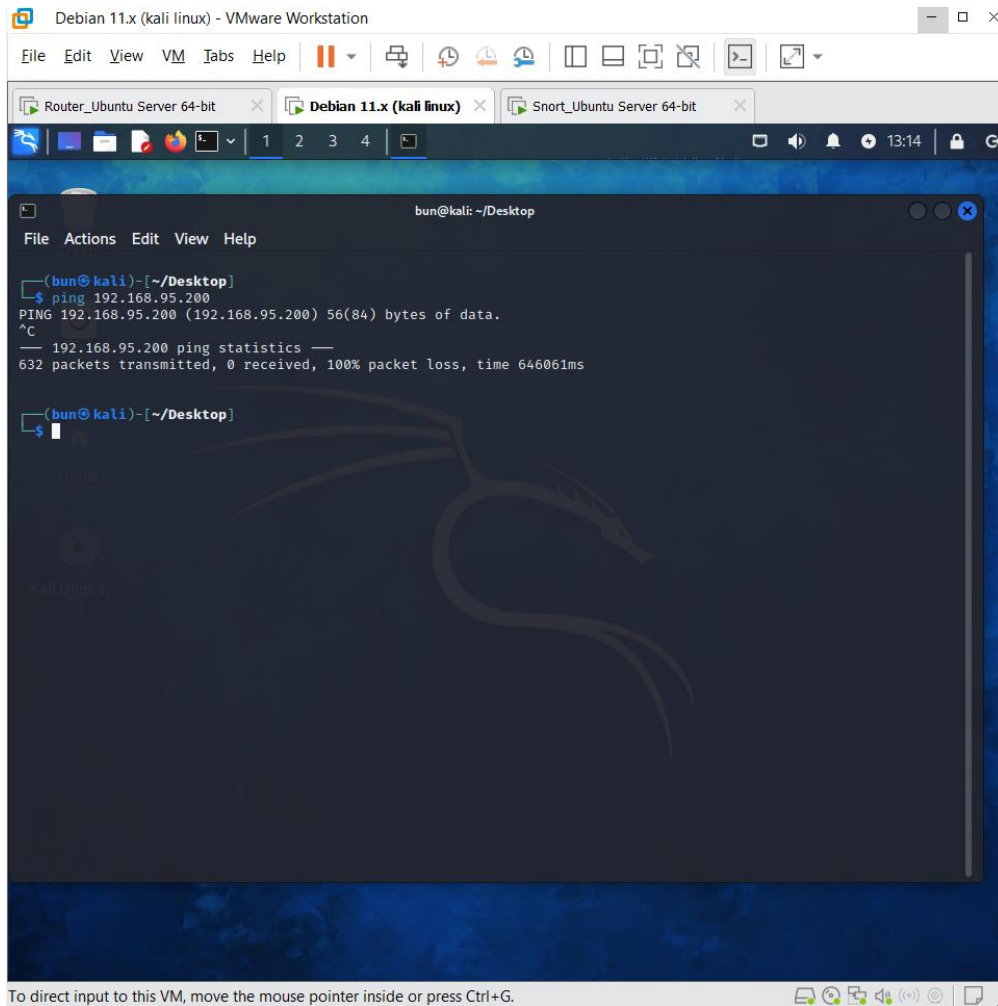
```
GNU nano 6.2 rules/nhom8.rules *
alert icmp any any -> 192.168.95.0/24 any \
( \
  msg:"ICMP test detected"; \
  GID: 1; sid: 10000001; rev: 001; \
)

drop icmp any any -> 192.168.95.200 any \
( \
  msg:"Drop ICMP test"; \
  itype: 8; \
  GID: 1; sid: 2; rev: 1; \
)

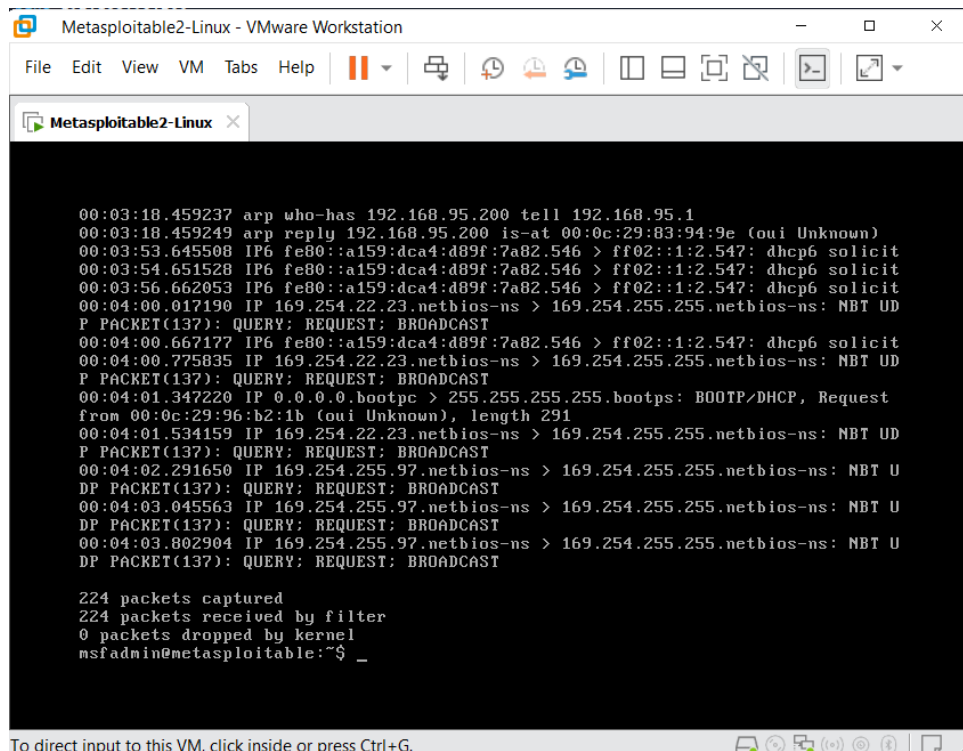
[ Read 12 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^N Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

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

- Máy Kali

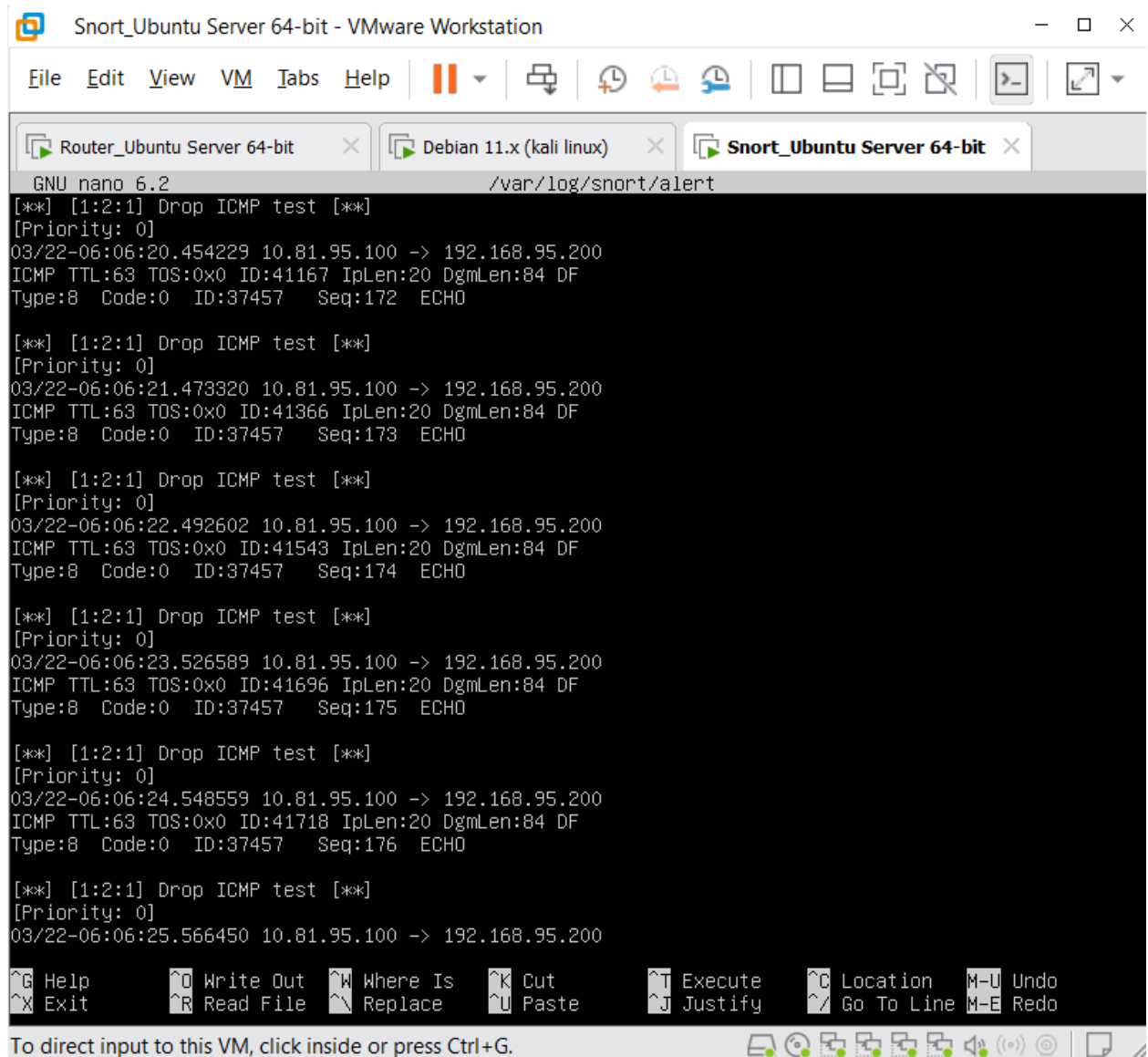


- Máy victim



- Máy Snort

Kiểm tra alert log của Snort để xem kết quả.



```
GNU nano 6.2 /var/log/snort/alert
[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:20.454229 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:41167 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:37457 Seq:172 ECHO

[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:21.473320 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:41366 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:37457 Seq:173 ECHO

[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:22.492602 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:41543 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:37457 Seq:174 ECHO

[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:23.526589 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:41696 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:37457 Seq:175 ECHO

[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:24.548559 10.81.95.100 -> 192.168.95.200
ICMP TTL:63 TOS:0x0 ID:41718 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:37457 Seq:176 ECHO

[**] [1:2:1] Drop ICMP test [**]
[Priority: 0]
03/22-06:06:25.566450 10.81.95.100 -> 192.168.95.200

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^G Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line   M-E Redo

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