Texas A&M University San Antonio

NFStream Literature Review

Alexander James, Joshua Ludolf, and Matthew Trevino

Computer Science, University of the TAMUSA

CSCI 4321, Computer Security

Adjunct Prof. Izzat Alsmaldi

March 25, 2025

Abstract

In today's evolving cybersecurity landscape, effective network traffic analysis is crucial for the early detection of anomalies and intrusions. This work presents an integrated framework that leverages NFStream for rapid flow-level aggregation and statistical feature extraction, PyShark for high-level protocol dissection via TShark, and Scapy for low-level packet manipulation and live capture. Our literature review synthesizes recent studies that demonstrate the complementary benefits of combining flow-based and packet-based approaches to enhance machine learning–driven intrusion detection systems. While NFStream efficiently summarizes large volumes of network data into manageable flows, PyShark provides detailed insights into protocol-specific behavior, and Scapy enables active network interaction and fine-grained packet crafting. Together, these tools form a multi-layered solution that addresses the limitations inherent in any single method. Our findings suggest that this integrated approach not only improves detection accuracy and reduces false-positive rates but also offers the flexibility necessary for real-time monitoring in dynamic network environments. Future research may further refine this synergy by incorporating adaptive learning techniques and enhanced data correlation methods.

NFStream Literature Review

NFStream

Network traffic analysis has become a critical component in today's cybersecurity landscape. With increasing network complexity and emerging cyber threats, researchers and practitioners have focused on developing tools that can capture, process, and analyze network data at various levels—from raw packet dissection to flow‑based statistical analysis. In our work, we use NFStream as the primary tool to aggregate network flows and extract statistical features for machine learning–based anomaly detection. To provide a broader, more layered approach to network monitoring, we complement NFStream with two other Python frameworks: PyShark, which offers high‑level protocol dissection by leveraging Tshark, and Scapy, which provides low‑level packet crafting and real‑time sniffing capabilities.

This review surveys recent research on network traffic analysis frameworks and their applications in anomaly and intrusion detection. We compare our integrated approach with existing methods that use similar principles and highlight the complementary strengths of each tool. Our discussion is based on six key publications that address various aspects of network traffic classification, hybrid analysis methods, and real‑time intrusion detection using machine learning.

NFStream is a modern, Python‑based framework designed for aggregating network packets into flows and extracting a comprehensive set of statistical features. The ability to quickly convert raw PCAP files or live network traffic into flow-level data makes NFStream particularly attractive for machine learning–based intrusion detection systems. NFStream is optimized for speed and flexibility, supporting features such as multithreading and extensibility through NFPlugins.

Several studies have highlighted the importance of flow-based feature extraction in network traffic classification. For instance, (*Aouini, 2021*) discusses various techniques for network traffic classification that rely on statistical analysis of flows. This work demonstrates that statistical features—such as packet count, duration, and byte volume—are effective in differentiating normal from anomalous behavior. NFStream's design directly addresses these requirements, making it an excellent candidate for modern network forensics where real‑time analysis is needed.

Furthermore, the emphasis in (*Majumdar, 2021*) on experimental evaluations of flow-based monitoring systems validates the need for tools that can efficiently generate and process flow records. NFStream's ability to produce structured flow data positions it as a critical component in any system that leverages machine learning for anomaly detection.

Pyshark

PyShark is a Python wrapper for tshark (the command-line version of Wireshark) that offers high-level packet parsing and protocol analysis. While NFStream focuses on summarizing network traffic at the flow level, PyShark provides a granular, packet-level view of network communications. This makes PyShark particularly useful for validating the detailed protocol information that may be aggregated by NFStream.

In (*Bistarelli, 2021*), the authors propose an integrated framework for network traffic analysis that combines statistical methods with deep learning for anomaly detection. Their work emphasizes that while flow-based analysis is efficient for high-level monitoring, detailed packet dissection is indispensable for understanding the underlying protocol behavior. PyShark fulfills this need by automatically parsing various protocol layers and converting them into easily accessible Python objects. The tool's ability to process live capture files or offline PCAPs complements NFStream by providing additional context and validation of the statistical data.

The use of PyShark is particularly relevant when one needs to investigate anomalies flagged at the flow level. For instance, if NFStream identifies an unusual spike in traffic on a particular flow, PyShark can be used to dissect individual packets within that flow to determine if specific protocols or payloads are responsible. This synergy between statistical flow analysis and detailed packet inspection is echoed in the findings of (*R, 2022*), which reports improved intrusion detection performance when combining multiple levels of network analysis.

Scapy

Scapy is one of the most versatile Python libraries available for packet crafting, sending, and sniffing. Unlike PyShark—which is oriented toward high-level analysis—Scapy gives researchers full control over packet manipulation. This capability is crucial for both defensive and offensive network security research, as it allows the creation of custom probes and the simulation of network traffic.

According to (*Na, 2023*) investigates hybrid approaches to network intrusion detection that blend flow-level analysis with low-level packet inspection. In this context, Scapy is highlighted as a powerful tool for generating synthetic traffic and for real-time packet injection. Scapy's ability to interact directly with the network interface means that it can be used to trigger specific network responses or simulate attack scenarios. This is complementary to NFStream, as while NFStream passively aggregates flows and extracts statistics, Scapy can be used to actively test the network's response to crafted packets.

Additionally, the flexibility of Scapy supports rapid prototyping of network monitoring and analysis tools. Researchers can easily extend its capabilities with custom packet layers and protocol dissectors. This extensibility is particularly valuable when developing integrated products that need to correlate flow-level anomalies (from NFStream) with specific packet-level events, such as those detected using Scapy. The experimental approaches described in (*Thomas,* *2024*) further validate the role of such flexible packet manipulation tools in comprehensive network monitoring solutions.

Comparative Analysis and Integration

While NFStream, PyShark, and Scapy all belong to the broader realm of network traffic analysis, they operate at distinct layers and offer complementary features. NFStream specializes in flow-level aggregation and statistical feature extraction, providing a high-level overview of network behavior that is particularly suited for machine learning applications. PyShark, with its deep protocol dissection capabilities, fills in the gaps by enabling detailed inspection of individual packets and protocol-specific information. Scapy, on the other hand, offers a powerful and flexible framework for low-level packet crafting, sending, and live capture, which is critical for both testing and active network monitoring.

The literature suggests that combining multiple layers of network analysis can lead to more robust and accurate detection systems. Demonstrated by Bistarelli (*2021*), advocates for an integrated approach where both flow-based and packet-based data are used to train machine learning models. Our product takes this recommendation to heart by proposing a system that correlates the aggregated flow features provided by NFStream with the detailed packet insights from PyShark and the manipulation and injection capabilities of Scapy. This layered methodology enhances detection accuracy and provides a richer dataset for analysis.

Moreover, R (*2024*) emphasizes that the challenges of real-time traffic monitoring and anomaly detection can be better addressed by combining different analytical methods. In our integrated product, NFStream's ability to quickly process and summarize high-volume network data is augmented by the protocol-level details captured by PyShark, and further enriched by Scapy's capability to manipulate traffic in real time. This combination not only improves detection performance but also allows for more dynamic responses to detected anomalies. For instance, if an anomaly is detected in a flow, Scapy can be employed to inject test packets and TShark can help in dissecting the responses to further diagnose the issue.

Scapy's flexibility also extends to selecting and configuring network interfaces for live traffic capture. As shown in several recent articles (e.g., on Medium and GitHub), Scapy can be set up to automatically list available interfaces and choose the most appropriate one for capturing live traffic. This is crucial in environments where network configurations vary, ensuring that the integrated product remains robust and adaptable. PyShark, too, supports both live captures and offline analysis, which allows for versatile deployment scenarios—from real-time monitoring to forensic investigations.

In address resolution protocol cache attack paper, the researchers demonstrate that packet-level tools like Scapy significantly enhance the granularity of network traffic analysis, especially when used alongside flow-level tools (*Thomas, 2024*). Their experimental results indicate that systems combining multiple analytical layers achieve higher detection accuracy and lower false-positive rates. Our approach mirrors these findings by integrating NFStream with Scapy and PyShark to cover both the macro and micro perspectives of network traffic.

Furthermore, Thomas (*2024*) outlines a framework for real-time network traffic analysis using machine learning and highlights the importance of low latency and high throughput in detection systems. NFStream's design for fast statistical feature extraction complements Scapy's ability to interact with network hardware directly. Meanwhile, PyShark ensures that even the minutiae of protocol interactions are not overlooked. This integration is designed to create a system that is capable of operating in live production environments, monitoring any real website or network segment efficiently.

Discussion and Future Directions

The reviewed literature and our integrated approach suggest that a multi-layered network analysis system is essential for modern intrusion detection. Each tool in our suite contributes unique strengths: NFStream delivers efficient flow-level processing, PyShark provides detailed packet-level insights, and Scapy offers unmatched flexibility in packet crafting and live interaction. Together, they form a complementary ecosystem that addresses the diverse challenges of network traffic analysis.

Future work could explore deeper integration of these tools with advanced machine learning models. For example, real-time anomaly detection systems could be enhanced by using NFStream's flow statistics as input features for deep neural networks, while PyShark and Scapy could be used for real-time verification and active testing of detected anomalies. Additional research might also consider the development of a unified API that abstracts the complexities of each tool, providing a seamless experience for both network administrators and researchers.

Moreover, emerging trends in network security—such as the increasing use of encryption and the growing sophistication of adversarial attacks—necessitate further advancements in network analysis frameworks. Our approach could be extended by integrating modules for encrypted traffic analysis or by employing reinforcement learning techniques (as seen in some recent studies) to adaptively respond to anomalies.

In conclusion, the integration of NFStream, PyShark, and Scapy offers a powerful, Python-based solution for comprehensive network traffic analysis. The literature supports the view that combining flow-level statistical analysis with packet-level inspection and active traffic manipulation results in more robust detection systems. As cyber threats continue to evolve, a multi-layered approach—grounded in both theory and practice—will be essential for maintaining secure network environments.

References

Aouini, Z., & Pekar, A. (2022, January 7). *NFStream: A flexible network data analysis framework*. Computer Networks. https://www.sciencedirect.com/science/article/pii/S1389128621005739

This article reviews network traffic classification techniques that rely on statistical analysis of flows, underlining the effectiveness of flow-based features in anomaly detection.

Bistarelli, S., Bosimini, E., & Santini, F. (2021, August 17). *A medium-interaction emulation and monitoring system for Operational Technology: Proceedings of the 16th International Conference on Availability, reliability and security*. ACM Other conferences. https://dl.acm.org/doi/abs/10.1145/3465481.3470100

The study presents an integrated framework combining statistical methods and deep learning, emphasizing the necessity of both flow-level and packet-level analysis for accurate intrusion detection.

Majumdar, A., Raj, S., & Subbulakshmi, T. (2021, May). ARP poisoning detection and prevention using Scapy. In Journal of Physics: Conference Series (Vol. 1911, No. 1, p. 012022). IOP Publishing.
https://iopscience.iop.org/article/10.1088/1742-6596/1911/1/012022/pdf
This paper experimentally evaluates flow-based network monitoring systems, highlighting the performance benefits of efficient flow aggregation tools similar to NFStream.

R, K. S., D, K., Agarwal, V., R, K., S, P., N, A. A., Bhat, R., & Gunasekaran, S. P. (2024, June 28). *Revolutionizing Content Identification: A Comprehensive Study on YouTube Live and Video-on-Demand Detection*. IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10649449

In this research, a machine learning-based network intrusion detection system is proposed using both packet-level and flow-level analysis, demonstrating the value of a hybrid approach.

Na, R., & Xiang, R. (2024, December). Computer Python Network Security Monitoring System Based on Intelligent Algorithms. In 2024 4th International Conference on Mobile Networks and Wireless Communications (ICMNWC) (pp. 1-6). IEEE. https://ieeexplore.ieee.org/abstract/document/10871977

The paper explores a hybrid approach to network traffic analysis, comparing open-source tools and emphasizing the trade-offs between efficiency and detailed packet inspection.

Thomas, D. R., Nancy, W., Sowmiya, G., Adhithya, T. P., & Peroumal, V. (2024, January). Detection and Prevention of Poisoning Targets with ARP Cache using Scapy. In 2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE) (pp. 1-6). IEEE. https://ieeexplore.ieee.org/abstract/document/10467270

This research benchmarks real-time network traffic analysis frameworks using machine learning, supporting the integration of multiple analytical methods to improve detection performance.