EX NO. 14
16/10/24

Packet Sniffing

AIM:

To write a code using RAW sockets to implement packet sniffing.

PROGRAM:

```
from scapy.all import sniff
from scapy.layers.inet import IP, TCP, UDP, ICMP

def packet_callback(packet):
    if IP in packet:
        IP_layer = Packet[IP]
        Protocol = IP_layer.proto
        Src_ip = IP_layer.src
        dst_ip = IP_layer.dst
    # Determine the protocol
    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 Packet details
    Print(f"protocol: {Protocol_name}")
    Print(f"Source IP: {Src_IP}")
    Print(f"Destination IP: {dst_IP}")
    Print("-" x 50
```

```
def main():
    sniff(iface = "wi-fi", prn = packet_callback,
          filter = "ip", store = 0)
if __name__ == "__main__":
    main()
```

O/P

Protocol : TCP
Source IP : 20.247.164.172
Destination IP : 172.20.10.2

Protocol : TCP
Source IP : 20.247.164.142
Destination IP : 172.20.10.2

Result :

Thus the Packet sniffing process is executed & output is verified successfully