Practical 14

AIM: - Write a code using RAW sockets to implement packet sniffing.

Algorithm:

```
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 name from protocol number
        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('-' * 30)

# Start packet capture on the default interface
sniff(prn=packet_callback, filter="ip")
```

Practical 14

Output:-

```
Protocol: TCP
Source IP: 192.168.1.5
Destination IP: 172.217.26.206
----------------------------
Protocol: ICMP
Source IP: 192.168.1.5
Destination IP: 8.8.8.8
----------------------------
Protocol: UDP
Source IP: 192.168.1.5
Destination IP: 224.0.0.251
----------------------------
Protocol: TCP
Source IP: 172.217.26.206
Destination IP: 192.168.1.5
----------------------------
```