

Cyber Security Tasks — CodeAlpha

Task-1 Basic Network sniffer

- Build a Python program to capture network traffic packets.
- Analyze captured packets to understand their structure and content.
- Learn how data flows through the network and the basics of protocols.
- Use libraries like `scapy` or `socket` for packet capturing.
- Display useful information such as source/destination IPs, protocols and payloads.

```
GNU nano 8.6                                         sniffer.py
from scapy.all import sniff, IP, TCP, UDP, Raw

def packet_analyzer(packet):
    if IP in packet:
        print("\n===== Packet Captured =====")
        print(f"Source IP      : {packet[IP].src}")
        print(f"Destination IP : {packet[IP].dst}")
        print(f"Protocol       : {packet[IP].proto}")

        if TCP in packet:
            print("Protocol Type   : TCP")
            print(f"Source Port     : {packet[TCP].sport}")
            print(f"Destination Port: {packet[TCP].dport}")

        elif UDP in packet:
            print("Protocol Type   : UDP")
            print(f"Source Port     : {packet[UDP].sport}")
            print(f"Destination Port: {packet[UDP].dport}")

        if Raw in packet:
            print(f"Payload         : {packet[Raw].load}")

    print("Starting Network Sniffer ... ")
    sniff(prn=packet_analyzer, store=False)
```

To develop a Python-based network sniffer using Scapy that captures live network packets and analyzes their source, destination, protocol, and payload to understand how data flows across a network.

