South Asian Journal of Engineering and Technology Research Article

Open Access Full Text Article 133(2023) 1-9 DOI: 10.26524/sajet.2023.13.10

Network traffic analysis and alerting system

Dinesh Gopal Dommeti*

*BTech CSE Adikavi Nannaya University, College of Engineering, Rajmahendravaram, AP, India.

Corresponding author.

Correspondence: Dinesh Gopal Dommeti

E-mail: 198297601012@aknu.edu.in

Article info

Received 28th May 2023 Received in revised form 30 July 2023

Accepted 20 November 2023

Keywords

Network Traffic Monitoring, Alerting System, Network Security, Performance Optimization, Data Analysis

https://sajet.in/index.php/journal/article/view/288

Abstract

A network traffic monitoring and alerting system is essential for network administrators. It ensures the security and performance of their networks by capturing and analyzing all data passing through the network. This system identifies potential security threats and network problems, sends alerts to the administrator, and protects organizations and users from potential risks. In summary, this system is crucial for maintaining network security and performance, helping administrators detect and resolve issues, and optimizing resources to improve the user experience.

1. INTRODUCTION

In today's modern organizations, networks have become an essential component of daily operations. With the increasing use of technology and the growing reliance on networks, it is crucial to have a system in place to ensure the network's security and performance. This is where network traffic monitoring and alerting systems come in.

Network traffic monitoring and alerting systems provide network administrators with the visibility and insights needed to maintain network security and performance. These systems capture using various tools like Wireshark and analyze all data passing through the network, enabling administrators to detect and prevent potential security threats, such as malware or phishing attempts, and eliminate unnecessary traffic that can slow down the network [4].

The collected data can also be used for forensic analysis, incident response, and troubleshooting. It provides insights into network usage patterns, which can be used to optimize network resources and improve the overall user experience. In short, network traffic monitoring and alerting systems play a vital role in ensuring the security and performance of modern

networks, helping organizations protect their assets and users from potential security threats [6].

Our research paper aims to develop a system that addresses these needs by capturing, analyzing, and alerting network traffic to identify users accessing blacklisted websites, peer-to-peer traffic, and unnecessary protocol usage. By identifying and alerting these types of traffic, we can prevent security threats and optimize network resources, thereby improving the overall user experience [7].

2. SYSTEM ARCHITECTURE

System architecture is a conceptual model that defines the structure, behavior, and views of a system. An architecture description is a final formal description and representation of a system, organized in a way that supports reasoning about the structure and behavior of the system.

A representation of a system, including a mapping of the functionality onto hardware and software components, a mapping of the software architecture onto the hardware architecture, and human interaction with these components [2].

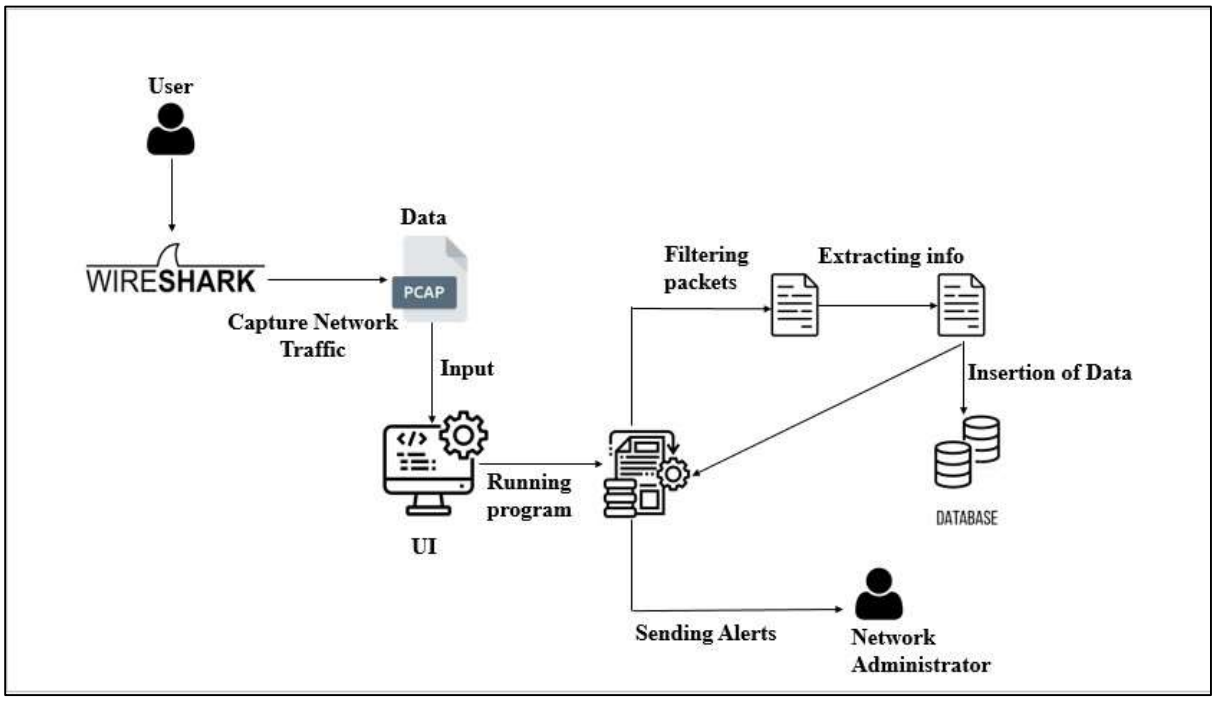


Figure 1 System Architecture

3. METHODOLOGY

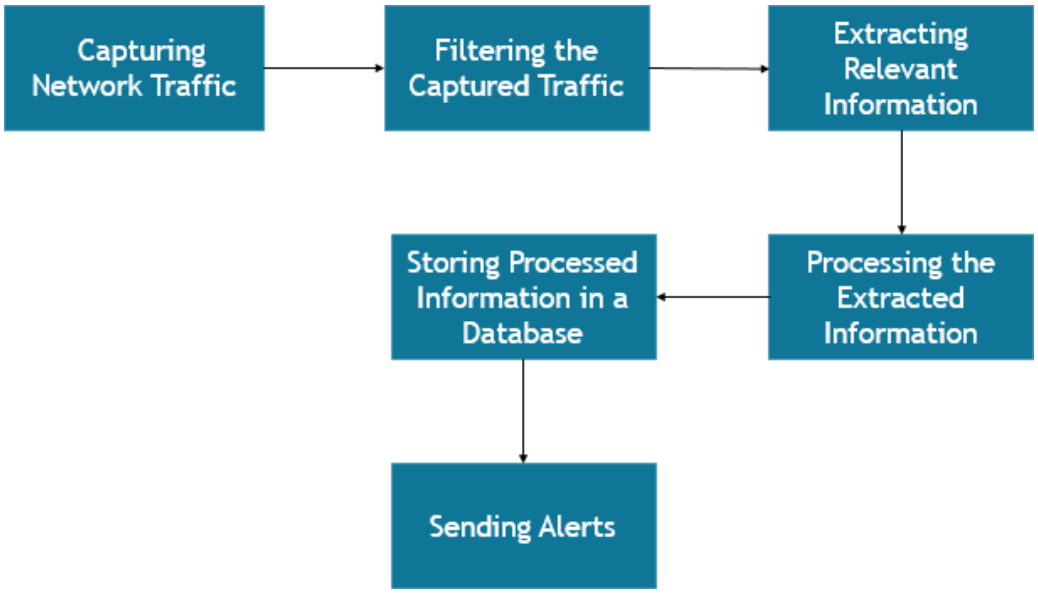
The implementation methodology involves:

1. **Requirements Gathering:** Consulting with network security experts to identify key system features.
2. **Design:** Creating a system design using Python and the Scapy library for packet capture and analysis, a database for packet storage, and email alerts for notifications.
3. **Implementation:** Developing the system to capture, filter, and analyze network packets, store relevant data, and send alerts.

4. **Testing:** Simulating various network conditions to ensure system functionality.
5. **Deployment:** Deploying the system in a live environment and providing a user manual.

4. IMPLEMENTATION

4.1 SYSTEM IMPLEMENTATION:



Block Diagram

4.2 Implementation Process

The implementation of the network traffic analysis [9] and alerting system involves several key steps, each crucial for ensuring the system functions as intended to enhance network security and performance. The process can be broadly divided into the following stages:

Installation of Required Libraries: The system utilizes several libraries that are essential for its operation. Key libraries include:

1. **Scapy:** This Python library is fundamental for packet manipulation and analysis. It allows the system to capture and filter network packets based on specific criteria such as IP addresses, ports, and protocols.
2. **Pybloom_live:** A library used for efficient set membership testing. In this system, it helps in quickly checking captured IP addresses against a predefined blacklist, facilitating real-time detection of blacklisted IPs.
3. **MariaDB:** An open-source relational database management system used to store captured network packets. The database is configured to support detailed analysis by storing relevant information such as packet source and destination IPs, ports, and packet sizes.
4. **SMTP:** Utilized for sending email alerts. The system uses this protocol to notify administrators of any suspicious activities or security threats.

5. **Other Libraries:** Additional libraries like `os`, `SSL`, and `tabulate` are used for system operations, secure communications, and formatting data respectively.
6. **Configuration of MariaDB Database:** The MariaDB database is set up to store captured network traffic data. The database schema is designed to include tables for storing packet information such as IP addresses, port numbers, timestamps, and packet sizes. This structured storage allows for efficient querying and analysis, enabling administrators to quickly identify patterns and potential security issues.
7. **Defining Packet Filters:** Packet filters are crucial for focusing the system's analysis on relevant data. Filters are defined based on specific IP addresses, ports, and protocols that are of interest. For instance, filters might target known malicious IP addresses, unusual port activity, or specific protocols that are generally not used within the organization's network. This selective filtering reduces the processing load on the system and focuses resources on analyzing potentially harmful traffic.
8. **Capturing and Filtering Network Packets:** Using the Scapy library, the system captures network traffic and applies the predefined filters. This step is critical as it determines which data will be stored and analyzed. The captured packets are processed in real-time, allowing the system to detect and respond to potential threats as they occur [1].
9. **Storing Captured Packets in MariaDB:** Once packets are captured and filtered, the relevant data is stored in the MariaDB database. This storage process includes recording details such as the packet's source and destination, port numbers, and packet size. Storing this information enables further analysis and is essential for generating reports, conducting forensic analysis, and understanding network usage patterns [8].
10. **Generating Email Alerts to System Administrators:** The system is designed to alert administrators immediately upon detecting suspicious activity. When certain conditions are met—such as detecting traffic from a blacklisted IP or encountering unusually large packet sizes—the system generates and sends an email alert. These alerts include details of the detected event, such as the involved IP addresses, the nature of the traffic, and the time of occurrence. The use of the SMTP library facilitates the sending of these alerts, ensuring that administrators are informed in real time.
11. **Logging and Reporting:** The system maintains logs of all activities, including captured packets, filter applications, and sent alerts. These logs are invaluable for troubleshooting, understanding system performance, and auditing security incidents. The tabulate library is used to format log data and analysis results into a readable and accessible format, providing clear insights into network traffic trends and potential security issues.

By integrating these components, the network traffic analysis and alerting system offers a comprehensive solution for maintaining network security and performance. The system not only detects and alerts administrators about potential threats but also provides a platform for in-depth analysis and continuous monitoring, essential for protecting organizational assets and data [5].

5. Dataset

The dataset comprises network connection data packets saved as PCAP files. These files contain comprehensive network traffic data, including packet headers, payload, and metadata, essential for developing and testing the network traffic analysis and alerting system.

6. DATABASE

6.1 Database Schema Overview

The database schema consists of a single table designed to store critical network packet information. Key fields include:

- **id:** A unique identifier for each record, set as the primary key and auto-incremented.
- **src_ip:** The source IP address from which the packet originated.
- **dst_ip:** The destination IP address to which the packet is sent.
- **trans_layer:** The transport layer protocol used, such as TCP, UDP, or ICMP.
- **src_port:** The source port number associated with the packet.
- **dst_port:** The destination port number associated with the packet.
- **protocol:** The protocol type, such as HTTP, DNS, or SMTP.
- **timestamp:** The date and time when the packet was captured.
- **packet_len:** The length of the packet in bytes.

This schema is structured to facilitate efficient data storage and retrieval, enabling detailed analysis of network traffic and enhancing the ability to identify and respond to potential security threats.

Conclusion

The proposed system for network traffic analysis and alerting presents an efficient and customizable solution for monitoring and analyzing network traffic [3]. Its design allows for filtering network packets based on specific criteria such as IP addresses, ports, and protocols, and stores relevant packets in a database for detailed analysis [10]. Additionally, the system can send email alerts to administrators when certain conditions are detected and generate customizable reports and analytics based on the captured data. This solution is not only cost-effective and scalable but also provides a robust alternative to commercial systems. It aims to empower system administrators to detect and prevent potential security threats, offering real-time alerts and comprehensive insights into network activity. This system stands out as a valuable tool for organizations seeking to enhance their network security and monitoring capabilities.

ACKNOWLEDGEMENT

I am thankful for the opportunity to pursue my Bachelor of Technology degree in the Department of Computer Science and Engineering at the University College of Engineering, Adikavi Nannaya University. I sincerely thank my research advisor, Dr. V. Persis, for her invaluable guidance and support throughout my research. I also extend my gratitude to Dr. P. Venkateshwara Rao, Principal, for his continuous encouragement, and to Dr. B. Kezia Rani, Head of the Department, for her assistance. I am also grateful to Dr. D. Latha, the Research

Coordinator, and the Review Committee Members, as well as the entire faculty of the Department of Computer Science and Engineering, for their constructive feedback and support.

Funding

No funding was received to carry out this study.

REFERENCES

1. S. Ansari, Rajeev S.G. and Chandrasekhar H.S, "Packet Sniffing: Brief Introduction", IEEE Potentials, Dec 2002- Jan 2003, Volume: 21 Issue: 5, pp: 17 – 19.
2. Qadeer M.A., Zahid M., Iqbal A., Siddiqui M.R "Network Traffic Analysis and Intrusion Detection Using Packet Sniffer" ICCSN '10 Second International Conference, 2010, Page(s): 313 – 317.
3. G. Varghese, "Network Algorithmic: An Interdisciplinary Approach To Designing Fast Networked Devices", San Francisco, CA: Morgan Kaufmann, 2005.
4. A. Dabir, A. Matrawy, "Bottleneck Analysis of Traffic Monitoring Using Wireshark", 4th International Conference on Innovations in Information Technology, 2007, IEEE Innovations '07, 18-20 Nov. 2007, Page(s):158 – 162.
5. Bo Yu" Based on the network sniffer implement network monitoring Computer Application and System Modeling (ICCASM), 2010 International Conference on Volume: 7, 2010, Page(s): V7-1 - V7-3.
6. Peel, R.M.A., "TCP/IP Networking Using Transputers", Proceedings of the 3rd Conference of the North American Transputer Users Group, IOS Press, April 1990, pp. 27-38.
7. Abdullah Konak. "Combining network reductions and simulation to Estimate network reliability", Proceedings of the 39th conference on Winter simulation, pp. 2301-2305, 2007.
8. Hornig, C., "A Standard for the Transmission of IP Datagrams over Ethernet Networks", RFC-894, Symbolic Cambridge Research Center, April 1984.
9. Kurose, James & Ross, Keith, "Computer Networking", Pearson Education, 2005.
10. G. Combs, "Ethereal", http://www.wireshark.org (Aug 15, 2007).