

Ex. No: 14

## Packet Sniffing

Date: 22-10-24

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:

ip\_layer = packet[10]

protocol = ip\_layer.proto

src\_ip = ip\_layer.src

dest\_ip = ip\_layer.dst

protocol\_name = ""

if protocol == 1:

protocol = 1:

protocol\_name = "ICMP"

elif protocol == 6:

protocol\_name = "TCP"

elif protocol == 17:

protocol\_name = "UDP"

else

protocol\_name = "Unknown Protocol"



```

printf (" Protocol: %s\n", protocol_name);
printf (" Source IP: %s\n", src_ip);
printf (" Destination IP: %s\n", dest_ip);
printf (" " * 50);
def main():
    iface = 'lo-fi', port = 8080
    host = 'ip', src = 0
    if __name__ == '__main__':
        main()

```

OUTPUT:

```

[01] testprog - refused - q1
Protocol: TCP
Source IP: 20.247.184.142
Destination IP: 172.20.247.134
Protocol: TCP
Source IP: 20.247.184.142
Dest IP: 172.20.247.134
Protocol: TCP
Source IP: 20.247.184.180
Dest IP: 172.20.10.2

```

Result:

Thus Packet sniffing using  
RAW packets was implemented

Ex. No: 126  
Date: 2.11.24

Aim:

Implement sockets

CODE:

server.py

import socket

def server():

port = 8080

host = '0.0.0.0'

with socket

server

S. bind

while

d, a

print

a =

S.

af