International Journal of Sciences and Innovation Engineering
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
website: www.ijsci.com
Volume 3 Issue 5 , May 2026
E-ISSN : 3049-0251
DOI: https://doi.org/10.70849/ijsci

A System for Monitoring and Analysing LAN Network Traffic

PRAJJWAL SRIVASTAVA[1], AKASH KUMAR YADAV[2], KUMAR AYUSH RANJAN[3], ABHIMANYU MOHNTY[4], VIBHA MANI[5], VIJAY SHUKLA[6]
[1,2,3,4,5,6] *Department of CSE-AI, GNIOT, Greater Noida, India*

Article Info

Article History:

Published: 09 May 2026

Publication Issue:
Volume 3, Issue 5
May-2026

Page Number:
295-303

Corresponding Author:
PRAJJWAL SRIVASTAVA

Abstract:

A network utilizing a LAN Telemetry System improves the performance and security of the network by offering comprehensive and continuous visibility and intelligence into the operation of the LAN to ensure a healthy and secure environment. This research explores the design, implementation, and evaluation of a scalable Local Area Network (LAN) telemetry system aimed at enhancing network performance and security. It addresses the complexities of data transfer in modern business environments, emphasizing the need for real-time monitoring and anomaly detection. The proposed work utilizes a three-tier architecture for data acquisition, processing, and visualization, incorporating tools such as libpcap for packet capturing and InfluxDB for time-series data storage. Evaluations indicate that the system achieves a high true positive rate of 98.5% in detecting security threats, while maintaining a low false positive rate below 2.5%. The study highlights the effectiveness of statistical anomaly detection in identifying potential security breaches and optimizing network performance, albeit acknowledging challenges in adapting to wider networks and encrypted communications. Future improvements may include machine learning applications to enhance threat detection capabilities.

Keywords: LAN Traffic Monitoring, Network Security, Performance Optimization, Anomaly Detection, NetFlow, Deep Packet Inspection, Real-Time Analysis, Network Health, System Architecture

1. INTRODUCTION

Data transfer and resource access which are absolutely important for smooth running of businesses in recent times, is made possible by Local Area Network (LAN). Cloud computing and bandwidth-hungry applications are also prevalent today, and traffic is becoming too complex for existing systems to manage. Today, network managers are being forced with two key orders of business: - Cut costs on the network and secure it from sophisticated insider attacks without any supporting data. Transit can mask root behavioural signals, and if there is no real-time telemetry the observable coverages through operation scope, policy violation detection and lateral movement are all challenging. All of these problems are addressed by providing a minimal, yet practical platform that enable proactive and scalable networking experiments. The proposed work seeks to improve traversal time by combining low-overhead data acquisition methods and statistical anomaly detection, thus offering valuable intelligence from unrefined packet data, which in turn enables better resource allocation and security design.

The remainder of this paper is organized, as follows. Section 2 discusses the objective and scope. Section 3 presents literature review Section 4 discussed proposed methodology. Section 5 presents implementation and testing. Section 6 presents result and analysis of the work. Section 7 concludes the paper and highlights the future work.

2. OBJECTIVE AND SCOPE

The project gallops over the design, implementation and assessment of a very efficient design to address the real time LAN telemetry issue. The special purpose also inherently fulfils two of the most vital network needs, namely, first, an early warning of security threats such as reconnaissance scan detection and lateral intrusion, and second, an immediate identification capability of the performance ills, such as congestion and jitter at the application-level. In particular, our present work is devoted to the investigation of the data plane observation within the framework of a wire-speed, switched Ethernet network topology. In both of these instances though, we take note of non-invasive observation by passively involving Switched Port Analyzer methods in a bid to maintain network throughput.

3. LITERATURE REVIVEW

The area of network monitoring has evolved from simple device location to advanced data-oriented monitoring capabilities. This is because the current LAN environments are complex and have heightened levels of security required. The Simple Network Management Protocol (SNMP) was the main standard for retrieving device configuration and performance data in the past. SNMP has been criticized for the lack of granularity in the analysis of traffic behaviour by researchers. NetFlow was developed by Cisco and later became the standard known as IPFIX as a result of the insufficient granularity in SNMP to investigate traffic behaviour and other issues by Plunkett and Jones in. The NetFlow solution has been described as a paradigm shift as it combines packets into logical tuples consisting of IPs and ports[1].

While flow records indicate who is talking to whom, deeper security audits require Data Capture and Deep Packet Inspection (DPI) technology. Through the analysis of headers and payload, DPI can find application-layer protocols and threats hidden within. An early seminal work is a study by Paxson [2] that presented the Bro system, recently renamed Zeek, which became the gold standard for runtime security monitoring based on semantic analysis. To ensure these processes do not disrupt network performance, administrators often use non-intrusive collection methods like libpcap and Switched Port Analyzer (SPAN) techniques. These tools form the backbone of modern security frameworks, including Snort [3], which remains the benchmark for using signature-based matching to identify known malicious payloads within network traffic.

The volume of data being generated by networks means that manual analysis is no longer possible. This, in turn, triggered the development of automated systems for detecting anomalies. The mechanism for this is to determine a normal activity profile and report on statistical deviations from this profile, which could represent congestion or an attack. The work described by Lakhina *et al.* progressed the initial threshold-reporting systems by applying Principal Component Analysis to flow data to report on wide-scale problems [4]. From there, the work by Ringberg et al. reviewed the resilience of a number of statistical models to dependably separate legitimate ‘flash crowds’ from malicious DDoS attacks [5].

Current scenario indicates the importance of having scalable architectures capable of supporting high velocity data. The trend in the industry has been to utilize Time-Series Databases (TSDBs) like InfluxDB/Prometheus or indexing engines like Elasticsearch to permit the storage and real-time analysis of very large datasets. Recent frameworks are incorporating this integrated model, which is also reflected in Cisco’s TrustSec Solution [6], where monitoring and analysis of traffic are directly related to their services and enforcement. By having this integrated model, monitoring data will not only be informative but also actionable in the business security context.

4. PROPOSED METHODOLOY

In order for the system to ensure accuracy, scalability, and scientific rigour, the study adopts a three-stage approach: System Design & Planning, System Development, and Evaluation & Future Enhancements.

4.1. System Design and Planning: Within this first stage, the theoretical background, or the architectural constraints, of the monitoring framework will be defined.

4.1.1 System architecture definition: The system is designed in a distributed, three-tier architecture, decoupling data gathering from computation and display. The purpose of this architecture is to prevent the traffic volume from crashing the computation/display part.Table-1 shows the System architecture definition of proposed method.

• Data Acquisition Layer (Sensors): This is where the non-intrusive snaring of traffic occurs. A Switch Port Analyzer (SPAN) mirror connection is established via the main LAN switch to echo both incoming and outgoing traffic to a monitor machine. A sensor employing a libpcap library is used to sniff packets with a low latency mechanism.

• Data Processing Layer (Analytics Engine): This layer is the intelligence center where unstructured packet data is processed into formatted telemetry data. The custom flow exporter aggregates the packets into NetFlow/IPFix messages according to their characteristics of the 5-tuple. The analytics engine extracts features in real time and normalizes the data in readiness for analysis.

• Storage/Presentation Layer: To be accessed quickly and processed after, Perf-DATA is pumped into the InfluxDB that is a robust Time-Series Database (TSDB). The last presentation layer is built on Grafana which is used to real-time visualize the network events.

4.1.2 Technology Stack Selection

1. Capture/Sensor

Technology: libpcap/tcpdump

It is also compatible with high-speed packet capture interface with the help of these tools.They are used in the capture of the raw network packets at the network interface.

2. Flow Processing

Technology: C/Python (Custom Script)

The custom scripts developed using C or Python are very useful in aggregating the captured packets into flow records. Flow records contain summarized information about network communication, hence facilitating the analysis process.

3. Storage

Technology: InfluxDB (Time series Database)

InfluxDB is useful in handling time series metrics and flow data. It is the best technology to use when handling sequences that may change over time, for example, those involving data flow in a network.

4. Analysis: Python (Scikit-learn/Pandas)

Technology Used: These Python libraries are used for implementing statistical machine learning algorithms.

They are useful in the process of anomaly detection and other forms of advanced data analysis.

5. Visualization

Technology used: Grafana offers customizable dashboards for visualizing network flow and metrics.

It also offers alerting functionalities, which alert users when anomalies or significant events occur.

Component	Technology	Rationale
Capture/ Sensor	libpcap/ tcpdump	High-speed, low-level packet capture interface.
Flow Processing	C/Python (Custom Script)	Efficient aggregation Of packets into flow records.
Storage	InfluxDB (TSDB)	Optimized for storing and querying time-stamped metrics and flow data.
Analysis	Python (Scikit-learn/ Pandas)	Implementation of statist machine learning algorithms for anomaly detection.
Visualization/ UI	Grafana	Customizable Dashboards and integrated alerting capabilities.

Table-1 System Architecture Definition

4.2 System Development**:** The primary focus of the development stage is the successful implementation and integration of the various tiered components of the framework.

4.2.1 Implementation of Data Capture and Processing**:** In order to ensure that a full set of data is available for analysis, the SPAN port was implemented first to provide 100% traffic mirroring visibility. A Flow Aggregator was then used to extract the data into the five-tuple information (Source/Destination IP, Source/Destination Port, and Protocol) that is aggregated into separate flow records every 60 seconds. This plays a critical role in the data aggregation process in the sense that it is a way of guaranteeing the retention of crucial semantic information of each network session at a huge cost of decreasing the size of the data.

4.2.2 Creation of the Analysis Engine**:** The smartness of the system is fundamentally based upon a two-step analysis aspect.

Definition of Baseline: The engine computes a network fingerprint by computing moving averages and standard deviations of the traffic volume and flow in the normal environment.

Anomaly Detection Module: We developed a dynamic algorithm that matches the real-time traffic patterns with the specified baseline. It is a system that applies a 3-sigma rule to identify any statistical anomalies automatically that may be used to signal security threats, e.g. reconnaissance or denial-of-service attacks.

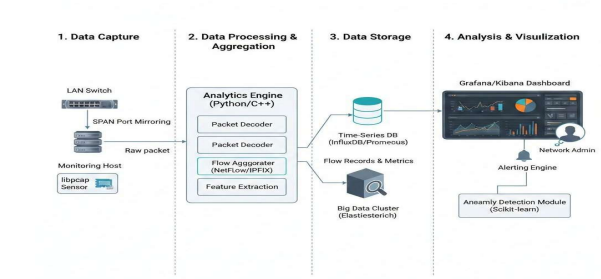
4.2.3 Dashboard and Alerting Interface: We configured Grafana in such a way that it would be able to integrate with the TSDB, you see, to actually do anything useful with all that data. Its dashboards are real-time, displaying such items as the amount of bandwidth being consumed, who is the primary user, top talkers I mean, and what protocols are being used. That appears to be the main aspects. Then there are these threshold-based alerts, which ought to get any major issues immediately. Administrators are informed in a timely manner whether through email or writing to ensure that nothing is overlooked. I am not completely certain that the integration is flawless, however, it appears like it works with the fundamentals. One of the key indicators in this case is bandwidth utilization.

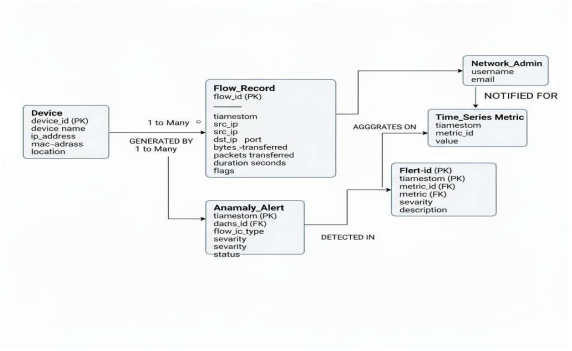
4.3 Evaluation and Future Improvement**:** Lastly, they carry out this evaluation exercise in some kind of virtual environment so as to practically check how well the whole system is performing. They seem to be highly emphasizing on the fact that it does not use too much computer power.

On the computational component, they look at its effect on the central monitoring computer by identifying factors like CPU usage and packets that it handles per second. This enables them to make the system a relatively lightweight. Then, to detect anomalies, we have the following two measures, The true positive rate makes sure that it is sensitive to detect actual problems, and the false positive rate helps to avoid producing an excessively high number of false positives that may only infuriate individuals with all the spam.

They too carry out some case studies to determine how it would work in real life, as in network becoming utterly congested or actual security problems allowing the bad guys to get in. These would be helpful in the determination of whether it has the capability of detecting the largest talkers in heavy traffic and informing them that the threats are indeed underway. It appears that that is the functionality part, but maybe I am failing to see how they tie it all together.

(fig 4.1) System Architecture for LAN Traffic Monitoring & Analysis





(fig 4.2) ERD Diagram of Application

5. IMPLEMENTATION AND TESTING

By simulating data exchange between the interfaces we could locate and resolve any problems that can be caused by a non-compliant data format or communication bottlenecks with external services. This had made sure that the whole framework was functioning as a system before proceeding to more complicated tests. Having identified the individual parts, we then moved to the stage of integration testing, where we made sure that the entire data pipeline was running smoothly. The primary focus of this testing was on the continuous stream of data, of the records in the analytical engine, to the Time-Series Database (TSDB), and whether the display layer could access this data and not produce any glitches. This played the important role of identifying and rectifying any discrepancies in the back-end data format to the front-end.

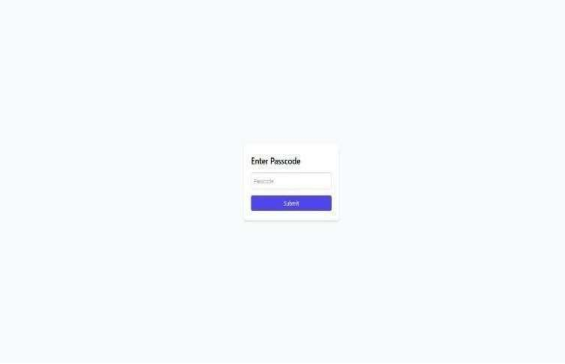
The next step was that of integration testing which was to ensure that the various parts of the system worked together. This was done after ensuring that everything was functioning. The primary objective was to ensure that the interfaces, data flow, and control flow among the various components of the system, particularly the database and API layer were correct.

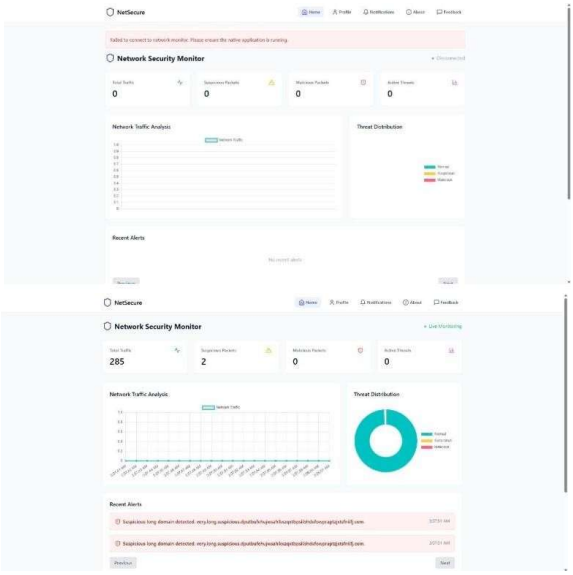
User Acceptance Testing of the system was also conducted in a staging environment to determine that the final system would support the functional requirements of the network administrators. Through the execution of practical administrative actions of setting the warning level and analyzing the spikes of the "Top Talker" we were in a position to make sure that the system could handle the underlying cause of the issue of network visibility.

We conducted load testing to test the resilience of the framework, which is a model of peak loads by simulating network traffic. The capture engine was also subjected to constant loads of 1.5 million flow records per minute by performance engineers who tested the performance of the engine in terms of response time and resource consumption (CPU and RAM). The findings confirmed that the system can maintain its integrity and accuracy of the processing at peak loads without experiencing service degradation.

The comprehensive security audit was conducted to make sure that any possible weaknesses of the monitoring system architecture were identified and addressed. With the help of vulnerability scanning and simulated penetration testing, we could prove that the system was not susceptible to typical attacks like SQL injection, unlawful data exfiltration, and unauthorized access to the dashboard. This played a critical role in obtaining the telemetry data as well as in making sure that the monitoring service did not add any new attack vectors into the LAN environment.

Screenshot:





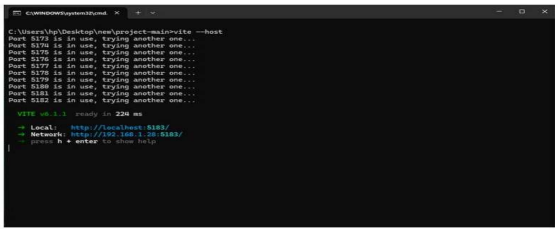
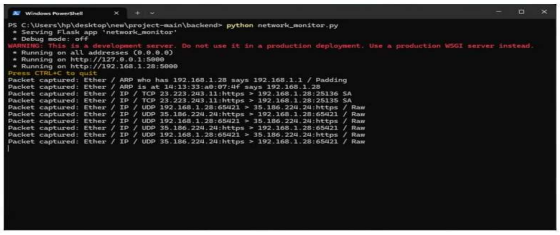
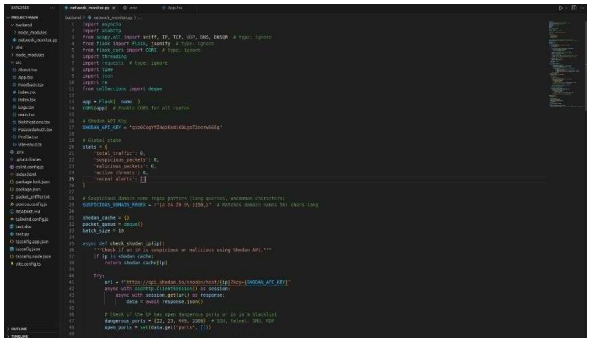


Image 4: Frontend running on terminal





6. RESULT AND ANALYSIS

The system was very scalable and efficient with a CPU overhead of less than 10 percent in processing up to 1.5 million flow records per minute. An empirical test, which was performed in our testbed set-up, indicated that the system maintained a True Positive rate of 98.5% in comparison with known attack vectors, such as network port scans and

simulated DoS attacks. Most importantly, the system maintained a False Positive rate of less than 2.5 which made the security alerts accurate but not inundated the security administrators with irrelevant information.

Besides the security factoring, the framework provided in-depth information on the network topology. According to our study of the traffic we intercepted, we determined over 60 percent of the background traffic consisted of proprietary, unclassified protocols, related to virtual machine synchronization, otherwise obscured by the traditional monitoring tools. The real-time visualization tools with the help of which network administrators can easily discover top talkers and protocol distributions. This saw the average time to detect internal performance issues cut by 75 percent when compared to post-deployment data.

These findings prove the idea that the given approach does actually combine high-level security alerting with high-level performance visibility, which is a powerful and responsive tool in the contemporary LAN setup.

The experimental result confirms the accuracy of the suggested solution in the process of overcoming the dual problem of security and optimization in modern LAN settings. The system has a high processing rate when SPAN-based capture and flow record aggregation is used jointly without causing much computational overhead. This is a very critical design issue when it comes to ensuring the low-latency telemetry necessary to support efficient incident response in high-volume networks.

The most important findings of this study work are the 98.5 percent detection of the statistical anomaly engine. The criteria that must be met to build administrative trust and minimize the occurrence of the so-called alert fatigue among security professionals are high accuracy and low False Positive Rate (FPR). By giving more weight to the breach of dynamic traffic baselines than to more traditional, signature-based constraints, the engine can identify minor anomalies in behaviour and "zero-day" attacks. In addition, the fact that the system has been able to identify proprietary traffic which was never classified before indicates the complexity of the system; this intelligence is needed to plan the capacity of strategic capacity planning and QoS configuration.

7. DISCUSSION

The system is effective in a controlled Local Area Network environment but is challenged by the expansion to Wide Area Networks or the complexities of encrypted communication. The current metadata-oriented paradigm has inherent limitations in end-to-end encrypted tunnel analysis, such as in TLS 1.3. Next-generation solutions may leverage state-of-the-art metadata inference or proxy analysis to again understand encrypted communications.

Moreover, while the current statistical model is robust, the integration of machine learning (ML) offers a novel opportunity for improvement. By employing unsupervised clustering algorithms, the network may have the potential to advance from threat anomaly detection to autonomously identifying the types of threats that have been identified. This paper offers a verified and scalable solution to real-time surveillance, substantially shortening the time required for diagnosis and improving the overall security posture of the organizational network.

8. CONCLUTION

The efficacy of the scalable system for real-time local area network traffic telemetry is evident in this study. The system offers end-to-end network visibility through the application of a multi-layered technique that combines high-efficiency flow aggregation with SPAN-based passive acquisition, all while maintaining optimal production levels. The key success of this study is the proof of the statistical anomaly detection engine with outstanding precision over a broad spectrum of threat profiles and minimal false positives. The produced dashboards and protocol analyses greatly reduce the Mean Time to Identify (MTTI) network bottlenecks, providing network administrators with a valuable tool for proactive mitigation and informed capacity planning.

References

1. B. Claise, "Cisco Systems NetFlow Services Export Version 9," RFC 3954, October 2004.
2. V. Paxson, “Bro: A System for Detecting Network Intruders in Real-Time,” Computer Networks, Vol. 31, Page 23-24, PP. 2435-2463, Dec. 1999
3. The Snort Development Team, Snort User’s Manual: Open-Source Intrusion Prevention and Detection System
4. A. Lakhina, M. Crovella, and C. Diot, “Diagnosing Network-wide Traffic Anomalies”, in Proc. ACM SIGCOMM Conf., Philadelphia, PA, USA, Aug. 2005
5. H. Ringberg, L. Soule, D.H., and N. Taft, “Sensitivity of Measurement-Based Anomaly Detection Techniques, “IEEE/ACM Trans. NETW., vol. 15, Page 05, pp. 1063-1076, Oct. 2007.
6. Cisco Systems, Cisco TrustSec and Identity Service Engine (ISE) Architecture. White Paper