

# CODTECH Cybersecurity Internship

## Task 4 - Packet Sniffer in Python

**Name:** Kodali Harshith

### Objective:

To create a simple packet sniffer using Python that listens to incoming network packets and displays their details in the terminal. This helps to understand how packet capturing works, which is essential in cybersecurity and network monitoring.

### Python Code Used:

```
import socket

def sniff_packets():
    # Create a raw socket and bind it to the public interface
    sniffer = socket.socket(socket.AF_INET, socket.SOCK_RAW, socket.IPPROTO_IP)

    # Bind to localhost IP (0.0.0.0 binds to all interfaces)
    sniffer.bind(("0.0.0.0", 0))

    # Include the IP headers in the captured packets
    sniffer.setsockopt(socket.IPPROTO_IP, socket.IP_HDRINCL, 1)

    print("\n[+] Sniffing incoming packets...\nPress Ctrl+C to stop.\n")

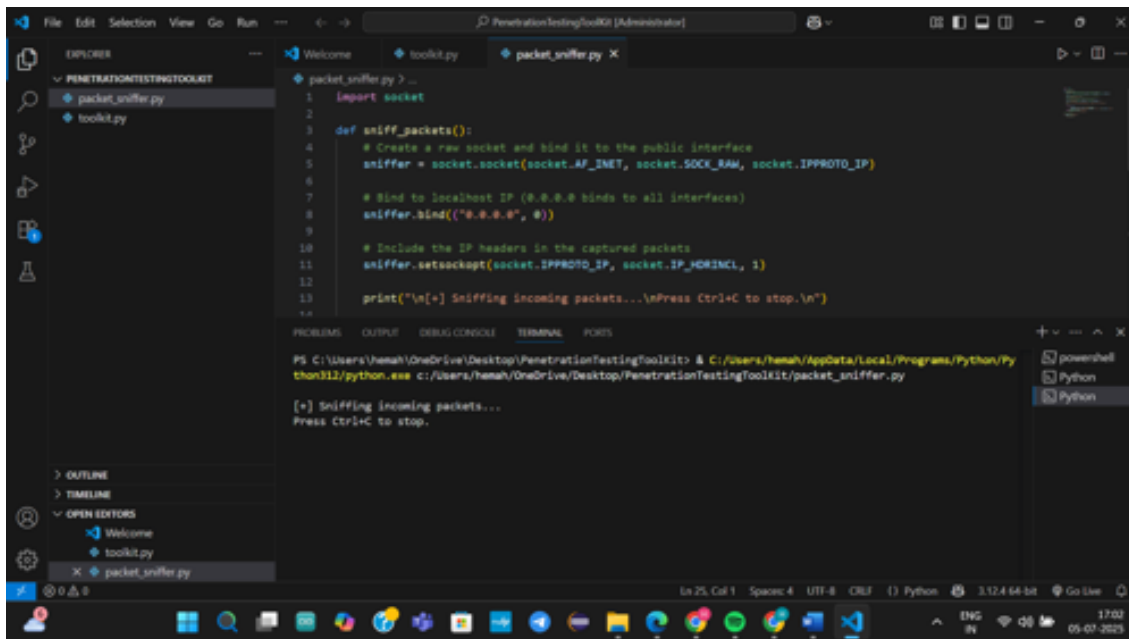
    try:
        while True:
            raw_data, addr = sniffer.recvfrom(65535)
            print(f"[Packet] Received from {addr[0]} | Size: {len(raw_data)} bytes")
    except KeyboardInterrupt:
        print("\n[+] Sniffing stopped.")

# Run the sniffer
if __name__ == "__main__":
    sniff_packets()
```

### Screenshot of Output:

# CODTECH Cybersecurity Internship

## Task 4 - Packet Sniffer in Python



The screenshot shows a Visual Studio Code editor window titled "PenetrationTestingToolKit [Administrator]". The Explorer pane on the left shows a folder named "PENETRATIONTESTINGTOOLKIT" containing two files: "packet\_sniffer.py" and "toolkit.py". The main editor area displays the code for "packet\_sniffer.py". The code is as follows:

```
1 import socket
2
3 def sniff_packets():
4     # Create a raw socket and bind it to the public interface
5     sniffer = socket.socket(socket.AF_INET, socket.SOCK_RAW, socket.IPPROTO_IP)
6
7     # Bind to localhost IP (0.0.0.0 binds to all interfaces)
8     sniffer.bind(("0.0.0.0", 0))
9
10    # Include the IP headers in the captured packets
11    sniffer.setsockopt(socket.IPPROTO_IP, socket.IP_HDRINCL, 1)
12
13    print("\n[+] Sniffing incoming packets...\nPress Ctrl+C to stop.\n")
14
```

Below the code editor, the TERMINAL pane shows the command prompt output:

```
PS C:\Users\hemah\OneDrive\Desktop\PenetrationTestingToolKit> & C:\Users\hemah\AppData\Local\Programs\Python\Python312\python.exe c:/Users/hemah/OneDrive/Desktop/PenetrationTestingToolKit/packet_sniffer.py

[+] Sniffing incoming packets...
Press Ctrl+C to stop.
```

The status bar at the bottom indicates the file is "packet\_sniffer.py" at line 25, column 1, using a UTF-8 encoding and a CRLF line ending. The Python version is 3.12.4 64-bit.

### Conclusion:

This task successfully demonstrated how a basic packet sniffer can be created using raw sockets in Python. It gave hands-on experience with socket programming and network interfaces. Understanding how packets are captured is essential for ethical hacking, intrusion detection, and network troubleshooting.