Cybersecurity Packet Sniffer with GUI

Rahul Kumar - BSCSY015

Name: Rahul Kumar

Roll No: BSCSY015

Department: Cybersecurity

University: Mehran University of Engineering and Technology

Project Type: Desktop-based Application (Python)

Date: August 2025

Objective

To develop a GUI-based network packet sniffer that allows real-time monitoring, filtering, analysis, and export of network traffic data for cybersecurity purposes.

Tools & Technologies

- Programming Language: Python
- Libraries Used:
  - scapy - for packet sniffing
  - tkinter - for building the GUI
  - matplotlib - for pie chart summary
  - json, csv - for log handling
  - geoip2 - for IP geolocation
- Platform: Windows

Key Features

Real-Time Sniffing

- Captures packets using filters (e.g., DNS, HTTP, HTTPS).
- Displays live packet logs in a scrollable GUI box.

Admin-Only Mode

- Password protected: Only authorized users (password: bscys015) can access the GUI.

Export Features

- Export Logs to CSV: Converts network_log_fixed.json to exported_logs.csv.
- Date Filter: Filter logs by a specific date (e.g., 2025-08-01) and save as JSON.

Graphical Summary

- Pie chart visualization of traffic types (DNS, HTTP, etc.).

Alerts & Logging

- Detects:
  - Suspicious domains (e.g., malicious.com)
  - Malicious IPs
- Saves alerts to alerts.json.

Editable Blacklist

- Allows the admin to update domain/IP blacklists via GUI dynamically.

Personalization

- Title: "Cybersecurity Packet Sniffer - GUI Edition | Rahul Kumar - BSCSY015"
- Footer: "Developed by Rahul Kumar - BSCSY015"

File Structure

```
basic-networking-project/
 packet_sniffer.py
 network_log.json
 network_log_fixed.json
 exported_logs.csv
 alerts.json
 GeoLite2-City.mmdb
```

Output Example

Live packet log shows:


```
{
  "timestamp": "2025-08-02 14:45:12.351237",
```

"summary": "DNS 192.168.10.5 > 8.8.8.8: google.com",
"type": "DNS",
"query": "google.com",
"src_ip": "192.168.10.5",
"dst_ip": "8.8.8.8"
}

Pie Chart Output:

- DNS: 65%
- HTTP: 30%
- Others: 5%

Testing

- Tested by simulating traffic (DNS, HTTP).
- Blacklisted domain and IP detection verified.
- Export and filter logs successfully.

Conclusion

This project provides a functional, user-friendly, and customizable packet sniffer tool with security features. It's suitable for beginner to intermediate cybersecurity students and can be extended to support deeper protocol analysis or cloud storage integration.