CHAPTER 1: INTRODUCTION

1.1 Overview

Network monitoring is the process of using logging and analytic tools to properly determine traffic flows, utilization, and other network performance metrics. Good network monitoring solutions provide real figures as well as graphical aggregate representations of the network's condition. This allows you to see exactly what's going on and determine where improvements might be required [1].

There was an additional amount of work and studies whose aim was to analyze and monitor the network. An overview of the patterns was provided in a recent study by [2] who elucidated processes including uncovering important and influential network actors, elucidating clandestine organizational structures, and detecting fraudulent or illicit efforts. One of our primary advantages is to add to the discussion [2] by including new network monitoring papers and teasing out colored patterns in the environment for a wide range of dependent work in synthetic process control and public health monitoring. Social network monitoring techniques are often illustrated using terrorist networks such as Al Qaeda (see Figure 1.1)

Figure 1.1: Illustration of the terrorist network. From Krebs (2002).

The successful functioning and security of network dispatches are complete in the dynamic and linked geography of ultramodern computing. Network monitoring and analysis are critical for ensuring the smooth operation of networks, identifying hidden vulnerabilities, and diagnosing performance difficulties [4]. To attain these goals, experts rely on technologies

that provide a thorough understanding of network business patterns and behaviors. The Scapy Library [4] is a comparable useful utility.

1.2 Problem Statement

Data communication networks are a critical area of study in computer science [5], [6]. Computers no longer exist as stand-alone equipment; computer technology is utilized to communicate, and there is now nearly no area on the planet where you cannot communicate using this technology [7]. Cloud computing, grid computing, virtualization, multiprocessing, and other advanced computing technologies rely on data networks.

Because of their greater reliance on cutting-edge computation and technology, networks are becoming larger, more sophisticated, and more diverse. Learning management systems, e-health, online banking, and online ticketing are just a few examples of network-based technology and services. Users do not want to sacrifice the quality or availability of the services they utilize. Because of this evolution, network complexity has risen; as a result, network administration takes a considerable deal of attention [8]. Network management refers to the actions involved in administering a network, the most important of which is monitoring [8].

There are several factors that might cause a network to fail; whether the cause is deliberate or accidental, only two factors can save the system from downtime: redundancy and monitoring [9]. Network monitoring is an essential aspect of network management [10] [11] because it provides vital data about the network, revealing information about the network's infrastructure and health, and it is also utilized for the majority of network administration duties [12] [13]. Since the advent of the first computer network [14], network monitoring has been seen as a critical building component for many areas of communication networks.

Table 1.1: Example of a Table Caption

Date	Time Range	Total Packets	Average Data Rate	Top Protocol	Top Source IP	Top Destination IP
2023-08-15	09:00-10:00	128,764	6.2 Mbps	HTTP	192.168.1.10	208.67.217.231
2023-08-15	10:00-11:00	156,329	7.8 Mbps	DNS	192.168.1.20	8.8.4.4
2023-08-15	11:00-12:00	142,511	6.9 Mbps	SMTP	192.168.1.30	173.194.205.26

1.3 Aims and Objectives

In today's increasingly complex networking world, effective network traffic monitoring and analysis are critical for sustaining network environments' security, performance, and dependability. This solution proposal describes how to create a network monitoring and analysis project in Python using the Scapy package. We can acquire deep insights into network operations, spot abnormalities, and respond to possible attacks quickly by exploiting Scapy's capabilities.

We will use Real-time Packet Capture: to create a Python script utilizing Scapy to capture network packets in real-time from a specified network interface, and Packet Analysis and Decoding: to analyze and decode network packets. Implement packet analysis routines to decode and dissect captured packets, extracting vital information such as source and destination IP addresses, protocol types, payload data, and Traffic Visualization to create visual representations of network traffic patterns, protocol distributions, and data flow to provide a comprehensive overview of network activities, and finally, Data Logging and

Storage to capture and store captured and analyzed data in a structured manner, allowing historical analysis.