

Task 1

57118209 余悦

1.1A

Ifconfig 查看广播地址之后完成 sniffer.py 代码：

```
~/Desktop/Labs_20.04/Network Security/Packet Sniffing and Spoofing Lab/L
1 from scapy.all import *
2
3 def print_pkt(pkt):
4     pkt.show()
5
6 pkt = sniff(iface='br-0ec83bcbe5b9', filter='icmp', prn=print_pkt)
```

Root 权限运行如下：

```
[07/04/21]seed@VM:~/.../volumes$ gedit sniffer.py
[07/04/21]seed@VM:~/.../volumes$ sudo python3 sniffer.py
###[ Ethernet ]###
  dst      = 02:42:0a:09:00:05
  src      = 02:42:1b:37:76:44
  type     = IPv4
###[ IP ]###
  version  = 4
  ihl      = 5
  tos      = 0x0
  len      = 84
  id       = 29652
  flags    = DF
  frag     = 0
  ttl      = 64
  proto    = icmp
  chksum   = 0xb2bd
  src      = 10.9.0.1
  dst      = 10.9.0.5
  \options \
###[ ICMP ]###
  type     = echo-request
```

普通权限则报错：因为普通用户没有权限创建 socket：

```

[21] stopped sudo python3 sniffer.py
[07/04/21]seed@VM:~/../volumes$ python3 sniffer.py
Traceback (most recent call last):
  File "sniffer.py", line 6, in <module>
    pkt = sniff(iface='br-0ec83bcbe5b9',filter='icmp',prn=print_pkt)
  File "/usr/local/lib/python3.8/dist-packages/scapy/sendrecv.py", line 1036, in sniff
    sniffer._run(*args, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/scapy/sendrecv.py", line 906, in _run
    sniff_sockets[L2socket(type=ETH_P_ALL, iface=iface,
  File "/usr/local/lib/python3.8/dist-packages/scapy/arch/linux.py", line 398, in __init__
    self.ins = socket.socket(socket.AF_PACKET, socket.SOCK_RAW, socket.htons(type)) # noqa: E501
  File "/usr/lib/python3.8/socket.py", line 231, in __init__
    _socket.socket.__init__(self, family, type, proto, fileno)
PermissionError: [Errno 1] Operation not permitted
[07/04/21]seed@VM:~/../volumes$

```

1.1B1

同上

1.1B2

Tcp_sniffer.py 代码:

```

1 from scapy.all import *
2
3 def print_pkt(pkt):
4     pkt.show()
5
6 pkt = sniff(filter='tcp and src host 10.9.0.5 and dst port 23',prn=print_pkt)

```

Send 代码:

```
1 from scapy.all import *
2
3 ip=IP()
4 ip.src='10.9.0.5'
5 ip.dst='10.9.0.2'
6 tcp=TCP()
7 tcp.dport=23
8 send(ip/tcp)
9
```

运行后结果：捕获成功

```
root@VM:/volumes# python3 sniffer.py
###[ Ethernet ]###
  dst      = ff:ff:ff:ff:ff:ff
  src      = 02:42:1b:37:76:44
  type     = IPv4
###[ IP ]###
  version  = 4
  ihl      = 5
  tos      = 0x0
  len      = 40
  id       = 1
  flags    =
  frag     = 0
  ttl      = 64
  proto    = tcp
  chksum   = 0x66b7
  src      = 10.9.0.5
  dst      = 10.9.0.2
  \options \
###[ TCP ]###
  sport    = ftp_data
  dport    = telnet
  seq      = 0
  ack      = 0
  dataofs  = 5
  reserved = 0
  flags    = S
  window   = 8192
```

```

        chksum      = 0x66b7
        src          = 10.9.0.5
        dst          = 10.9.0.2
        \options     \
###[ TCP ]###
        sport        = ftp_data
        dport        = telnet
        seq          = 0
        ack          = 0
        dataofs      = 5
        reserved     = 0
        flags        = S
        window       = 8192
        chksum       = 0x7b9f
        urgptr       = 0
        options      = []

```

1.1B3

Send 代码:

```

1 from scapy.all import *
2
3 ip=IP()
4 ip.src='10.9.0.5'
5 ip.dst='128.230.164.1'
6 send(ip)

```

Sniffer 代码:

```

1 from scapy.all import *
2
3 def print_pkt(pkt):
4     pkt.show()
5
6 pkt = sniff(filter='ip and dst host 128.230.164.1', prn=print_pkt)

```

运行之后:

```

root@VM:/volumes# python3 subnet_sniffer.py
###[ Ethernet ]###
  dst      = 00:50:56:e4:7f:b5
  src      = 00:0c:29:2c:3e:1a
  type     = IPv4
###[ IP ]###
  version  = 4
  ihl      = 5
  tos      = 0x0
  len      = 20
  id       = 1
  flags    =
  frag     = 0
  ttl      = 64
  proto    = hopopt
  chksum   = 0x4bf4
  src      = 10.9.0.5
  dst      = 128.230.164.1
  \options \

^Z
[1]+  Stopped                  python3 subnet_sniffer.py
root@VM:/volumes#

```

1.2

Spoofing 代码:

```

from scapy.all import *

a = IP()
b = ICMP()
a.dst = '10.9.0.5'
p = a/b
send(p)

```

运行:

```

[07/06/21] seed@VM:~/.../volumes$ sudo python3 icmp_spoofing.py
.
Sent 1 packets.

```

Wireshark 查看:

No.	Time	Source	Destination	Protocol	Length	Info
1	2021-07-06 08:5...	02:42:9b:a0:a2:a7	Broadcast	ARP	42	Who has 10.9.0.5? Tell 10.9.0.1
2	2021-07-06 08:5...	02:42:0a:09:00:05	02:42:9b:a0:a2:a7	ARP	42	10.9.0.5 is at 02:42:0a:09:00:05
3	2021-07-06 08:5...	10.9.0.1	10.9.0.5	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (reply in 4)
4	2021-07-06 08:5...	10.9.0.5	10.9.0.1	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=64 (request in 3)

伪装 spoofing 代码:

```
#!/usr/bin/python3

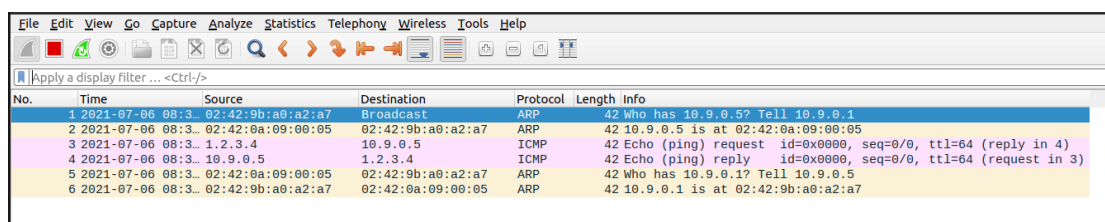
from scapy.all import*

ip= IP()
ip.src = '1.2.3.4'
ip.dst = '10.9.0.5'
b = ICMP()
p = ip/b
send(p)
```

运行:

```
[07/06/21] seed@VM:~/.../volumes$ sudo python3 icmp_spoofing.py
Sent 1 packets.
```

Wireshark 查看: 伪装成功



No.	Time	Source	Destination	Protocol	Length	Info
1	2021-07-06 08:3...	02:42:9b:a0:a2:a7	Broadcast	ARP	42	Who has 10.9.0.5? Tell 10.9.0.1
2	2021-07-06 08:3...	02:42:0a:09:00:05	02:42:9b:a0:a2:a7	ARP	42	10.9.0.5 is at 02:42:0a:09:00:05
3	2021-07-06 08:3...	1.2.3.4	10.9.0.5	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (reply in 4)
4	2021-07-06 08:3...	10.9.0.5	1.2.3.4	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=64 (request in 3)
5	2021-07-06 08:3...	02:42:0a:09:00:05	02:42:9b:a0:a2:a7	ARP	42	Who has 10.9.0.1? Tell 10.9.0.5
6	2021-07-06 08:3...	02:42:9b:a0:a2:a7	02:42:0a:09:00:05	ARP	42	10.9.0.1 is at 02:42:9b:a0:a2:a7

1.3

代码如下: 无限循环 ttl

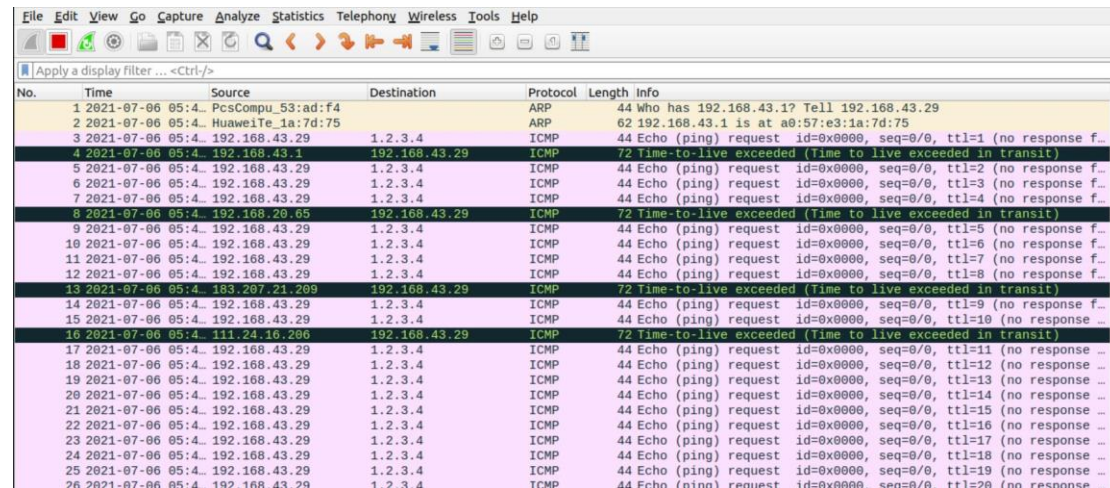


```
Open [icon] ~/Desktop/Labs_2
*sniffer.c

1 from scapy.all import *
2
3 ttl=1
4 while True:
5     a = IP()
6     a.dst = '1.2.3.4'
7     a.ttl = ttl
8     b = ICMP()
9     send(a/b)
10    ttl += 1
```

运行之后观察 wireshark: 途径 ip 地址有

192.168.43.1, 192.168.20.65, 183.207.21.20, 111.24.16.206, 到达
1.2.3.4



No.	Time	Source	Destination	Protocol	Length	Info
1	2021-07-06 05:4...	PcsCompu_53:ad:f4		ARP	44	Who has 192.168.43.1? Tell 192.168.43.29
2	2021-07-06 05:4...	HuaweiTe_1a:7d:75		ARP	62	192.168.43.1 is at a0:57:e3:1a:7d:75
3	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=1 (no response f...
4	2021-07-06 05:4...	192.168.43.1	192.168.43.29	ICMP	72	Time-to-live exceeded (Time to live exceeded in transit)
5	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=2 (no response f...
6	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=3 (no response f...
7	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=4 (no response f...
8	2021-07-06 05:4...	192.168.20.65	192.168.43.29	ICMP	72	Time-to-live exceeded (Time to live exceeded in transit)
9	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=5 (no response f...
10	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=6 (no response f...
11	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=7 (no response f...
12	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=8 (no response f...
13	2021-07-06 05:4...	183.207.21.209	192.168.43.29	ICMP	72	Time-to-live exceeded (Time to live exceeded in transit)
14	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=9 (no response f...
15	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=10 (no response f...
16	2021-07-06 05:4...	111.24.16.206	192.168.43.29	ICMP	72	Time-to-live exceeded (Time to live exceeded in transit)
17	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=11 (no response f...
18	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=12 (no response f...
19	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=13 (no response f...
20	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=14 (no response f...
21	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=15 (no response f...
22	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=16 (no response f...
23	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=17 (no response f...
24	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=18 (no response f...
25	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=19 (no response f...
26	2021-07-06 05:4...	192.168.43.29	1.2.3.4	ICMP	44	Echo (ping) request id=0x0000, seq=0/0, ttl=20 (no response f...

1.4

1.2.3.4

在宿主主机 ping 不通, 因为网络地址不存在:

```
[1]: Stopped ping 1.2.3.4
root@VM:/# ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
```

代码:

```

1|from scapy.all import*
2
3|def spoof_pkt(pkt):
4|    if ICMP in pkt and pkt[ICMP].type==8:
5|        ip=IP(src=pkt[IP].dst,dst=pkt[IP].src,ihl=pkt[IP].ihl)
6|        icmp=ICMP(type=0,id=pkt[ICMP].id,seq=pkt[ICMP].seq)
7|        data= pkt[Raw].load
8|        newpkt = ip/icmp/data
9|        send(newpkt)
10
11|pkt=sniff(filter='icmp',prn=spoof_pkt)
12

```

捕获 ICMP 报文，宿源地址对调，设置 ICMP 为 reply 类型，发出即可伪造。

运行代码之后，则可以 ping 通：伪造成功

```

Sent 1 packets.
[07/06/21]seed@VM:~/.../volumes$ sudo python3 sniff_spoof.py
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
root@VM:/# ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
64 bytes from 1.2.3.4: icmp_seq=68 ttl=64 time=33.3 ms
64 bytes from 1.2.3.4: icmp_seq=69 ttl=64 time=33.4 ms
64 bytes from 1.2.3.4: icmp_seq=70 ttl=64 time=24.4 ms
64 bytes from 1.2.3.4: icmp_seq=71 ttl=64 time=26.6 ms
64 bytes from 1.2.3.4: icmp_seq=72 ttl=64 time=30.7 ms
64 bytes from 1.2.3.4: icmp_seq=73 ttl=64 time=25.5 ms
64 bytes from 1.2.3.4: icmp_seq=74 ttl=64 time=31.7 ms
64 bytes from 1.2.3.4: icmp_seq=75 ttl=64 time=35.1 ms
64 bytes from 1.2.3.4: icmp_seq=76 ttl=64 time=36.6 ms
64 bytes from 1.2.3.4: icmp_seq=77 ttl=64 time=33.3 ms
64 bytes from 1.2.3.4: icmp_seq=78 ttl=64 time=29.2 ms
64 bytes from 1.2.3.4: icmp_seq=79 ttl=64 time=33.1 ms
64 bytes from 1.2.3.4: icmp_seq=80 ttl=64 time=23.7 ms
64 bytes from 1.2.3.4: icmp_seq=81 ttl=64 time=35.4 ms
64 bytes from 1.2.3.4: icmp_seq=82 ttl=64 time=21.8 ms

```


10.9.0.99

运行代码前后均不可 ping 通，因为此为不存在的本机地址，不经过路由器：

```
[07/06/21]seed@VM:~$ ping 10.9.0.99
PING 10.9.0.99 (10.9.0.99) 56(84) bytes of data.
From 10.9.0.1 icmp_seq=1 Destination Host Unreachable
From 10.9.0.1 icmp_seq=2 Destination Host Unreachable
From 10.9.0.1 icmp_seq=3 Destination Host Unreachable
From 10.9.0.1 icmp_seq=4 Destination Host Unreachable
From 10.9.0.1 icmp_seq=5 Destination Host Unreachable
From 10.9.0.1 icmp_seq=6 Destination Host Unreachable
^Z
[8]+  Stopped                  ping 10.9.0.99
[07/06/21]seed@VM:~$
```

8.8.8.8

运行代码前后均可以 ping 通，因为主机存在：

```
root@VM:/# 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=128 time=31.0 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=31.0 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=29.7 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=30.1 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=128 time=33.9 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=128 time=31.9 ms
^Z
root@VM:/# 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=64 time=20.1 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=31.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=64 time=20.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=30.9 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=64 time=19.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=32.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=64 time=25.5 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=33.0 ms
^Z
[1]+  Stopped                  ping 8.8.8.8
```