**Assignment 04**

**Port Discovery Techniques**

**CS4061**

**Ethical Hacking Concepts and Practices**

**Submitted by:** Momenah Saif

**Roll number:** 21i-1909

**Date:** 21-03-2024

**Table of Contents**

[ **Introduction** 3](#_Toc161944395)

[ **Code Explanation** 3](#_Toc161944396)

[ **Outputs** 4](#_Toc161944397)

[ **Summary** 8](#_Toc161944398)

[ **References** 8](#_Toc161944399)

# **Introduction**

In this assignment we were given to write a python code that shows options to run number of scans on a target machine. My code runs the following scans.   
1. ICMP Ping Scan

2. UDP Ping Scan

3. TCP Scan

o SYN Scan (Full Open Scan)

o Stealth Scan (Half Open Scan)

o Inverse TCP Flag Scan

* FIN Scan
* Null Scan
* XMAS Scan
* Maimon Scan

o ACK Flag Scan

* TTL Based Scan
* Window Scan

I mostly used Scapy for scan but in some areas scapy wasn’t working write so I used other methods like nmap

# **Code Explanation**

**Imports:**

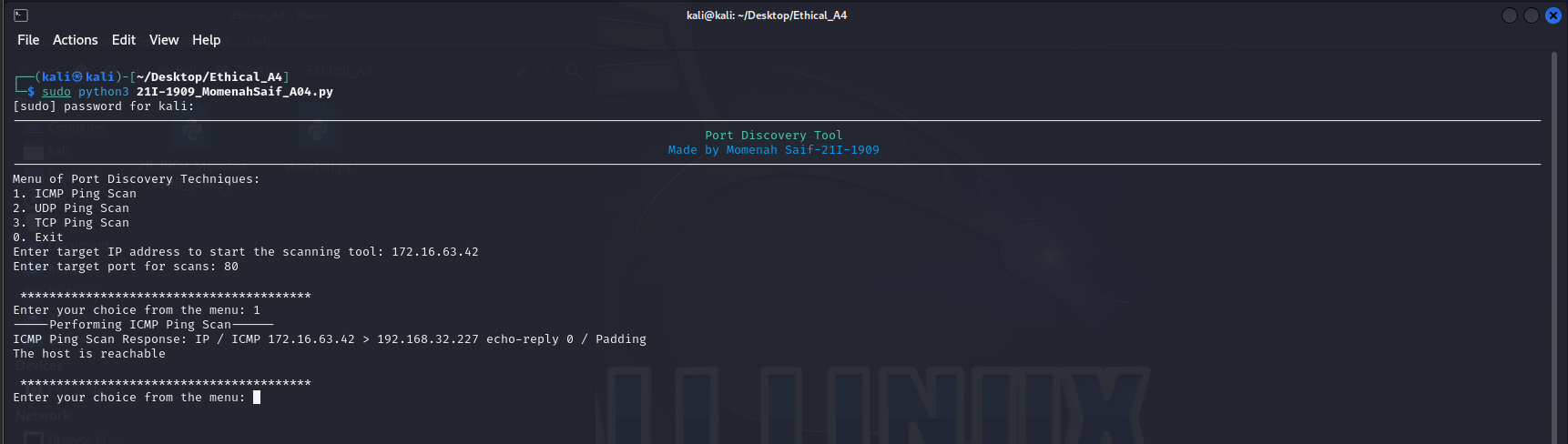
* logging: It is used to control informational messages from external libraries.
* scapy: A packet manipulation library for crafting and sending network packets.
* nmap: It is a library for using the Nmap port scanner within the script to check for OS.
* shutil: It provides utility functions for interacting with the file system. It was used in code to create a better output

**Functions:**

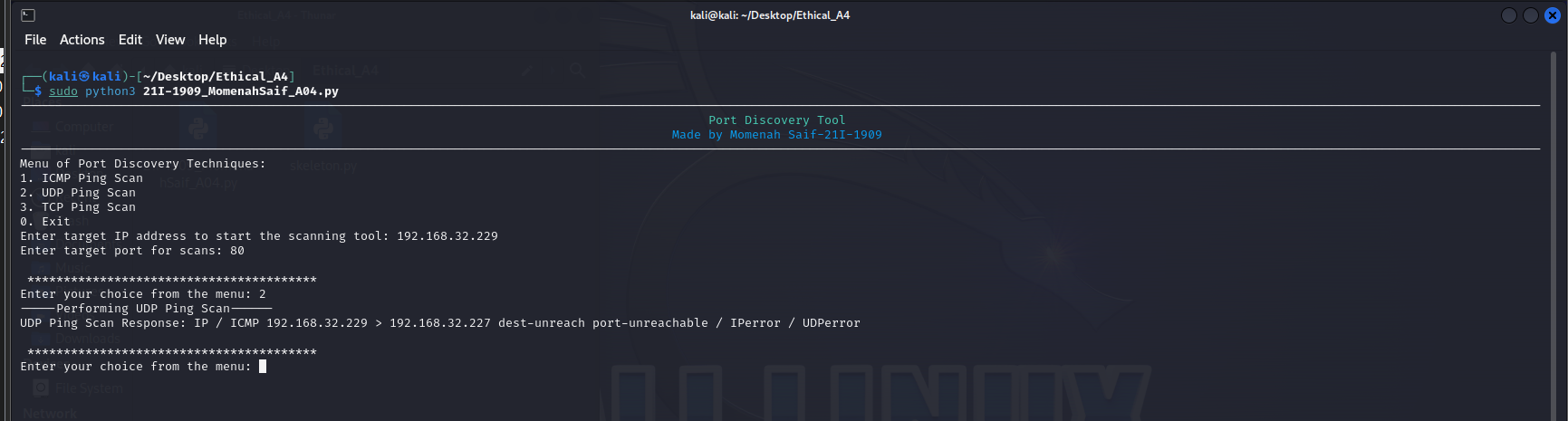
* icmp\_ping\_scan(target): It sends ICMP echo requests to check host reachability.
* udp\_ping\_scan(target): It sends a UDP packet with a random destination port to see if there's a response. Useful for basic reachability checks.
* tcp\_syn\_scan(target, port): It sends a TCP SYN packet to a specific port to identify open or closed ports ( TCP 3-way handshake).
* tcp\_stealth\_scan(target, port): It is similar to SYN scan but sends a packet with an invalid source port, making it harder to detect by firewalls.
* inv\_tcp\_scan(target, port, flag):It sends a TCP packet with various flags (FIN, NULL, XMAS) to probe a port and gather its state based on the response.
* ack\_ttl\_scan(target, port): It sends a TCP packet with the ACK flag set to determine if the port is filtered or unfiltered based on the response flags.
* ack\_windows\_scan(target, port): It sends a TCP packet with the ACK flag and a zero window size to differentiate open ports from filtered ones.
* maimon\_scan(target, port): It sends a specific TCP packet to identify open, closed or filtered ports.
* check\_os(target\_ip, system): This function uses Nmap to perform a OS detection on the target IP address and checks if the OS belongs to a specified system.
* print\_centered(text, color\_code): This function is used for printing text with a centered alignment and a specific color code. It retrieves the terminal width and calculates padding to center the text visually.
* print\_menu(): This function creates the menu displayed to the user of the tool. It uses the print\_centered function for the title options.
* main(): This function first calls the print\_menu() function and then asks the user for target IP. Next it takes port to scan from user and converts the input into integer. In this function next a while loop has been used that asks user to enter option for a scan and exits for 0 input.

# **Outputs**

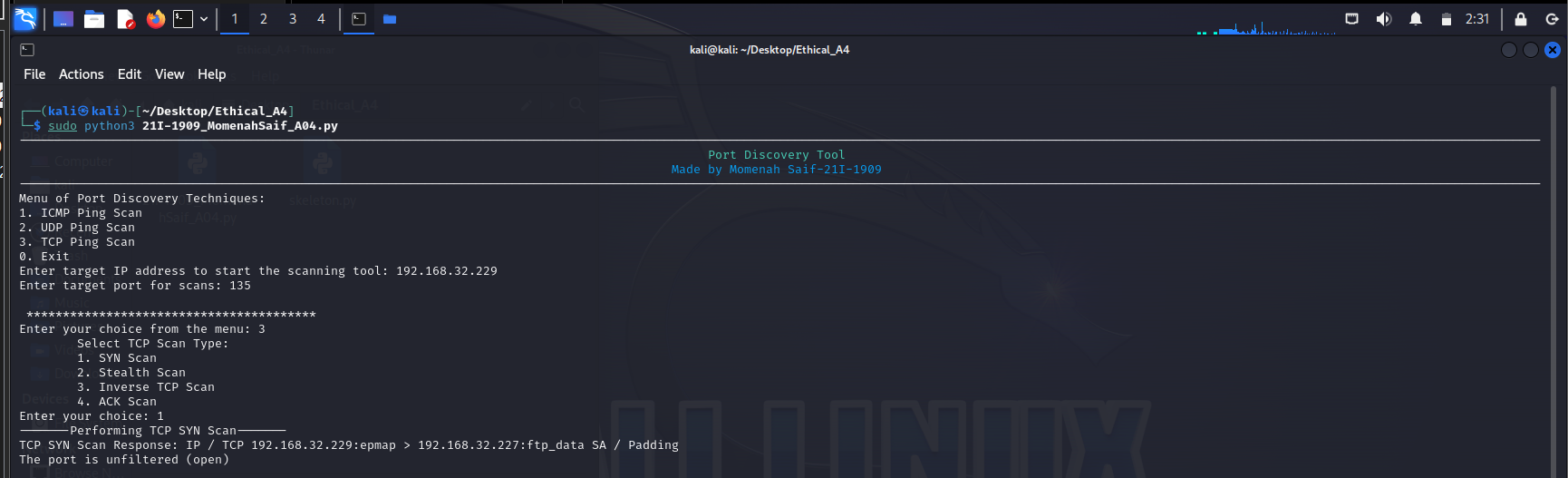
1. **ICMP Ping Scan**



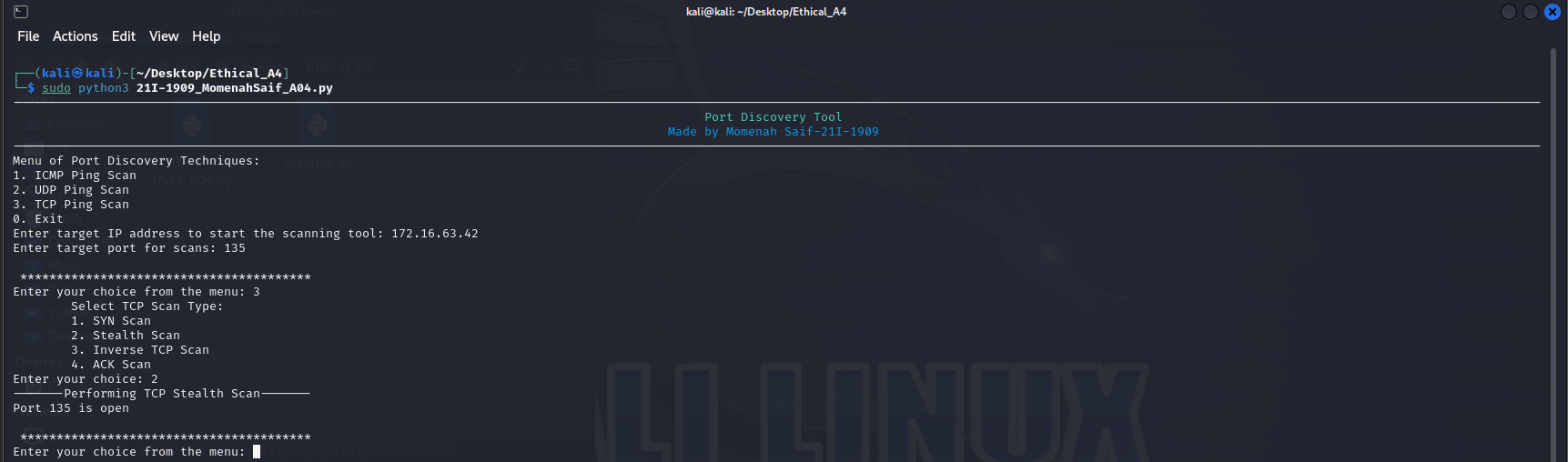
1. **UDP Ping Scan**



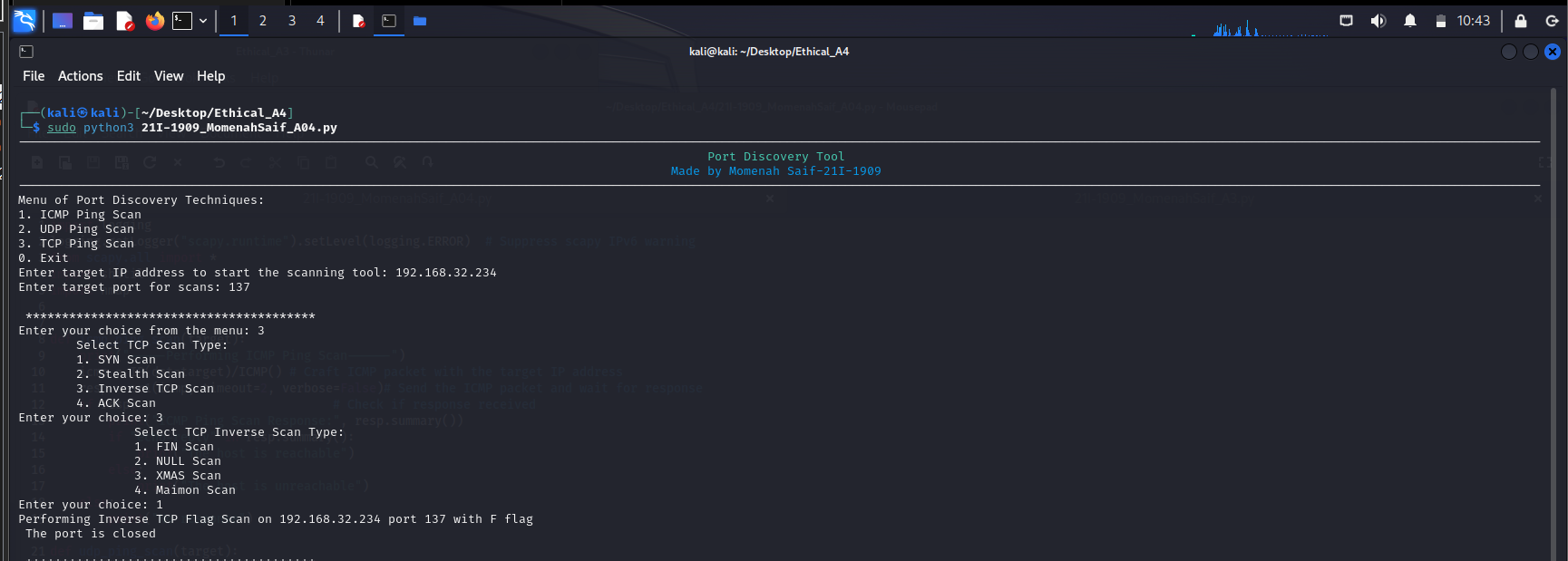
1. **TCP Scan-SYN Scan**



1. **TCP Scan-Stealth Scan**



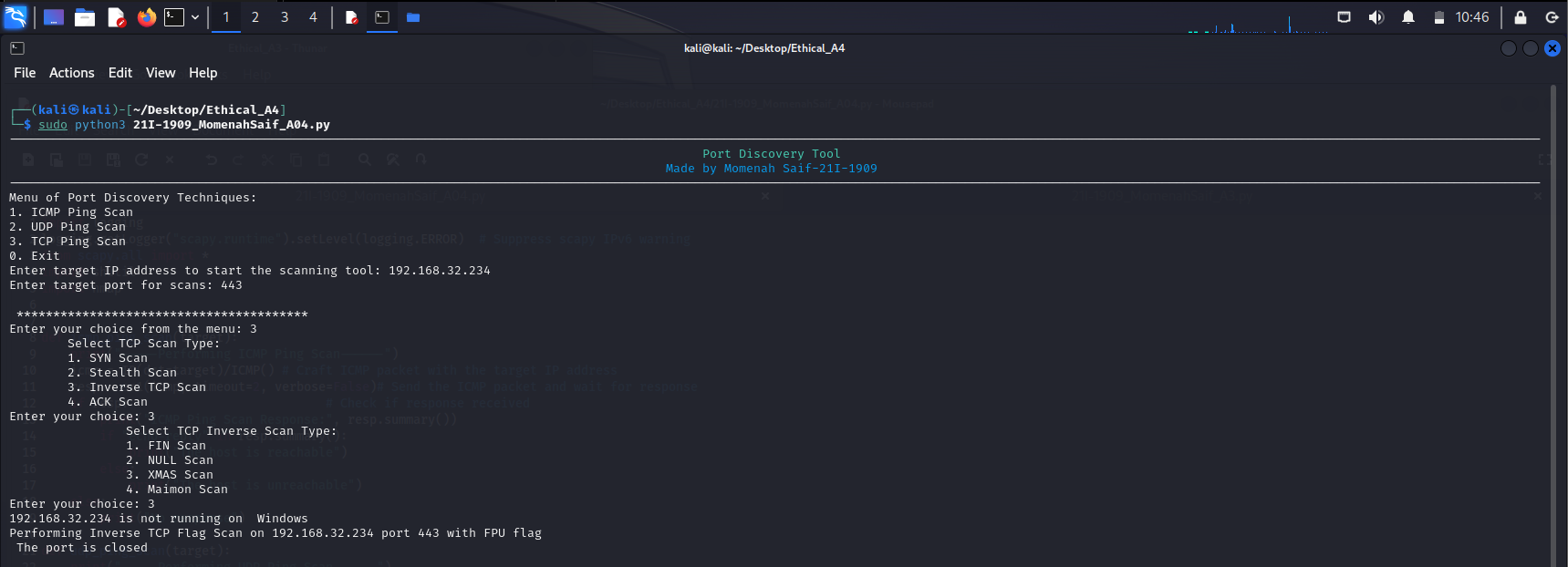
1. **TCP Scan-Inverse TCP Scan-Fin Scan**



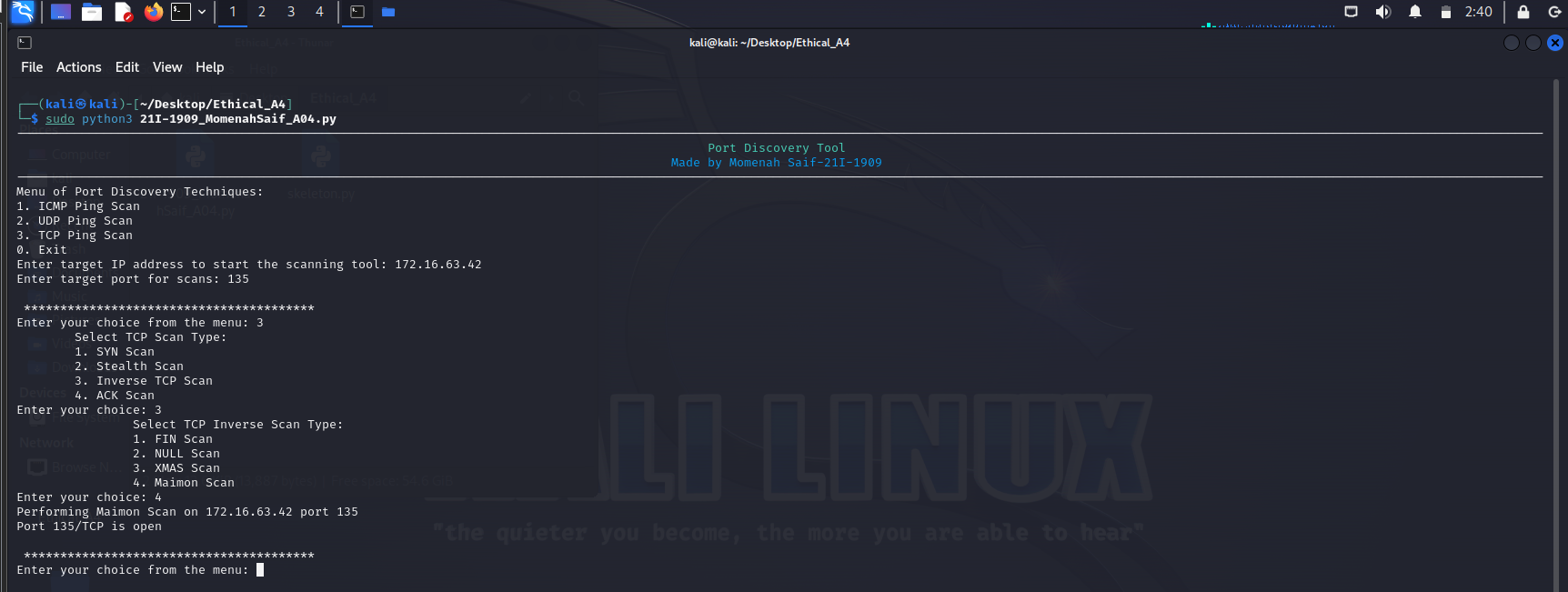
1. **TCP Scan-Inverse TCP Scan-Null Scan**



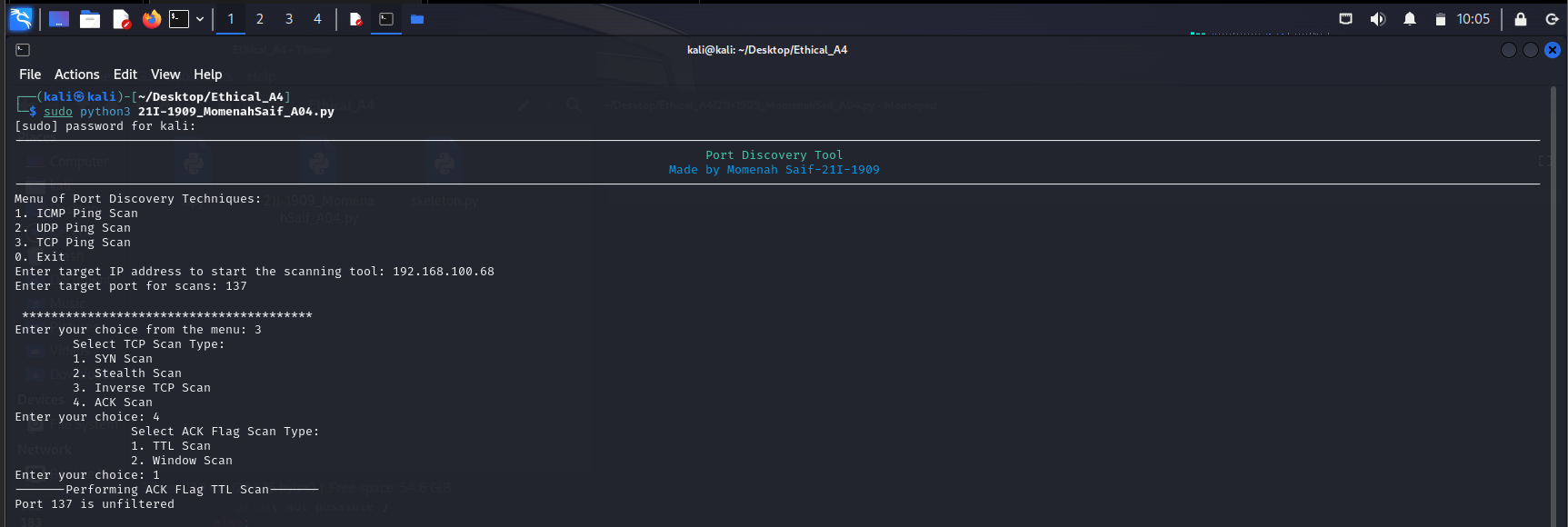
1. **TCP Scan-Inverse TCP Scan-XMAS Scan**



1. **TCP Scan-Inverse TCP Scan-Maimon Scan**



1. **TCP Scan-ACK Flag Scan-TTL Scan**



1. **TCP Scan-ACK Flag Scan-Windows Scan**



# **Summary**

We were given to make a port scanning tool using Python code. This code allows users to choose from various techniques to scan a target machine's ports and determine if they are open, closed, or filtered. The tool performs ICMP ping scans, UDP ping scans, and different types of TCP scans (SYN, Stealth, Inverse, ACK). It uses the Scapy library to craft and send packets, and the nmap library for optional OS detection. The code uses functions for each scanning method, menu display, and user interaction. The main function guides the user through the process, takes input for target and scan type, and calls the appropriate functions to execute the scan and display the results.

# **References**

Used chatgpt multiple times to get functions that works properly for each of the scan.

