

Nguyen Viet Hoang Nam

ITITI19162

# System and Network Security

## Lab 2

```
[10/26/23]seed@VM:~/.../Labsetup$ dockkps
1ceb2fdb91f attacker
d14314f6ad7e hostA
6d756689287a hostB
[10/26/23]seed@VM:~/.../Labsetup$ docksh attacker
root@seed-attacker:/# ip -br addr
lo UNKNOWN 127.0.0.1/8 ::1/128
enp0s3 UP 10.0.2.15/24 fe80::48d5:f93f:c20c:f35e/64
docker0 DOWN 172.17.0.1/16
br-7a05fc595219 UP 10.9.0.1/24 fe80::42:68ff:fe90:2b98/64
veth3f2ea76@if5 UP fe80::e8ce:baff:fe20:92fc/64
veth5ba7b2b@if7 UP fe80::2435:1bff:feda:de68/64
root@seed-attacker:/#
```

My interface name of 10.9.0.1 is br-7a05fc595219

### Task 1.1:

a) With the root privilege - sudo

```
root@seed-attacker:/# sudo python3 sniffer.py
bash: python3: command not found
root@seed-attacker:/# exit
exit
[10/26/23]seed@VM:~/.../Labsetup$ docksh attacker
root@seed-attacker:/# ifconfig
br-7a05fc595219: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
    inet6 fe80::42:68ff:fe90:2b98 prefixlen 64 scopeid 0x20<link>
    ether 02:42:68:fe90:2b98 txqueuelen 0 (Ethernet)
    RX packets 27 bytes 1753 (1.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 78 bytes 8453 (8.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:54:00:00:02 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::48d5:f93f:c20c:f35e prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:fe:59:63 txqueuelen 1000 (Ethernet)
    RX packets 5674 bytes 536797 (536.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3573 bytes 926673 (926.6 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 185 bytes 17663 (17.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 185 bytes 17663 (17.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
veth3f2ea76: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::e8ce:baff:fe20:92fc prefixlen 64 scopeid 0x20<link>
    ether 08:00:20:92:fc:2e txqueuelen 0 (Ethernet)
    RX packets 27 bytes 2131 (2.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 107 bytes 11640 (11.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@seed-attacker:/# exit
exit
[10/26/23]seed@VM:~/.../Labsetup$ docksh attacker
root@seed-attacker:/# ifconfig
br-7a05fc595219: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
    inet6 fe80::42:68ff:fe90:2b98 prefixlen 64 scopeid 0x20<link>
    ether 02:42:68:fe90:2b98 txqueuelen 0 (Ethernet)
    RX packets 27 bytes 1753 (1.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 78 bytes 8453 (8.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:54:00:00:02 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::48d5:f93f:c20c:f35e prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:fe:59:63 txqueuelen 1000 (Ethernet)
    RX packets 5674 bytes 536797 (536.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3573 bytes 926673 (926.6 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 185 bytes 17663 (17.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 185 bytes 17663 (17.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
veth3f2ea76: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::e8ce:baff:fe20:92fc prefixlen 64 scopeid 0x20<link>
    ether 08:00:20:92:fc:2e txqueuelen 0 (Ethernet)
    RX packets 27 bytes 2131 (2.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 107 bytes 11640 (11.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@seed-attacker:/# exit
exit
[10/26/23]seed@VM:~/.../Labsetup$ docksh attacker
root@seed-attacker:/# ifconfig
br-7a05fc595219: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
    inet6 fe80::42:68ff:fe90:2b98 prefixlen 64 scopeid 0x20<link>
    ether 02:42:68:fe90:2b98 txqueuelen 0 (Ethernet)
    RX packets 27 bytes 1753 (1.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 78 bytes 8453 (8.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:54:00:00:02 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::48d5:f93f:c20c:f35e prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:fe:59:63 txqueuelen 1000 (Ethernet)
    RX packets 5674 bytes 536797 (536.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3573 bytes 926673 (926.6 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 185 bytes 17663 (17.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 185 bytes 17663 (17.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
veth3f2ea76: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::e8ce:baff:fe20:92fc prefixlen 64 scopeid 0x20<link>
    ether 08:00:20:92:fc:2e txqueuelen 0 (Ethernet)
    RX packets 27 bytes 2131 (2.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 107 bytes 11640 (11.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

The sudo command not only gave the icmp request – reply but also show the source MAC address (mine was 02:42:68:90:2b:98) and the destination MAC address.

Without the root privilege:

```
[10/26/23]seed@VM:~/../volumes$ python3 sniffer.py
Traceback (most recent call last):
  File "sniffer.py", line 6, in <module>
    pkt = sniff(iface="br-7a05fc595219", filter="icmp", prn=print_pkt, count=5)
  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_socket(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: E
501
  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
```

It seem that the file could not run without the root permission – sudo.

b) Only ICMP packets (ping 10.9.0.5 - hostA): with the source IP address is 10.9.0.1 – attacker to destination IP address 10.9.0.5.

<pre>root@seed-attacker:~# ping 10.9.0.5 PING 10.9.0.5 (10.9.0.5) 56(84) bytes of data: 64 bytes from 10.9.0.5: icmp_seq=1 ttl=64 time=0.071 ms 64 bytes from 10.9.0.5: icmp_seq=2 ttl=64 time=0.188 ms ^C --- 10.9.0.5 ping statistics --- 2 packets transmitted, 2 received, 0% packet loss, time 1012ms rtt min/avg/max/mdev = 0.071/0.129/0.188/0.058 ms root@seed-attacker:~#</pre>	<pre>[10/26/23]seed@VM:~/../volumes\$ sudo python3 sniffer.py #### Ethernet #### dst      = 02:42:0a:09:00:05 src      = 02:42:68:90:2b:98 type     = IPv4 #### IP #### version  = 4 ihl      = 5 tos      = 0x0 len      = 84 id       = 52752 flags    = DF frag     = 0 ttl      = 64 proto    = icmp chksum   = 0x5489 src      = 10.9.0.1 dst      = 10.9.0.5 \options \ #### ICMP #### type     = echo-request code     = 0 chksum   = 0x9302 id       = 0x9 seq      = 0x1 #### Raw #### load     = '():e\x00\x00\x00\x00\x00\x06\x91\n\x00\x00\x00\x00\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f !"#%&amp;'()*+,-./01234567' #### Ethernet #### dst      = 02:42:68:90:2b:98 src      = 02:42:0a:09:00:05 type     = IPv4 #### IP #### version  = 4 ihl      = 5 tos      = 0x0 len      = 84 id       = 49493 flags    = frag     = 0 ttl      = 64 proto    = icmp chksum   = 0xa53c src      = 10.9.0.5 dst      = 10.9.0.1 \options \ #### ICMP ####</pre>
--	--

TCP packets:

```
root@seed-attacker:~# telnet 10.9.0.6
Trying 10.9.0.6...
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
host@10.9.0.6 login:
Login timed out after 60 seconds.
Connection closed by foreign host.
root@seed-attacker:~#

[10/26/23]seed@VM:~/../volumes$ sudo python3 sniffer.py
###[ Ethernet ]###
dst      = 02:42:0a:09:00:06
src      = 02:42:68:90:2b:98
type     = IPv4
###[ IP ]###
version  = 4
ihl      = 5
tos      = 0x10
len      = 60
id       = 40095
flags    = DF
frag     = 0
ttl      = 64
proto    = tcp
chksum   = 0x89f4
src      = 10.9.0.1
dst      = 10.9.0.6
Options  \
###[ TCP ]###
sport    = 60060
dport    = telnet
seq      = 3083629346
ack      = 0
dataoffs = 10
reserved = 0
flags    = S
window   = 64240
chksum   = 0x1447
urgptr   = 0
options  = [('MSS', 1460), ('AckOK', b''), ('Timestamp', (1198172519, 0)), ('NOP', N
one), ('WScale', 7)]

###[ Ethernet ]###
dst      = 02:42:0a:09:00:06
src      = 02:42:68:90:2b:98
type     = IPv4
###[ IP ]###
version  = 4
ihl      = 5
tos      = 0x10
len      = 52
id       = 40096
flags    = DF
frag     = 0
ttl      = 64
proto    = tcp
chksum   = 0x89ff
```

A TCP packet showed an AckOK flag. And the sniffer.py was adjusted as follow:

```
# sniffer.py
#!/usr/bin/python3
from scapy.all import *
def print_pkt(pkt):
    pkt.show()
f = "tcp and dst port 23 and src net 10.9.0.1"
pkt = sniff(iface="br-7a05fc595219",filter=f, prn=print_pkt, count=100)
```

Any packets:

```

seed@VM: ~
root@seed-attacker:~# ping 125.212.138.21
PING 125.212.138.21 (125.212.138.21) 56(84) bytes of data.
64 bytes from 125.212.138.21: icmp_seq=1 ttl=50 time=45.9 ms
64 bytes from 125.212.138.21: icmp_seq=2 ttl=50 time=42.9 ms
64 bytes from 125.212.138.21: icmp_seq=3 ttl=50 time=32.3 ms
64 bytes from 125.212.138.21: icmp_seq=4 ttl=50 time=32.3 ms
^C
--- 125.212.138.21 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 32.261/38.339/45.879/6.133 ms
root@seed-attacker:~#

[10/27/23]seed@VM:~/volumes$ sudo python3 sniffer.py
###[ Ethernet ]###
dst      = 52:54:00:12:35:02
src      = 08:00:27:fe:59:63
type     = IPv4
###[ IP ]###
version  = 4
ihl      = 5
tos      = 0x0
len      = 84
id       = 53180
flags    = 0x
frag     = 0
ttl      = 64
proto    = icmp
chksum   = 0x56f4
src      = 10.0.2.15
dst      = 125.212.138.21
\options \
###[ ICMP ]###
type     = echo-request
code     = 0
chksum   = 0x2477
id       = 0x3
seq      = 0x1
###[ Raw ]###
load     = '\x00\x00\x00\x00\xaa\xd1\x06\x00\x00\x00\x00\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f !"#%&'()*+,-./01234567'
###[ Ethernet ]###
dst      = 08:00:27:fe:59:63
src      = 52:54:00:12:35:02
type     = IPv4
###[ IP ]###
version  = 4
ihl      = 5
tos      = 0x0
len      = 84
id       = 1765
flags    =
frag     = 0
ttl      = 50
proto    = icmp
chksum   = 0x6d0c
src      = 125.212.138.21
dst      = 10.0.2.15
\options \
###[ ICMP ]###

```

I have captured ICMP, Ethernet, IP packets.

Here is my sniffer.py:

```

# sniffer.py
#!/usr/bin/python3
from scapy.all import *
def print_pkt(pkt):
    pkt.show()
target_subnet = "125.212.128.0/17"
filter_str = f"net {target_subnet}"
pkt = sniff(filter=filter_str, prn=print_pkt, count=5)

```

## Task 1.2

```

seed@VM: ~
[10/27/23]seed@VM:~/.../volumes$ sudo python3 sniffer.py
[[[ Ethernet ]]]
dst      = 02:42:0a:09:00:05
src      = 02:42:52:62:43:db
type     = IPv4
[[[ IP ]]]
version  = 4
ihl      = 5
tos      = 0x0
len      = 28
id       = 1
flags    =
frag     = 0
ttl      = 64
proto    = icmp
chksum   = 0x66c9
src      = 10.9.0.1
dst      = 10.9.0.5
options  \
[[[ ICMP ]]]
type     = echo-request
code     = 0
chksum   = 0xf7ff
id       = 0x0
seq      = 0x0

[[[ Ethernet ]]]
dst      = 02:42:52:62:43:db
src      = 02:42:0a:09:00:05
type     = IPv4
[[[ IP ]]]
version  = 4
ihl      = 5
tos      = 0x0
len      = 28
id       = 2475
flags    =
frag     = 0
ttl      = 64
proto    = icmp
chksum   = 0x5d1f
src      = 10.9.0.5
dst      = 10.9.0.1
options  \
[[[ ICMP ]]]
type     = echo-reply
code     = 0
chksum   = 0xffff

[10/27/23]seed@VM:~/.../volumes$

```

The echo-request packet was captured which showed the source IP address is 10.9.0.1 and destination IP address is 10.9.0.5.

Now, if I changed the source IP address to 10.9.0.99 (a.dst = “10.9.0.99”) then the echo-request packet would also display the source IP address is it.

```

[10/27/23]seed@VM:~/.../volumes$ sudo python3 sniffer.py
[[[ Ethernet ]]]
dst      = 02:42:0a:09:00:05
src      = 02:42:52:62:43:db
type     = IPv4
[[[ IP ]]]
version  = 4
ihl      = 5
tos      = 0x0
len      = 28
id       = 1
flags    =
frag     = 0
ttl      = 64
proto    = icmp
chksum   = 0x66c7
src      = 10.9.0.99
dst      = 10.9.0.5
options  \
[[[ ICMP ]]]
type     = echo-request
code     = 0
chksum   = 0xf7ff
id       = 0x0
seq      = 0x0

[[[ Ethernet ]]]
dst      = 02:42:52:62:43:db
src      = 02:42:0a:09:00:05
type     = IPv4
[[[ IP ]]]
version  = 4
ihl      = 5
tos      = 0x0
len      = 28
id       = 2475
flags    =
frag     = 0
ttl      = 64
proto    = icmp
chksum   = 0x5d1f
src      = 10.9.0.5
dst      = 10.9.0.1
options  \
[[[ ICMP ]]]
type     = echo-reply
code     = 0
chksum   = 0xffff

[10/27/23]seed@VM:~/.../volumes$

```

## Task 1.3

When TTL = 1, the router IP address is 10.0.2.2



When TTL = 10, the router IP address is 108.170.241.33

[illegible]

When TTL = 15, the router IP address is 209.85.252.101





So after about 25 hops, the ICMP packet made it way to the destination hop - 74.125.24.105.

## Task 1.4

```
root@hostA-10_9_0_5:/# ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
^C
--- 1.2.3.4 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9231ms

root@hostA-10_9_0_5:/# ping 10.9.0.99
PING 10.9.0.99 (10.9.0.99) 56(84) bytes of data.
From 10.9.0.5 icmp_seq=1 Destination Host Unreachable
From 10.9.0.5 icmp_seq=2 Destination Host Unreachable
From 10.9.0.5 icmp_seq=3 Destination Host Unreachable

--- 10.9.0.99 ping statistics ---
5 packets transmitted, 0 received, +3 errors, 100% packet loss, time 4080ms
pipe 4
root@hostA-10_9_0_5:/# 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=113 time=42.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=50.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=42.8 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=113 time=43.3 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3032ms
rtt min/avg/max/mdev = 42.753/44.845/50.530/3.288 ms
```

Before turning on the spoof.py, host A can not ping to the 1.2.3.4 and 10.9.0.99 because the IP address did not exist in this LAN or on the Internet.

Now, the spoof.py would be turned on to capture those ping packets from host A and send them a reply packet from the attacker but with the destination IP address.

<pre> root@hostA-10_9_0_5:/# [5/129] root@hostA-10_9_0_5:/# 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=1 ttl=99 time=55.5 ms ^C --- 1.2.3.4 ping statistics --- 1 packets transmitted, 1 received, 0% packet loss, time 0ms rtt min/avg/max/mdev = 55.514/55.514/55.514/0.000 ms root@hostA-10_9_0_5:/# </pre>	<pre> python3 spoof.py Original Packet... Source IP: 10.9.0.5 Destination IP: 1.2.3.4 Spoofed Packet... Source IP: 1.2.3.4 Destination IP: 10.9.0.5 </pre>
---	--

Host A now would receive the request back from the attacker pretending to be the destination IP address. And so as for the 8.8.8.8

<pre> 3 time=42.8 ms --- 8.8.8.8 ping statistics --- 1 packets transmitted, 1 received, 0% packet loss, time 0ms rtt min/avg/max/mdev = 42.755/42.755/42.755/0.000 ms root@hostA-10_9_0_5:/# </pre>	<pre> Original Packet... Source IP: 10.9.0.5 Destination IP: 8.8.8.8 Spoofed Packet... Source IP: 8.8.8.8 Destination IP: 10.9.0.5 </pre>
---	---

However, the 10.9.0.99 can not be spoofed it was because the packet did not go to the Internet which the attacker can sniff it to spoof.