

25/12/25

write a code using RAW sockets

to implement packet sniffing

Aim:

(C:\> & "q" = 0/13

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

Algorithm :

```

from scapy.all import sniff
from scapy.layers.net 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
        dist_ip = ip_layer.dist
        protocol_name = "Unknown"
        if protocol == 1:
            protocol_name = "ICMP"
        elif protocol == 6:
            protocol_name = "TCP"
        elif protocol == 17:
            protocol_name = "UDP"
        else:
            protocol_name = "Unknown protocol"
        print(f"Protocol: {protocol_name}")
        print(f"Source IP: {src_ip}")
        print(f"Destination IP: {dist_ip}")
        print(f"- {src_ip} {dist_ip}")

```

print packet details

```

    pointf("Protocol : {protocol.name}")
    pointf("Source IP : {src_ip}")
    pointf("Destination IP : {dist_ip}")
    pointf("- " + so)

```

11.1.2018 - 26

Lab interface 9 giving back a return

Sniffing iface == "wlan0"; port's packet - callback
filter = "ip", size > 0

ob Input: was given about a new OT
pinging = server (ping) \rightarrow normally in
output:

Protocol: TCP
Link layer: No queue no cost

Source IP: 192.168.1.5 Port: 5000
Destination IP: 172.17.0.28 Port: 5000

- - - - - : backlog in q5_qi - - - - -

Protocol: ICMP seq = 0x00_qi

Source IP: 192.168.1.5 Destination

Destination IP: 192.168.1.5 Port: 5000

- - - - - : backlog in q5_qi - - - - -

Protocol: UDP source: 192.168.1.5

source IP: 192.168.1.5 Destination

Destination IP: 224.0.0.1 Port: 5000

"q5_qi" = max. backlog

: 10 = backlog size

"q5_UV" = max. backlog

: 9216

"backlog window" = max. backlog

Result: Max backlog trying #

(forwarding): backlog "3" tries
This code using raw sockets to
implement packet sniffing executed successfully.
(q5_hacks): q5 initialized "7" tries
(or "1" - "1" tries)