Deliverable 1 – Project Proposal & Conceptual Design

Project Title: Cyber Network Threat Analyzer (Graph-Based Security Analysis System)

Instructor: Sir Zubair Ahmed

Course: Data Structures and Algorithms (CS221)

NAME	REG: NO
Abdullah Munir	2024042
Hasnain Zia Wazir	2024222
Hadeed Haider	2024189

1. Problem Description and Aim

In the modern cybersecurity landscape, detecting malicious activity within a network requires rapid and accurate analysis of connections between devices. Traditional security systems often fail to represent network interactions in a structured and analyzable way. Our project aims to develop a Cyber Network Threat Analyzer that models a computer network as a graph, where each node represents a device or user and edges represent network connections. By applying Data Structures and Algorithms (DSA), the system will analyze this graph to detect anomalies, identify high-risk nodes, and trace possible attacks or infection paths.

2. Key Data Structures

The project will utilize several core data structures to ensure efficient representation and analysis of the network:(These Data Structure can be varied as well according to our need)

• Graph (Adjacency List) – To represent the overall network connections between devices.

• Queue – Used in Breadth-First Search (BFS) for tracing attack spread or shortest connection paths.

• Stack – Used in Depth-First Search (DFS) for exploring compromised subnetworks.

• Hash Map – To store information such as device ID, IP address, and threat score for fast lookup.

• Priority Queue – To rank devices by their threat level or connection degree.

• Set – To keep track of visited nodes during graph traversals.

3. Main Algorithms

The Cyber Network Threat Analyzer will integrate the following algorithms for analyzing network activity:

• Breadth-First Search (BFS) – To find the shortest attack or connection path between devices.

• Depth-First Search (DFS) – To detect clusters of compromised or connected nodes.

• Sorting Algorithms – To identify and rank the most vulnerable or highly connected devices.

• Centrality Calculation – To determine which nodes have the highest degree and thus higher attack exposure.

4. Data Flow: Input, Processing, and Output

• Input: The system will take network data as input, either manually entered or loaded from a text file. Each line will define a device and its connections (e.g., A-B, B-C).

• Processing: The program will construct a graph using adjacency lists and apply algorithms such as BFS and DFS to identify anomalies, suspicious connections, and possible attack paths.

• Output: The analyzer will produce a report showing detected high-risk devices, infection spread simulations, and shortest attack routes. Results can be displayed textually on the console or optionally visualized using ASCII or Graphviz representations.

5. Integration with Course Concepts

This project integrates the key Data Structures and Algorithms concepts taught (or will be taught) in CS221, including:

• Graphs

• Queues & Stacks

• Hash Tables

• Sorting & Searching

• Complexity Analysis

6. Conclusion

The Cyber Network Threat Analyzer project will serve as a practical demonstration of how Data Structures and Algorithms can be effectively utilized in cybersecurity to identify vulnerabilities, detect threats, and enhance network resilience. It not only strengthens DSA understanding but also builds foundational skills for threat analysis and secure system design.