

Exp No: 14

## Packet Sniffing

Date: 09/10/24

AIM:

Implement packet sniffing using RAW sockets.

CODE:

from scapy.all import sniff

from scapy.layers.inet import IP, TCP, UDP, ICMP

def packet\_callback(packet):

if IP in packet:

up\_layer = packet[IP]

protocol = up\_layer.proto

src\_ip = ip\_layer.src

dst\_ip = ip\_layer.dst

protocol\_name = ""

if protocol == 1:

protocol\_name = "ICMP"

elif protocol == 6:

protocol\_name = "TCP"

elif protocol == 17:

protocol\_name = "UDP"

else:

protocol\_name = "unknown protocol"

```

print (" Protocol : {protocol_name}")
print (" Source IP : {src_ip}")
print (" Destination IP : {dst_ip}")
print (" - " + "\n")

```

```

def main():
    # Create socket
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    # Bind socket to the port
    sock.bind((" ", 50))
    # Listen for incoming connections
    sock.listen(1)
    # Accept incoming connection
    conn, addr = sock.accept()
    # Send data
    conn.sendall(" ".join(addr))
    # Close connection
    conn.close()

```

## OUTPUT :

```

Protocol : TCP
Source IP : 20.244.184.142
Destination IP : 172.20.10.2

```

```

Protocol : TCP
Source IP : 20.244.184.142
Dest IP : 172.20.244.184

```

Protocol : TCP

```

Source IP : 20.244.184.142
Dest IP : 172.20.10.2

```

## Result:

Thus, packet sniffing using RAW sockets is implemented successfully, and output is verified.