10/10/25 14. RAW Sockets to implement packet sniffing

Aim

To write a code using Raw Sockets to implement packet sniffing.

Algorithm

1. Start sniffing network packets on the wi-fi interface using scapy.
2. for each captured packets, check if it contains an IP layer.
3. Extract the source IP, destination IP. and protocol number from the IP header.
4. Identify the protocol (ICMP, TCP, UDP or unknown) and print the details.

Program:

```
from scapy.all import sniff.
from scapy.layer.inet import IP, TCP, UDP
ICMP def packet - callback (packet):
    if IP packet:
        IP - layer : callback [packet]):
        Ip - layer = packet [IP]
        protocol = ip - layer.proto
        src - Ip = ip - layer proto
        dst - Ip = Ip - layer dst
     # Determine the protocol.
     protocol _ name = ' '
        if protocol == 1:
     protocol - name = "ICMP"
        elif protocol == 6:
     protocol - name = "T/P"
    elif protocol = = 17:
        protocol - name - " UDP "
else
    protocol - name - "unknown protocol"
        # print packet details.
print packet details
 print (f"protocol .{protocol - name}"}
 print (f"source ip : {src - ip - 3"]
 print (f 'Destination IP = dst - ip 3")
 print ("-' * 50)
# capture packets on the default network interface
sniff (iface = "wifi", prn = packet - callback.
   filter = "ip", store = 0)
```

Output

```
      protocol : Tcp
Source IP = 192 . 168 . 1 . 10
Destination IP = 172 . 217 . 160 . 142
- - - - - - - - - - - - - - - - - - - -
protocol = UDP
Source IP = 192 . 168 . 1 . 10
Destination IP = 8 . 8 . 8 . 8
- - - - - - - - - - - - - - - - - - - -
protocol = ICMP
Source ID = 192 . 168 . 1 . 10
Destination ID = 192 . 168 . 1 . 1 .
```

14/x 10/10

Result

Thus the code to implement packet sniffing using Raw sockets have been executed Succesfully.