

---

## TASK ----1

### Python coding :

```
from scapy.all import sniff, IP, TCP, UDP, Raw, Ether
def analyze_packet(packet):
    print("\n===== PACKET =====")
    if Ether in packet:
        eth = packet[Ether]
        print(f"Ethernet: {eth.src} -> {eth.dst}")
        print(f"Type: {hex(eth.type)}")
    if IP in packet:
        ip = packet[IP]
        print(f"Source IP: {ip.src}")
        print(f"Destination IP: {ip.dst}")
        print(f"Protocol: {ip.proto}")
    if TCP in packet:
        tcp = packet[TCP]
        print("TCP Segment:")
        print(f" Source Port: {tcp.sport}")
        print(f" Destination Port: {tcp.dport}")
        print(f" Flags: {tcp.flags}")
    if UDP in packet:
        udp = packet[UDP]
        print("UDP Datagram:")
        print(f" Source Port: {udp.sport}")
        print(f" Destination Port: {udp.dport}")
    if Raw in packet:
        payload = packet[Raw].load
        print(f"Payload ({len(payload)} bytes):")
        print(payload[:100]) # print first 100 bytes to keep output readable
def main():
    print("Starting packet sniffer... (Press CTRL+C to stop)")
    sniff(prn=analyze_packet, store=False)
if __name__ == "__main__": main()
output:
```

```
===== PACKET =====
Ethernet: 8c:85:90:1a:2b:44 -> dc:a6:32:c9:55:10
Type: 0x800
Source IP: 192.168.1.10
Destination IP: 142.250.64.110
Protocol: 6
TCP Segment:
Source Port: 53214
Destination Port: 443
```

Flags: S  
Payload (0 bytes):