

NAME:UDAYA SANKAR C

ROLL NO:231901058

## **EXP NO:12.A DATE:5.11.24 Packet Sniffing Using Socket**

### **AIM:**

To study packet sniffing concept and implement it using sockets.

### **Algorithm:**

**Import Libraries:** Import necessary modules from `scapy` for packet capturing and IP layers.

### **Define Packet Callback:**

- Check if the packet contains an IP layer.
- Extract protocol number, source IP, and destination IP from the IP layer.
- Identify the protocol type (ICMP, TCP, UDP) based on the protocol number.
- Print the protocol name, source IP, and destination IP.

### **Main Function:**

- Use `sniff` to capture packets on the default network interface.
- For each packet, call `packet_callback` to process and display packet information.

### **Run Program:**

- Execute the `main` function to start packet sniffing when the script runs.

### **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("-
" * 50) def main():
    # Capture packets on the default network interface

sniff(prn=packet_callback, filter="ip", store=0) if
__name__ == "__main__":
    main()

```

### Output:

Protocol: TCP

Source IP: 192.168.1.10

Destination IP: 93.184.216.34

-----

Protocol: ICMP

Source IP: 192.168.1.10

Destination IP: 8.8.8.8

-----

Protocol: UDP

Source IP: 192.168.1.10

Destination IP: 8.8.4.4

----- Protocol:

TCP

Source IP: 192.168.1.10 Destination

IP: 172.217.14.206

-----**Result:**

Packet sniffing concept and implement it using sockets is studied and successfully executed.